



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

OBJETOS DE APRENDIZAJE: METODOLOGÍA DE DESARROLLO Y EVALUACIÓN DE LA CALIDAD

Tesista: Mg. Lic. Stella Maris MASSA

Directores: Ing. Armando DE GIUSTI y Lic. Patricia PESADO

TESIS PRESENTADA PARA OBTENER EL GRADO DE
DOCTOR EN CIENCIAS INFORMÁTICAS

La Plata, Buenos Aires, Argentina
2012

A mis hijos: Pablo, Martín, Matías y Julián

INDICE

INDICE.....	I
INDICE DE FIGURAS	IV
INDICE DE TABLAS	VI
NOMENCLATURA	VIII
AGRADECIMIENTOS	X
RESUMEN	XI
ABSTRACT	XI
CAPITULO 1.INTRODUCCIÓN	1
1.1. Motivación.....	1
1.2. Objetivos de la tesis.....	2
1.3. Estructura general de la tesis	2
1.4. Producción científica derivada de resultados parciales de la tesis	3
CAPITULO 2. CONCEPTOS BÁSICOS	5
2.1. Gestión y generación del conocimiento en la era de las tecnologías de la información y la comunicación	5
2.2. Entornos virtuales de enseñanza aprendizaje	6
2.3. Objetos de aprendizaje.....	10
2.3.1. Definición	11
2.3.2. Atributos de los OA	13
2.3.3. Componentes de un OA o estructura de un OA.....	14
2.3.4. Ventajas y desventajas	15
2.3.5. Repositorios de OA	15
CAPITULO 3. ELEMENTOS DE ANÁLISIS EN LOS OBJETOS DE APRENDIZAJE. 18	
3.1. Especificaciones y estándares en e-learning.....	18
3.1.1. AICC (Aviation Industry CBT Comitee).....	21
3.1.2. IEEE learning object meta-data / IMS learning resource metadata specification (LOM).....	22
3.1.3 Dublin Core	24
3.1.4. IMS learning resource meta-data specification	25
3.1.4.1. IMS Content Packaging (Empaquetado de contenidos IMS-CP).....	25
3.1.4.2. IMS Learning Design (Diseño de aprendizaje IMS-LD)	27
3.1.4.3. IMS Question & Test Interoperability Specification (Interoperabilidad test y cuestiones IMS-QTI)	30
3.1.4.4. IMS Common Cartridge specification (Cartucho común IMS-CC).....	31
3.1.5. ADL/SCORM	33
3.1.6. ISO/IEC 19788 MLR	40
3.1.7. Perfiles de aplicación LOM.....	41
3.2 Dimensiones de análisis de un OA	43
3.2.1. Aproximación pedagógica	43
3.2.2. Aproximación tecnológica: las teorías del aprendizaje y el diseño instruccional	46
3.3. Creación de OA	51
3.3.1. Empaquetado de OA.....	52
3.3.2. Etiquetado de OA	54
CAPITULO 4. METODOLOGÍAS DE DISEÑO DE UN OBJETO DE APRENDIZAJE (OA).....	56

4.1. Ingeniería de software educativo	56
4.2. Metodologías de desarrollo de la IS	58
4.2.1. Modelo de proceso RUP (Rational Unified Process)	58
4.2.1.1. Modelo de proceso RUP para software educativo.....	61
4.2.2. Modelo de prototipos evolutivos con refinamientos sucesivos	63
4.2.2.2. Modelo de prototipos evolutivos con refinamientos sucesivos para software educativo.....	64
4.2.3. Sistemas interactivos	67
4.2.3.1. Diseño centrado en el usuario (DCU)	69
4.2.3.2. Modelo de proceso MPIU+a	73
CAPITULO 5. PARÁMETROS DE CALIDAD DE UN OBJETO DE APRENDIZAJE (OA).....	76
5.1. Introducción.....	76
5.2. La calidad en la Ingeniería de Software	77
5.3. La calidad de un recurso digital educativo	79
5.4. Métodos de evaluación de usabilidad.....	79
5.4.1. Clasificación de los métodos de evaluación de usabilidad.....	79
5.4.2. Métodos de evaluación de la usabilidad	82
CAPITULO 6. METODOLOGÍA DE DESARROLLO Y EVALUACIÓN DE UN OBJETO DE APRENDIZAJE (OA)	92
6.1. Introducción.....	92
6.2. MPOBA: Modelo de Proceso para el desarrollo de Objetos de Aprendizaje.....	93
6.2.1. Fases del modelo MPOBA	94
6.2.1.1. Prototipado	94
6.2.1.2. Planificación del OA	100
6.2.1.3. Elicitación y especificación de requerimientos	100
6.2.1.4. Diseño.....	105
6.2.1.5. Puesta en funcionamiento del OA	114
6.2.1.6. La hoja de trabajo del modelo MPOBA	122
CAPITULO 7. INSTRUMENTOS E INDICADORES	125
7.1. Introducción.....	125
7.2. Instrumento para la evaluación por expertos.....	127
7.2.1. Trabajos relacionados	127
7.2.2. Plantilla GEHOA (Guía de evaluación heurística para Objetos de Aprendizaje)	131
7.3. Instrumento para la evaluación por estudiantes.....	141
7.3.1. Trabajos relacionados	141
7.3.2. Instrumento CUSEOA (Cuestionario de satisfacción de estudiantes de un OA)	148
CAPITULO 8. TRABAJO EXPERIMENTAL: OBJETO DE APRENDIZAJE PARA UN CURSO DE PROGRAMACIÓN INICIAL O ALGORITMIA	152
8.1. Introducción.....	152
8.2. Aplicación del Modelo de proceso MPOBA	155
8.2.1. Planificación del OA	156
8.2.2. Elicitación y Especificación de requerimientos.....	160
8.2.3. Diseño.....	168
8.2.4. Puesta en funcionamiento del OA	178
9. CONCLUSIONES Y LÍNEAS FUTURAS DE INVESTIGACIÓN Y DESARROLLO	209
9.1. Conclusiones.....	209

9.2. Futuras líneas de investigación y desarrollo.....	211
CAPITULO 10. REFERENCIAS	212
APENDICE A. MATERIAL DEL OA	240
A.1. Escenarios	240
A.2. Introducción teórica: pasaje de parámetros	247
A.3. Enunciados de los problemas para los videos.....	255
A.4. Enunciados de los problemas para actividades.....	255
APENDICE B. PROBLEMAS RESUELTOS	256
APENDICE C. EVALUACION DE EXPERTOS.....	266
C.1. Plantilla GEHOA (Guía de evaluación heurística para Objetos de Aprendizaje)..	266
C.2. Resultados de la evaluación heurística	269
C.3. Estadísticos comparativos de los puntajes	272
C.4. Puntajes ponderados	272
APENDICE D. EVALUACION POR ESTUDIANTES.....	273
D.1. Instrumento CUSEOA (Cuestionario de satisfacción de estudiantes de un Objeto de Aprendizaje)	273
D.2. Puntajes parte a del instrumento CUSEOA	274
D.3. Valoración de ítems de la parte b del instrumento CUSEOA.....	274
D.4. Distribución de frecuencias de ítems parte b.....	276
D.5. Gráficos de sectores de ítems parte b	279
D.6. Registros de LOGS	286
D.7. Histogramas de las variables en cada categoría.....	287
D.8. Pruebas de normalidad.....	301
D.9. Pruebas de diferencias de medias	302

INDICE DE FIGURAS

Figura 2.2 . CMS	9
Figura 2.2 . LMS	9
Figura 2.3 . LCMS.....	10
Figura 2.4. Estructura de un Objeto de Aprendizaje.....	15
Figura 2.5. Repositorio de OA (ROA).....	16
Figura 3.1. Pasos para establecer estándares de e-learning	19
Figura 3.2. Clasificación de Metadatos	21
Figura 3.3: Representación de la jerárquica de elementos en Modelo de datos LOM.....	24
Figura 3.4. IMS Content Packaging alcance	27
Figura 3.5. Integración de IMS-LD dentro de IMS-CP.....	28
Figura 3.6. Estructura de alto nivel de una UDA en IMS-LD.....	29
Figura 3.7. Estructura de un examen dentro de un paquete IMS.....	31
Figura 3.8. Formato de intercambio IMS-CC.....	32
Figura 3.9. Arquitectura conceptual de SCO.....	35
Figura 3.10. Representación conceptual de las actividades.....	36
Figura 3.11. Ilustración conceptual de una Organización de contenidos.	36
Figura 3.12. Ilustración conceptual de Agregación de Contenido.....	37
Figura 3.13. Entorno de Ejecución (RTE).	38
Figura 3.14. Relación entre una Organización de Contenidos y el Árbol de actividad	39
Figura 3.15. La biblioteca SCORM.....	39
Figura 3.16. Pasos a seguir para la creación de OAs.....	52
Figura 3.17. Ejemplo de un Paquete de Agregación de Contenidos y sus componentes	53
Figura 4.1 Capas de IS.....	57
Figura 4.2. Dimensiones del proceso RUP.....	60
Figura 4.3: Esquema del ciclo de vida del prototipado evolutivo	66
Figura 4.4. Proceso iterativo del DCU según ISO 13407.....	70
Figura 4.5. Esquema MPIu+a.	73
Figura 5.1. Clasificación de los Métodos de Evaluación.	80
Figura 5.2. Métodos de evaluación de usabilidad	82
Figura 5.3. Mapa de color y diagrama de visión	90
Figura 6.1. Modelo de proceso MPOBA.....	93
Figura 6.2. Prototipo en papel de Jack Dorsey del status de twitter.....	95
Figura 6.3. Storyboard de un sistema remoto de control de funcionalidades de una vivienda	96
Figura 6.4. Maqueta digital.....	97
Figura 6.5. Storyboard navegacional	98
Figura 6.6. Prototipo de software educativo.....	100
Figura 6.7. Construcción del OA con la herramienta de autor ExeLearning.....	115
Figura 6.8. Incorporación de Metadatos del OA con la herramienta de autor ExeLearning	115
Figura 6.9. Empaquetado del OA con la herramienta de autor ExeLearning.....	116
Figura 6.10. Incorporación de un OA (SCORM) en Moodle	117
Figura 6.11. Vista de un OA en la Plataforma Moodle.....	118
Figura 6.12. Reportes de Moodle de un OA (SCORM)	122
Figura 6.13. Hoja de trabajo del modelo MPOBA	124
Figura 7.1. Extracto de la rúbrica “Calidad de los contenidos” de LORI.	129

Figura 7.2. Cuestionario SUS	144
Figura 7.3. Parte 3: Impresión General del Usuario	146
Figura 8.1. Pantalla de un Curso de la Plataforma de la Facultad de Ingeniería.....	159
Figura 8.2. Editor de símbolos Genero incorporado a la Plataforma de la Facultad de Ingeniería.....	159
Figura 8.3. Roles de la Plataforma Moodle.....	160
Figura 8.4 Clases de Laboratorio de la Asignatura Computación.....	162
Figura 8.5. Tarjetas para la técnica Card Sorting.....	169
Figura 8.6. Tarjetas disponibles.....	169
Figura 8.7. Ordenamiento de las tarjetas por los expertos.....	170
Figura 8.8. Recorrido ideal del OA: de acuerdo a la jerarquía de los expertos.....	170
Figura 8.9. Recorrido del OA.....	171
Figura 8.10. Diagrama de Navegación.....	172
Figura 8.11. Prototipo 1 en papel para videos.....	173
Figura 8.12. Prototipo 2 en papel para videos.....	174
Figura 8.13. Prototipo en papel para Problemas Resueltos.....	174
Figura 8.14. Maqueta digital para el video 1 – Versión 1.....	175
Figura 8.15. Maqueta digital para el video 1 – Versión 2.....	176
Figura 8.16. Maqueta digital para el video 4 – Versión 1.....	176
Figura 8.17. Maqueta digital para el video 4 – Versión 2.....	177
Figura 8.18 . Storyboard para el video 1.....	177
Figura 8.19. Storyboard para el video 4.....	178
Figura 8.20. Pantalla correspondiente al Objetivo del OA.....	179
Figura 8.21. Pantalla correspondiente a la presentación del Video1.....	179
Figura 8.22. Pantalla correspondiente a la Introducción Pasaje de parámetros por referencia.....	180
Figura 8.23. Pantalla correspondiente a la Presentación del video 4.....	180
Figura 8.24. Pantalla correspondiente a la Presentación de los Problemas resueltos.....	180
Figura 8.25. Incorporación de Metadatos en el OA. Estándar Dublin Core.....	182
Figura 8.26. Incorporación del prototipo de software (SCORM) en un Curso de Prueba de la Plataforma Educativa.....	183
Figura 8.27. Vista del OA en la Plataforma Moodle.....	183
Figura 8.28. Entrenamiento previo a la evaluación.....	184
Figura 8.29. Evaluación heurística.....	185
Figura 8.30. Observación del experto.....	189
Figura 8.31. Recorrido Cognitivo.....	189
Figura 8.32. Estructura del OA modificada.....	193
Figura 8.33. Modificación de fuente y tamaño.....	194
Figura 8.34. Modificación de espacios.....	194
Figura 8.35. Modificación del video 1.....	195
Figura 8.36. Modificación del video 4.....	195
Figura 8.37. Modificación de textos.....	196
Figura 8.38. OA en la Plataforma Educativa Moodle.....	196
Figura 8.39. Recomendación de los estudiantes.....	199

INDICE DE TABLAS

Tabla 6.1. Esquema de escenario de Leite.....	99
Tabla 7.1. Descripción de las 9 dimensiones de LORI.....	128
Tabla 7.2a. Dimensiones, categorías y subheurísticas.....	136
Tabla 7.2b. Dimensiones, categorías y subheurísticas.....	137
Tabla 7.2c. Dimensiones, categorías y subheurísticas.....	138
Tabla 7.2c. Dimensiones, categorías y subheurísticas.....	139
Tabla 7.3. Pesos de cada categoría	140
Tabla 7.4: Criterios de evaluación de usabilidad en recursos educativos de Nokelainen .	143
Tabla 7.5. Categorías y criterios heurísticos.....	151
Tabla 7.6. Pesos de cada categoría	151
Tabla 8.1. Programa de Computación	161
Tabla 8.2. Resultados de la percepción de los estudiantes en relación al estudio	163
Tabla 8.3. Escenario 1: video explicando pasaje de parámetros por valor con variables simples.....	166
Tabla 8.4. Escenario 5: Problemas resueltos	167
Tabla 8.5. Escenario 7: Actividad grupal.....	167
Tabla 8.6. Pesos de cada categoría	186
Tabla 8.7. Medidas estadísticas de la Evaluación Heurística	188
Tabla 8.8. Escala de Diferencial Semántico	197
Tabla 8.9. Valores medios y Desviación típica.....	198
Tabla 8.10. Estadísticos del Grupo A - Parte 1.....	202
Tabla 8.11. Estadísticos del Grupo A - Parte 2.....	202
Tabla 8.12. Estadísticos del Grupo B - Parte 1.....	203
Tabla 8.13. Estadísticos del Grupo B - Parte 2.....	203
Tabla A.1. Escenario 1: video explicando pasaje de parámetros por valor con variables simples.....	240
Tabla A.2. Escenario 2: video explicando pasaje de parámetros por referencia con variables simples.....	241
Tabla A.3. Escenario 3: video explicando pasaje de parámetros por valor con variables estructuradas	242
Tabla A.4. Escenario 4: video explicando pasaje de parámetros por referencia con variables estructuradas	243
Tabla A.5. Escenario 5: Problemas resueltos.....	244
Tabla A.6. Escenario 6: Actividad individual: autoevaluación	245
Tabla A.7. Escenario 7: Actividad grupal	245
Tabla A.8. Escenario 8: Actividad de pares	246
Tabla C.1. Instrumento GEHOA	268
Tabla C.2. Puntajes de expertos en el instrumento GEHOA	271
Tabla C.3. Estadísticos comparativos en cada categoría para cada experto	272
Tabla C.4. Puntajes ponderados.....	272
Tabla D.1. Parte A del instrumento CUSEOA.....	273
Tabla D.2. Ítems Parte B del instrumento CUSEOA.....	273
Tabla D.3. Puntajes de la parte A.....	274
Tabla D.4a. Valoración de ítems parte B	275
Tabla D.4b. Valoración de ítems parte B	275
Tabla D.4c. Valoración de ítems parte B	276

Tabla D.4d. Valoración de ítems parte B	276
Tabla D.5. Los objetivos indican lo que se espera que sea aprendido.....	276
Tabla D.6. El nivel de dificultad de los contenidos fue elevado para mis conocimientos previos	277
Tabla D.7. El material teórico me ayudó a comprender los conceptos	277
Tabla D.8. Las actividades han sido claras y significativas para mi aprendizaje	277
Tabla D.9. El sistema informa sobre mi progreso	277
Tabla D.10. Las pistas sobre los errores cometidos son inútiles	277
Tabla D.11. El texto es conciso y preciso	277
Tabla D.12. Los títulos son inadecuados, no se sabe cual es la acción que se debe realizar	278
Tabla D.13. Las imágenes empleadas me ayudaron a aclarar los contenidos	278
Tabla D.14. Me encontré perdido cuando recorría el recurso, no sabía dónde me encontraba	278
Tabla D.15. Los videos y las animaciones me ayudaron a aclarar los contenidos	278
Tabla D.16. La información está mal organizada	278
Tabla D.17. En general, los colores y el diseño de todo el recurso son adecuados	278
Tabla D.18. Recomendaría este recurso a otra persona	279
Tabla D.19. Registros de Logs de las categorías	287
Tabla D.20 a. Prueba de normalidad.....	301
Tabla D.20 b. Prueba de normalidad	301
Tabla D.21. Prueba T-STUDENT	302
Tabla D.22. Prueba U de Mann-Whitney	302

NOMENCLATURA

ADL	Advanced Distributed Learning
AICC	Aviation Industry CBT Comitee
CAM	Content Aggregation Model
CANCORE	Canadian Core Learning Resource Metadata Application Profile
CMS	Content Management System o Course Management System
CUSEOA	Cuestionario de satisfacción de estudiantes de un OA
DCU	Diseño Centrado en el Usuario
ET	Eye Tracking
EVEA	Entorno Virtual de Enseñanza Aprendizaje
GEHOA	Guía de Evaluación Heurística para Objetos de Aprendizaje
HEODAR	Herramienta de Evaluación de Objetos Didácticos de Aprendizaje Reutilizables
IS	Ingeniería de Software
IMS	Instructional Management System
IMS-CC	Common Cartridge specification
IMS-CP	IMS Content Packaging
IMS-LD	IMS Learning Design
IMS-QTI	Question & Test Interoperability Specification
IMS-SS	Simple Sequencing
IPO	Interacción Persona-Ordenador
IS	Ingeniería de Software
IU	Ingeniería de Usabilidad
LCMS	Learning Content Management Systems
LMS	Learning Management System
LOM	Learning Object Meta-data
LORI	Learning Object Review Instrument
MLR	Metadata Learning Resource
MOSCA	Modelo Sistémico de Calidad
MPIu+a	Modelo de Proceso de la Ingeniería de la usabilidad y de la accesibilidad
MPOBA	Modelo de Proceso para el desarrollo de Objetos de aprendizaje

OA	Objeto de Aprendizaje
PMLQ	Pedagogically Meaningful Learning Questionnaire
QUIS	Questionnaire for User Interface Satisfaction
ROA	Repositorio de Objetos de aprendizaje
RTE	Run-Time Environment
RUP	Rational Unified Process
SCO	Sharable Content Object
SCORM	Sharable Courseware Object Reference Model
SS	Modelo de Secuenciamiento y Navegación
SUMI	Software Usability Measuring Inventory
SUS	System usability scale
TIC	Tecnología de la información y la comunicación
UA	Unidad Asignable
UDA	Unidad de Aprendizaje
ZDP	Zona de Desarrollo Próximo

AGRADECIMIENTOS

A mis seres queridos por todo el amor que me dan en cada momento.

A mis padres por su vida, su ejemplo y sostén.

A mis directores, Armando y Patricia por su guía y apoyo durante el desarrollo de este trabajo de tesis.

A la Facultad de Informática de la Universidad Nacional de La Plata por la oportunidad de crecimiento académico.

A la Facultad de Ingeniería de la Universidad Nacional de Mar del Plata por facilitarme el espacio para mi desarrollo profesional.

A mis amigos María Elsa Fernández, Adriana Pirro y Carlos Rico por su ayuda desinteresada desde lo personal y lo profesional en todo momento.

A mis compañeros de cátedra, en especial a María Laura Distéfano, Felipe Evans, Antonio Marsiglio y Horacio Mateos por su participación en la parte experimental.

A mi amigas de la vida Leo y Valeria por escucharme y aconsejarme.

A mis estudiantes por enseñarme cada día.

¡A todos gracias por acompañarme!

RESUMEN

El propósito de esta tesis es formular una metodología transferible de desarrollo y evaluación de la calidad de Objetos de Aprendizaje (OA).

En principio se analizaron los diseños: pedagógico, funcional, gráfico, tecnológico y operacional de las propuestas existentes y se elaboraron criterios de valoración de un OA de acuerdo a las funcionalidades, los estándares y el contexto educativo de implementación.

El Modelo de Proceso propuesto contempla en cada una de sus fases iterativas los aspectos tecnológicos y pedagógicos del OA. Se integraron las actividades del Diseño Centrado en el Usuario con la metodología de desarrollo de software de sistemas interactivos.

En el trabajo experimental presentado se detallan y documentan tanto el proceso de elaboración, como los resultados obtenidos de la aplicación del Modelo de Proceso, a modo de prueba concreta del trabajo de Tesis.

ABSTRACT

The purpose of this thesis is to formulate a transferable methodology of the development and assessment of the quality of Learning Objects (LO).

At first the pedagogical, functional, graphic, technological and operational designs of the existing approaches were analyzed and the assessment criteria of a LO according to the functions, the standards and the educational context of implementation were elaborated.

The Process Model proposed in each one of its iterative phases takes into account the technological and pedagogical aspects of the LO. The activities of the User Centered Design (UCD) were integrated with the development of interactive systems software methodology.

In the presented experimental work, the process of elaboration and the results obtained from the implementation of the Process Model are specified and documented as concrete proof of the Thesis work.

CAPITULO 1.INTRODUCCIÓN

Se presenta en este capítulo la motivación de la tesis (sección 1.1), sus objetivos (sección 1.2), la estructura general (sección 1.3) y la producción científica derivada de resultados parciales de la misma (sección 1.4).

1.1. Motivación

La sociedad en su conjunto necesita proporcionar los medios y los recursos para dar respuesta a las nuevas formas de enseñanza y aprendizaje en las que están implícitas las Tecnologías de la información y la comunicación (TICs), el paradigma educativo y la socialización, entre otros. Para llevar a cabo este proceso el docente ha hecho uso de diferentes recursos didácticos como el pizarrón, presentaciones y recientemente los recursos digitales educativos que permiten guiar y apoyar la educación presencial los cuales no son transferibles automáticamente al aprendizaje basado en la Web.

Estas acciones generan una acumulación del capital académico que se da por ejercicio de la enseñanza, y que puede trascender el uso individual en el aula al ponerlo a disposición en red.

La aparición del concepto de Objeto de Aprendizaje (OA), permite considerar los recursos educativos como unidades mínimas de contenido didáctico con sentido por sí mismos que sirvan de anclaje para aprendizajes posteriores.

De acuerdo con Chiappe Laverde (2009), un acercamiento apropiado al concepto de OA, nos conduce a entenderlo como una mediación del aprendizaje circunscrita dentro del fenómeno de la incorporación de TICs, como un cierto material educativo digital.

Asimismo, los OA, deben poseer elementos motivadores, interactivos, haciendo uso de la narrativa hipertextual con instancias comunicativas y colaborativas. Entonces, en su creación, convergen varias áreas de conocimiento como: Educación, Ingeniería de Software, Tecnología, Diseño multimedial y Comunicación.

El desafío es lograr un equilibrio inteligente para generar OA de mayor calidad integrándolos a los ambientes de aprendizaje y propiciando nuevos recorridos de la currícula que favorezcan el desarrollo de las competencias previstas en el mismo. En este sentido, es necesario generar metodologías para evaluarlos en todos los momentos del proceso de desarrollo a través de diferentes criterios, instrumentos y técnicas, indicando

además quiénes deben participar en esta tarea.

1.2. Objetivos de la tesis

El propósito de esta tesis es formular una metodología transferible de desarrollo y evaluación de la calidad de Objetos de Aprendizaje (OA).

Para ello se analizan el diseño pedagógico, funcional, gráfico, tecnológico y operacional de las propuestas existentes y se elaboran criterios de valoración de un OA de acuerdo a las funcionalidades, los estándares y el contexto educativo de implementación.

Asimismo se presenta un trabajo experimental correspondiente a la aplicación del Modelo de Proceso desarrollado a un OA para un Curso de Programación inicial o Algoritmia. Se detallan y documentan tanto el proceso de elaboración, como los resultados obtenidos de la aplicación del Modelo de Proceso, a modo de prueba concreta del trabajo de Tesis.

1.3. Estructura general de la tesis

En el capítulo 2 se desarrollan los conceptos básicos, que constituyen los marcos generales de referencia que sustentan esta investigación. Se realiza una introducción a la gestión y generación del conocimiento en la era de las TICs. Luego se caracteriza a los Entornos Virtuales de Enseñanza Aprendizaje (EVEA) mediante la descripción de las particularidades de estos sistemas de acuerdo al tipo de información que pueden gestionar. Finalmente se introduce el concepto de OA, se analizan las diversas definiciones presentes en la literatura, se describen sus características, atributos y los problemas que aún se encuentran en debate.

En el capítulo 3 se presentan las herramientas y conocimientos sobre estándares y especificaciones de e-learning requeridos para la creación de un OA. Luego se describen las concepciones referidas a la utilización y creación de un OA desde las perspectivas pedagógica y tecnológica. Finalmente se detallan los pasos para la creación de un OA bajo los estándares y especificaciones de e-learning.

En el capítulo 4 se analizan las consideraciones propuestas por varios autores en cuanto al desarrollo de software educativo. Luego a partir de la investigación bibliográfica, se presentan distintas metodologías de desarrollo de la Ingeniería de Software (IS) adaptables a la construcción de OA. Se destaca además el marco de trabajo de esta tesis, el cual integra las actividades del Diseño Centrado en el Usuario (DCU) con la metodología de

desarrollo de software de sistemas interactivos.

En el capítulo 5 se presenta una introducción sobre la perspectiva de calidad de un OA que se aborda en esta tesis. A continuación se introducen los conceptos de calidad de la Ingeniería de Software en donde la usabilidad juega un papel importante. Luego se expone la visión de la tesis en cuanto a los criterios de evaluación de calidad de un OA. Finalmente, teniendo en cuenta las actividades necesarias para la evaluación de la calidad del OA, se presentan una clasificación de los métodos de evaluación de la usabilidad y una descripción de los mismos.

En el capítulo 6 se expone el aporte de esta tesis: una propuesta de Metodología de desarrollo y evaluación de OA. Se desarrolla el Modelo de Proceso para el desarrollo de Objetos de aprendizaje (MPOBA) con la descripción de cada una de sus fases.

En el capítulo 7 se realiza una revisión bibliográfica de los instrumentos más utilizados para la evaluación de la calidad de un OA. Finalmente se presentan los instrumentos desarrollados en el marco del Modelo de Proceso MPOBA correspondientes a la evaluación del OA por parte de expertos y estudiantes y sus indicadores de calidad respectivos.

En el capítulo 8 se presentan los resultados experimentales correspondientes a la aplicación del Modelo de Proceso MPOBA a un OA para un Curso de Programación inicial o Algoritmia. Se detallan y documentan tanto el proceso de elaboración, como los resultados obtenidos de la aplicación del Modelo de Proceso. Se realiza el análisis de los resultados obtenidos y las conclusiones que se derivan de los mismos.

En el capítulo 9 se exponen las conclusiones de la tesis y las futuras líneas de investigación.

1.4. Producción científica derivada de resultados parciales de la tesis

a) Algunos de los resultados de esta tesis y otros relacionados con esta investigación fueron presentados en eventos científicos nacionales e internacionales y publicados en los libros de resúmenes y/o memorias de dichos eventos, como:

- Massa S.M., Pirro A. L. (2011).” Integración de un entorno virtual con los sistemas de gestión docente y de formación académica en la Universidad”. Conferencia Internacional ICDE (International Council for Open and Distance Education), Buenos Aires.

- Massa S.M., Pirro A. L., Fernández, M.E. y Daher, N. (2011) . “Métricas de calidad de Objetos de Aprendizaje: una mirada pedagógica entrelazada con la tecnología. VI Congreso de Tecnología en Educación y Educación en Tecnología (TEYET 2011), Salta.
- Massa S.M., De Giusti, Pesado P. (2011) . “MPOBA: un Modelo de Proceso para el desarrollo de Objetos de Aprendizaje”. XVII Congreso Argentino de Ciencias de Computación (CACIC 2011). La Plata.
- Massa S.M., De Giusti, Pesado P. (2012). “Métodos de evaluación de usabilidad: una propuesta de aplicación en Objetos de Aprendizaje”. WICC 2012 | XIV Workshop de Investigadores en Ciencias de la Computación. Posadas. Argentina.
- Massa S.M., Rico, C., Huapaya, R. (2012). “Generación de requerimientos de un Objeto de Aprendizaje a partir de escenarios: un caso de estudio para un curso de Programación Inicial”. XVIII Congreso Argentino de Ciencias de Computación (CACIC 2012). Bahía Blanca.

b) Revistas con referato

- Massa S.M., Pesado P. (2012). Revista TE&ET. “Evaluación de la usabilidad de un Objeto de Aprendizaje por estudiantes”. En etapa de evaluación.

CAPITULO 2. CONCEPTOS BÁSICOS

En este capítulo se desarrollan los conceptos básicos, que constituyen los marcos generales de referencia que sustentan esta investigación. En la sección 2.1, se realiza una introducción a la gestión y generación del conocimiento en la era de las tecnologías de la información y la comunicación (TICs). En la sección 2.2. se caracteriza a los Entornos Virtuales de Enseñanza Aprendizaje (EVEA) mediante la descripción de las particularidades de estos sistemas de acuerdo al tipo de información que pueden gestionar. En la sección 2.3 se introduce el concepto de OA, se analizan las diversas definiciones presentes en la literatura, se describen sus características, atributos y los problemas que aún se encuentran en debate.

2.1. Gestión y generación del conocimiento en la era de las tecnologías de la información y la comunicación

Para Davenport y Prusak (1999) el conocimiento es una mezcla de experiencia, valores, información y “saber hacer” que sirve como marco para la incorporación de nuevas experiencias e información, y es útil para la acción. Se origina y aplica en la mente de los conocedores.

Harris (1996) presenta una definición que ubica dato, información y conocimiento en distintos niveles clarificando la diferencia de dichos conceptos:

El nivel más bajo de los hechos conocidos son los datos. Los datos no tienen un significado intrínseco. Deben ser ordenados, agrupados, analizados e interpretados. Cuando los datos son procesados de esta manera, se convierten en información. La información tiene una esencia y un propósito. Cuando la información es utilizada y puesta en el contexto o marco de referencia de una persona, se transforma en conocimiento. El conocimiento es la combinación de información, contexto y experiencia. (p.1).

La noción de "sociedad del conocimiento" (knowledge society) emergió hacia finales de los años 90 y es empleada en ámbitos académicos como alternativa a la denominada “sociedad de la información”. Particularmente, UNESCO (2003), adoptó el término "sociedad del conocimiento", o "sociedades del saber", dentro de sus políticas

institucionales.

En cuanto a los aportes de las tecnologías de la información a la gestión del conocimiento, en un informe de UNESCO (2005) se mencionan los avances importantes en la accesibilidad y manejo del conocimiento a partir de Internet.

Un paso más adelante en esta “Nueva Sociedad “ es la generación de nuevos conocimientos a partir de la creación y el intercambio de insumos. Chan (2002) destaca que el valor agregado está dado por la significatividad de la información para las comunidades interesadas en determinados campos del saber.

La Figura 2.1, ilustra el ciclo información -conocimiento

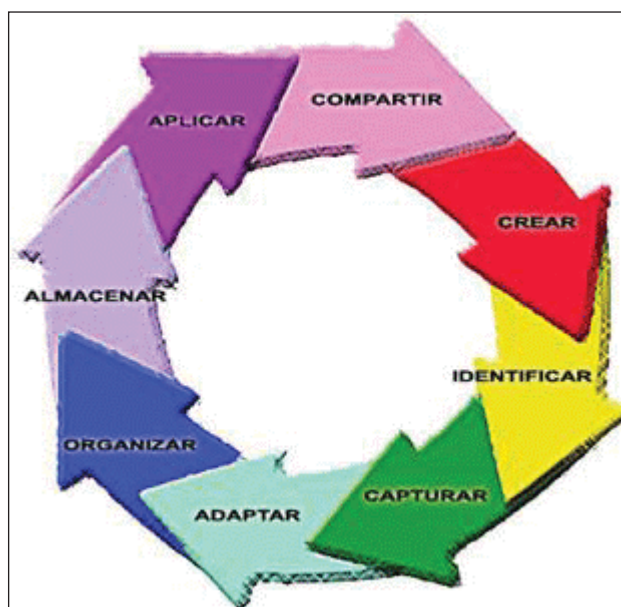


Figura 2.1 ciclo información –conocimiento. Tomado de <http://www.edukanda.es>

En este escenario, la gestión del conocimiento a través de las tecnologías de la información está sostenida por una parte, por el desarrollo de aplicaciones cada vez más eficaces, pero fundamentalmente, por el uso que las personas hacen de las mismas. El conocimiento, tal como lo señala Malhotra (2000), reside en el usuario y no en la recopilación de información o en el mero producto informático.

2.2. Entornos virtuales de enseñanza aprendizaje

Un Entorno virtual de enseñanza aprendizaje (EVEA) es una aplicación informática diseñada para facilitar el proceso de educativo a través de redes telemáticas. Forma parte del grupo de tecnologías basadas en Internet desarrolladas para educación. Un EVEA

permite la distribución de materiales educativos en formato digital (textos, imágenes, audio, videos, simulaciones, juegos, etc.) y facilita la comunicación mediante debates en tiempo real o diferido.

Para lograr estos objetivos posee herramientas de comunicación, de gestión de materiales y de gestión de los usuarios participantes (incluidos sistemas de seguimiento de las actividades)

Los entornos virtuales se utilizaron exclusivamente en un primer momento para la formación a distancia (e-learning), pero en los últimos años se han incorporado muy fuertemente a la educación presencial, generando así modalidades como el “extended learning” o “blended learning”.

Adell (2004) señala que en su camino evolutivo los EVEA centraron su atención en los siguientes aspectos:

- Integración con los sistemas de gestión de alumnos y docentes.
- Modularización de los sistemas monolíticos en arquitecturas de niveles y en componentes interoperables y estandarizados.
- Gestión separada de los contenidos y su creación, distribución e integración en unidades didácticas.
- Mayor preocupación por los aspectos pedagógicos del aprendizaje y la enseñanza on-line.
- Mayor flexibilidad y posibilidades de integración.

Así, la primera generación de entornos basados en la distribución de materiales y la evaluación mediante pruebas automáticas, dio paso a una segunda generación de entornos inspirados en los nuevos conocimientos sobre cómo se aprende usando recursos on-line. Esta segunda generación se orientó más hacia la comunicación didáctica, el diseño y monitorización del flujo de trabajo colectivo con los materiales y, en general, hacia la actividad didáctica o experiencia de aprendizaje. Al mismo tiempo, se trabajó sobre la estandarización de los contenidos donde surgieron estándares técnicos como SCORM que favorecían la reutilización y recombinación de los recursos didácticos.

Sin embargo, coincidiendo con Adell (2004), la adopción de un EVEA no garantiza per se la mejora de la calidad de la enseñanza. A la disponibilidad de los recursos tecnológicos, deben añadirse planes de formación del profesorado en estrategias didácticas y tecnologías de la información, medidas de apoyo a la innovación educativa y a la generalización de

buenas prácticas, estímulos a la producción y distribución de materiales formativos de calidad, planes para promover el aumento de la calidad y la cantidad de la comunicación entre profesores y estudiantes en la función tutorial, etc.

Desde esta perspectiva, poner a disposición de profesores y estudiantes un EVEA es sólo un requisito necesario pero no suficiente para la mejora de la calidad de la enseñanza universitaria. Es necesario enmarcar esta iniciativa en un proyecto de desarrollo que involucre todos los factores organizativos, materiales con la participación coordinada de todas las instancias de la universidad con responsabilidades en dichas áreas.

A fin de clarificar el lugar que ocupan los EVEA en los sistemas de administración de contenidos:

2. Sistemas centrados en el contenidos

- **CMS** (Content Management System o Course Management System): el objetivo de estos sistemas es la creación y gestión de información en línea (textos, imágenes, gráficos, videos, sonido, etc.). No poseen herramientas colaborativas.

b) Sistemas centrados en el aprendizaje

- **LMS** (Learning Management System) son aplicaciones Web que proveen las funciones administrativas y de seguimiento necesarias para posibilitar y controlar el acceso a los contenidos, implementar recursos de comunicaciones y llevar a cabo el seguimiento de quienes utilizan la herramienta. En general, los LMS facilitan la interacción entre los docentes y los estudiantes, aportan herramientas para la gestión de contenidos académicos y permiten el seguimiento y la evaluación. Es decir, facilitan la “simulación” del modelo real en el mundo virtual, por lo que también se les conoce como EVEA.
- **LCMS** (Learning Content Management Systems): integran las funcionalidades de los CMS y los LMS, que incorporan la gestión de contenidos para personalizar los recursos de cada estudiante. Los contenidos usualmente se almacenan como objetos descritos e identificables de forma única. En un LCMS se tienen contenedores o repositorios para almacenar los recursos, que pueden ser utilizados de manera independiente o directamente asociados a la creación de cursos dentro del mismo sistema. Es decir que el repositorio puede estar disponible para que los profesores

organicen los cursos o también pueden estar abiertos para que cualquier usuario recupere recursos no vinculados a ningún curso en particular.

En las Figuras 2.2, 2.3 y 2.4, se ilustran los distintos sistemas de administración de contenidos.

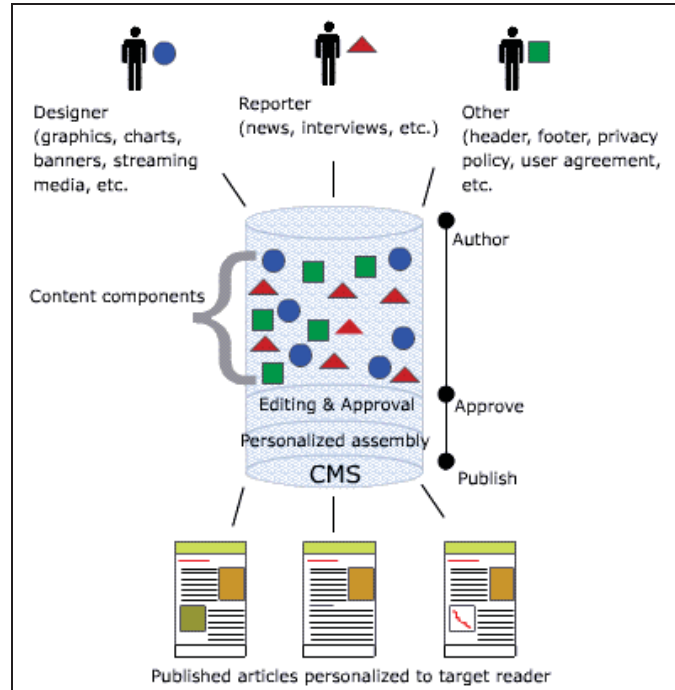


Figura 2.2 . CMS.Tomado de <http://www.elearningpost.com>

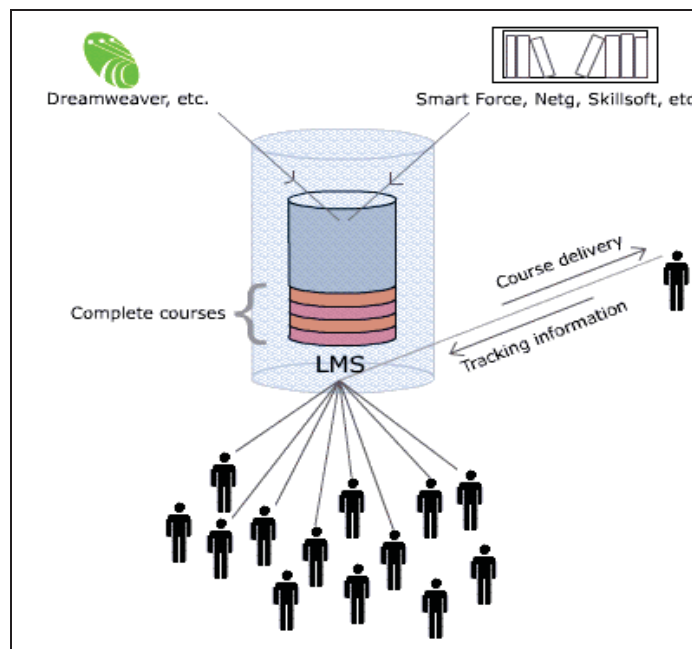


Figura 2.2 . LMS .Tomado de <http://www.elearningpost.com>

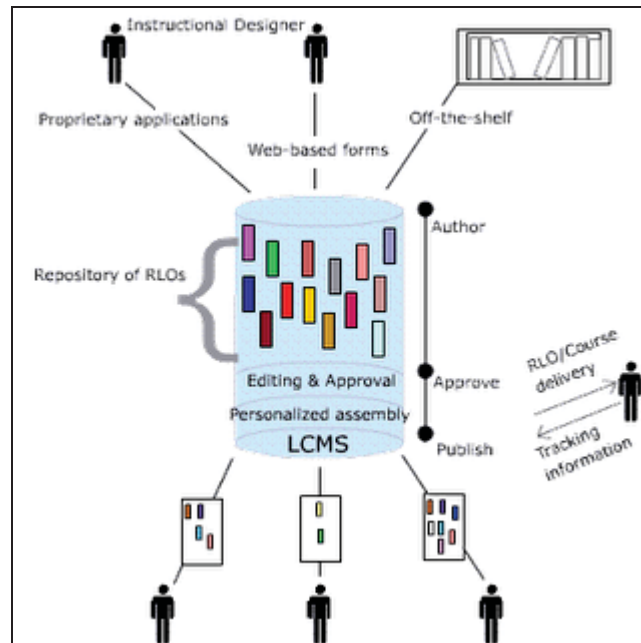


Figura 2.3 . LCMS.Tomado de <http://www.elearningpost.com>

2.3. Objetos de aprendizaje

La sociedad necesita proporcionar los medios y los recursos para dar respuesta a las nuevas formas de educación en las que están implícitas las TICs, el paradigma educativo y la socialización, entre otros.

En ese sentido, se considera además que en todo curso universitario se generan recursos para enseñar, y que éstos pueden digitalizarse de modo que se integren a un acervo compatible. Desde la gestión del conocimiento la finalidad es promover la creación y disposición de recursos para el aprendizaje. Estas acciones suponen una visión de acumulación del capital académico que se da por ejercicio de la enseñanza, y que puede trascender el uso individual en el aula al ponerlo a disposición en red.

Existe pues, un proceso de aprendizaje que requiere de materiales digitales educativos que lo propicien y donde el concepto OA tome relevancia.

Los OAs deben cumplir con ciertas características que permitan su reutilización en diversas situaciones sin problemas de compatibilidad con otras plataformas.

Varias organizaciones se encuentran desarrollando estándares y especificaciones e-learning. Sin embargo, la posibilidad de que los OAs puedan ser intercambiados no

significa que el contenido de éstos sea de calidad.

En ocasiones un OA puede cumplir con estándares para el e-learning y posibilitar así su interoperabilidad, pero su diseño no atiende a objetivos educativos específicos inmersos en un contexto de aprendizaje.

Con esta perspectiva, la calidad toma un sentido que va más allá de lo tecnológico y abarca otros aspectos como los pedagógicos.

2.3.1. Definición

Existen numerosas definiciones de los OA que han generado diversas controversias en el campo, podríamos citar algunas de ellas:

L'Allier (1998) define al OA como: “la mínima estructura independiente que contiene un objetivo, actividades de aprendizaje y una evaluación”.

En el año 2000 la IEEE presenta, dentro de LOM Working draf v4.1, su definición de objetos de aprendizaje, la cual trajo consigo una gran discusión (Wiley, 2000; Friesen, 2001): “un objeto de aprendizaje es definido como cualquier entidad -digital o no- que puede ser usada para aprender, enseñar o capacitar (training),” (Wiley, 2000; IEEE LTSC, 2002)

David Wiley (2000), retoma y reelabora la definición de la IEEE: “Cualquier recurso digital que pueda ser reutilizado para apoyar el aprendizaje”.

María Elena Chan (2002), miembro de la Comisión Académica del CUDI (Corporación Universitaria para el Desarrollo de Internet) afirma que:

Un objeto de aprendizaje es una entidad informativa digital que se corresponde (representa) con un objeto real, creada para la generación de conocimientos, habilidades, actitudes y valores, y que cobra sentido en función de las necesidades del sujeto que lo usa. (p.17).

García Aretio (2005) expresa que:

Los OAs son archivos digitales o elementos con cierto nivel de interactividad e independencia, que podrían utilizarse o ensamblarse, sin modificación previa, en diferentes situaciones de enseñanza-aprendizaje, sean éstas similares o desiguales entre sí y que deberían disponer de las indicaciones suficientes para su referencia e identificación. (p.3).

La reutilización depende de los metadatos (datos sobre datos) y éstos son la componente innovadora de los OAs. Sicilia y Alonso (2009) afirmaron que la esencia de los OAs son

los metadatos. Al eliminar este elemento podría hablarse genéricamente de “diseño de recursos digitales educativos”

Por otra parte en el portal de la comunidad “Aprendiendo con Objetos de Aprendizaje” se extiende la definición de L’Allier y definen un OA como:

La mínima estructura independiente que contiene un objetivo, una actividad de aprendizaje, un metadato y un mecanismo de evaluación, el cual puede ser desarrollado con tecnologías de infocomunicación de manera de posibilitar su reutilización, interoperabilidad, accesibilidad y duración en el tiempo. (APROA, 2005)

Aunque las definiciones anteriores comparten elementos importantes que permiten identificarlos, en este trabajo asumimos y extendemos la dada por APROA (2005) de la siguiente manera:

La mínima estructura independiente que contiene un objetivo, **un contenido**, una actividad de aprendizaje, un metadato y un mecanismo de evaluación, el cual puede ser desarrollado con tecnologías de infocomunicación de manera de posibilitar su reutilización, interoperabilidad, accesibilidad y duración en el tiempo.

Esta definición más extensa es entendida desde la concepción del OA como unidad independiente que engloba el objetivo de aprendizaje con los contenidos involucrados en él, las actividades diseñadas para alcanzar dicho objetivo y una evaluación de los saberes referidos al mismo.

En esa misma línea, Chan (2002) sostiene que el OA debe obtener y explicitar una intención de hacer aprender, por lo que el objeto debe contener instrucciones para el procesamiento de la información.

No obstante, los OAs pueden ser observados desde una reflexión epistemológica, la cual permite identificar diferentes posibilidades de relación del sujeto con el objeto, derivando en enfoques teóricos y metodológicos diversos para el diseño y su uso en educación.

A la luz de los nuevos avances tecnológicos, las herramientas de autor tendrán que funcionar en diferentes plataformas y comunicarse con otros medios utilizados para construir sistemas de aprendizaje. Contenido y cursos deben ser reutilizables, interoperables y de fácil manejo a distintos niveles de complejidad en todo el EVEA.

2.3.2. Atributos de los OA

A continuación se sistematizarán algunas características destacadas que encuadran a los OA en este trabajo. Además las mismas se convertirán en indicadores de la calidad de los OA, por lo que es importante recordar los siguientes parámetros:

- **Reusabilidad:** capacidad para ser usado en contextos y propósitos educativos diferentes y para adaptarse y combinarse dentro de nuevas secuencias formativas.
- **Interoperabilidad:** capacidad de integración en estructuras y sistemas (plataformas educativas) diferentes, posibilidad de ser exportados e importados, sin problemas de compatibilidad.
- **Accesibilidad:** facilidad para ser identificados, localizados a través de un etiquetado con diversos descriptores (metadatos).
- **Durabilidad:** vigencia de la información de los objetos con información sobre tiempos y recomendaciones de actualización.
- **Escalabilidad:** capacidad de integración en estructuras más complejas o extensas dentro del dominio de aprendizaje para el que fueron creados.
- **Generatividad:** capacidad para construir nuevos OA derivados de él. Posibilidad de actualización o modificación por medio de la colaboración de la comunidad de desarrollo de OA, aumentando sus potencialidades.
- **Autocontención conceptual:** Capacidad para autoexplicarse y posibilitar experiencias de aprendizaje íntegras.
- **Granularidad:** Término relacionado con el tamaño y el nivel de agregación. A menor tamaño mayor será la granularidad ya que permite integrarlo en diferentes escenarios de aprendizaje.

2.3.3. Componentes de un OA o estructura de un OA

De acuerdo a la definición conceptual que se considera en este trabajo y que fue presentada en la subsección 2.3.1, un OA está constituido por cuatro componentes internos: objetivos, contenidos, actividad de aprendizaje y evaluación.

El valor pedagógico del OA esta presente en su estructura interna cuyas características se detallan a continuación:

- **Objetivos:** Expresan las metas de aprendizaje propuestas.
- **Contenidos:** Conjunto de definiciones, explicaciones, artículos, videos, entrevistas, lecturas, opiniones, explicaciones, escenarios, casos, proyectos, fuentes o referencias entre otros que guían al estudiante para alcanzar los objetivos.
- **Actividad de aprendizaje:** tareas que debe llevar a cabo el estudiante que contribuyen a alcanzar los objetivos propuestos.
- **Evaluación :** actividades emprendidas por los docentes y estudiantes con el propósito de proveer información para ser usada como feedback útil para modificar y mejorar las actividades de enseñanza y aprendizaje en las que están implicados (Black & Dylan ,1998)

La estructura de información externa del objeto se refiere a los Metadatos que es información acerca del contenido del objeto (su contenido, tema, uso, autor, etc.). Esta información debe estar estructurada y esto debe guardar una distribución y un orden.

Se definen los metadatos como “datos estructurados acerca de datos”. La IEEE (2002), los define como “información sobre un objeto, sea éste físico o digital”. La Australian Government Information Managment Office (AGIMO, 2004) define metadato como “información estructurada que es creada específicamente para describir un recurso”.

En síntesis, el OA podría ilustrarse con la Figura 2.4, pensado como una unidad o un todo.

Figura 2.4. Estructura de un Objeto de Aprendizaje

2.3.4. Ventajas y desventajas

Actualmente existe una tendencia que considera OA a recursos digitales que brindan información. De acuerdo a la concepción asumida en esta tesis, se destaca y diferencia los OA de los objetos informativos, coincidiendo con la clasificación propuesta por CISCO Systems (1999) con relación a los RLO (Reusable Learning Objects) y los RIO (Reusable Informative Objects).

Lo anterior hace evidente la importancia del referente conceptual sobre el cual se diseñen, desarrollen y cataloguen los OAs.

Por otra parte, el concepto de objeto didáctico – del inglés learning object (LO) – ha evolucionado en los últimos años como la noción subyacente (tácita o explícita) alrededor de la cual se estructura un conjunto de tecnologías y estándares que se presupone conducen a una “industria del aprendizaje” más eficiente y evolucionada (Martinez, 2001), o si se prefiere, a un nuevo panorama en la educación basada o apoyada en la Web.

Como se mencionó en la subsección 2.3.1 las definiciones existentes contienen un elemento en común: la reutilización. Coincidiendo con Sicilia (2005), esta reutilización repetida es de hecho la fuente de los potenciales beneficios del concepto.

2.3.5. Repositorios de OA

Una de los atributos de los OA es su reutilización, los sistemas de repositorios son la infraestructura clave para el desarrollo, almacenamiento, administración, localización y recuperación de todo tipo de contenido digital (ADL, 2002).

Como se ha buscado una solución particular que facilite la recopilación, el acceso y el

compartir recursos educativos, surge también la posibilidad de crear un sistema de almacenamiento de contenidos que se integre y comunique fácilmente con los otros sistemas que operan en los ambientes de e-learning (McLean & Lynch, 2003).

El programa Canaire (2002) define Repositorio de Objetos de aprendizaje (ROA) como “un catálogo electrónico/digital que facilita las búsquedas en Internet de objetos digitales para el aprendizaje” (p.9).

Para Downes (2004): “los repositorios de objetos de aprendizaje son bases de datos con búsquedas que alojan recursos digitales y/o metadatos que pueden ser utilizados para el aprendizaje mediado” (p.250).

El JORUM + project (2004) adopta la siguiente definición:

Un ROA es una colección de OA que tienen información (metadatos) detallada que es accesible vía Internet. Además de alojar los OA los ROA pueden almacenar las ubicaciones de aquellos objetos almacenados en otros sitios, tanto en línea como en ubicaciones locales (p.16).

Se puede decir entonces, que los ROA (Figura 2.5) son un tipo de bibliotecas digitales especializadas en recursos educativos que utilizan los estándares de metadatos que han desarrollado los organismos encargados de la estandarización de e-learning, preparadas tecnológicamente para interoperar con otros repositorios y con otras aplicaciones de los entornos e-learning..

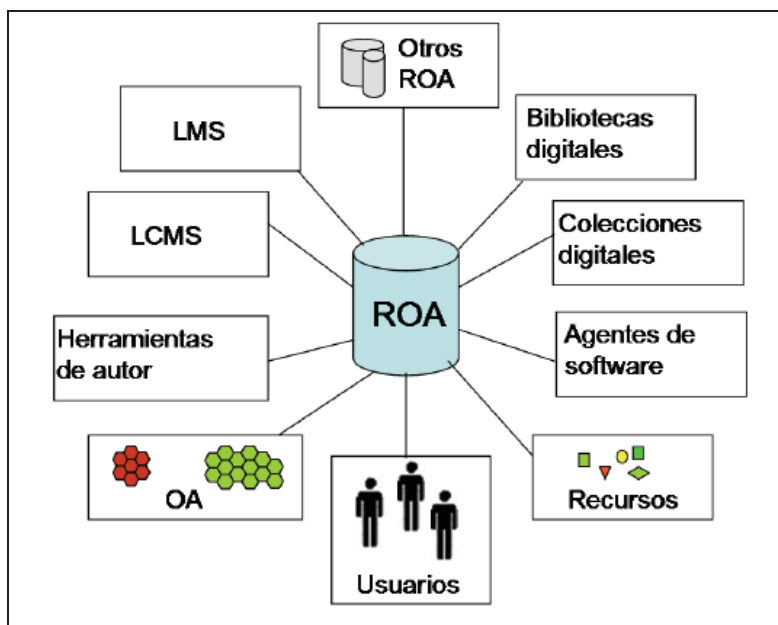


Figura 2.5. Repositorio de OA (ROA). Fuente: López (2005).

Entre las iniciativas más conocidas podemos citar: MERLOT (Multimedia Educational Resource for Learning and Online Teaching), (<http://www.merlot.org/>) , CAREO (Campus Alberta Repository of Educational Objects) (<http://careo.ucalgary.ca>), CLOE (Co-operative Learning Object Exchange) (<http://cloe.on.ca/>) , SMETE (Science, Mathematics, Engineering and Technology Education) (<http://www.smete.org/>), DLNET (Digital Library Network for Engineering and Technology) (<http://www.dlnet.vt.edu/>), GEM (Gateway to Educational Materials) (<http://www.thegateway.org/>) , CeLeBraTe (Context eLearning with Broadband Technologies) http://celebrate.eun.org/eun.org2/eun/en/index_celebrate.cfm/

CAPITULO 3. ELEMENTOS DE ANÁLISIS EN LOS OBJETOS DE APRENDIZAJE

En este capítulo, se presentan en la sección 3.1 las herramientas y conocimientos sobre estándares y especificaciones de e-learning requeridos para la creación de un Objeto de Aprendizaje (OA). Luego en la sección 3.2 se describen las concepciones referidas a la utilización y creación de un OA desde las perspectivas pedagógica y tecnológica. Finalmente en la sección 3.3 se detallan los pasos para la creación de un OA bajo los estándares y especificaciones de e-learning.

3.1. Especificaciones y estándares en e-learning

La reutilización de contenidos entre sistemas distintos requiere que los objetos de aprendizaje estén estandarizados, por lo que muchas organizaciones dedican sus esfuerzos al desarrollo de estándares (LOM, 2002), especificaciones (IMS Global, www.imsglobal.org) y modelos de referencia (SCORM, 2002) que faciliten la interoperabilidad y la reutilización de objetos de aprendizaje.

Según el diccionario de la Real Academia de la Lengua (RAE), un estándar es lo “que sirve como tipo, modelo, norma, patrón o referencia”. Desde el punto de vista técnico la estandarización es el proceso por el cuál se establecen unas normas comúnmente aceptadas que permiten la cooperación de diferentes empresas o instituciones sin menoscabar su posibilidad de competir. Un estándar proporciona ventajas no sólo a las empresas, si no también al usuario, ya que así no ve limitada su capacidad de elección a un determinado proveedor, si no a todos aquellos que cumplen un estándar determinado y que, por tanto, crean productos que son compatibles.

Sin embargo, es importante destacar que entre los estándares es posible distinguir tres tipos: de facto, de jure y propietarios.

- De jure: es definido por grupos u organizaciones oficiales tales como la ITU, ISO, ANSI, IEEE, es decir, provienen de una organización acreditada que certifica su especificación.

- De facto: tienen una alta penetración y aceptación en el mercado, pero aún no son oficiales. Se generan cuando la especificación se adopta por una comunidad sin haber pasado por un proceso formal de acreditación.
- Propietario: son propiedad absoluta de una corporación u entidad y su uso todavía no logra una alta penetración en el mercado.

Los pasos para la creación de estándares para la comunidad de e-learning se muestran en la Figura 3.1 y se explican a continuación.

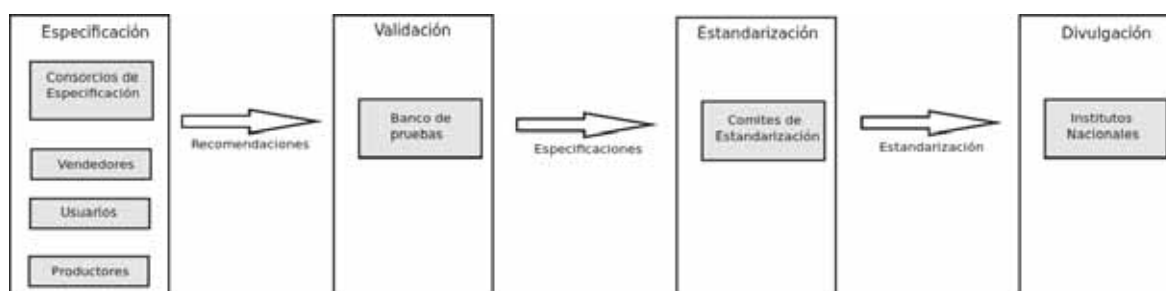


Figura 3.1. Pasos para establecer estándares de e-learning. Adaptado de Varlamis & Apostolakis(2006)

a) Especificación:

Una especificación se define como “una descripción documentada del comportamiento, las características y la forma de comportarse de los objetos ya sean virtuales o reales” (Foix & Zavando, 2002). Es una descripción detallada de las características de los objetos, especificadas en un documento técnico, que describe tanto sus componentes (parte estática) como el comportamiento (parte dinámica). El seguimiento de las especificaciones por los desarrolladores, permite una estructura única y común, para lograr la interoperabilidad de la información en diversas plataformas, permitiendo con ello incorporar contenidos que provengan de diferentes proveedores.

El análisis de las necesidades y la especificación de los requisitos están a cargo actualmente por consorcios tales como:

- IMS Global Learning Consortium (www.imsglobal.org)
- CEN/ISS Workshop on Learning Technology
(www.cenorm.be/iss/Workshop/lt/)
- Customized Learning Experiences Online – CLEO, which involves IBM, Microsoft, Cisco, NetG, and click2learn

(<http://www.lsal.cmu.edu/lsal/expertise/projects/cleo/>)

- Aviation Industry CBT Committee (www.aicc.org/)
- HR-XML Consortium (www.hr-xml.org/).

b) Validación

Con base en el mercado, el trabajo del paso anterior y las especificaciones sugeridas, los vendedores desarrollan nuevos productos incorporando los nuevos requisitos. Se inician pruebas piloto de eficacia y utilidad de los nuevos productos y se establece un banco de pruebas para la validación de conformidad con las especificaciones. La salida de la etapa de validación consta de varios modelos de referencias que muestran cómo las especificaciones y los diferentes estándares pueden apoyar un completo entorno e-learning (Varlamis & Apostolakis, 2006).

Las organizaciones que crean bancos de prueba y modelos de referencia son:

- Advanced Distributed Learning initiative - ADL/SCORM (www.adlnet.org)
- Advanced Learning Infrastructure Consortium – ALIC (www.cen-ltso.net)
- Education Network Australia – Edna (www.edna.edu.au)

c) Estandarización

Esta etapa consiste en lo siguiente:

- Perfeccionamiento de especificaciones (concretas y claras para los usuarios).
- Consolidación en una sola salida.
- Clarificación de requisitos de conformidad (evitar incompatibilidades).
- Reconocimiento oficial de las normas por las organizaciones internacionales y comités.

Los organismos acreditados de creación de estándares para e-learning son:

- IEEE Learning Technology Standards Committee (ieeeltsc.org)
- ISO/IEC Joint Technology Committee Subcommittee on Standards for Learning

d) Divulgación

Para el establecimiento de normas es necesaria la difusión de las mismas a toda la comunidad e-learning. Frecuentemente los términos estándar y especificación se utilizan indistintamente, no obstante, es importante puntualizar su diferencia. Si una tecnología, formato o método ha sido ratificado por algún organismo oficial de estandarización, se

trata de un estándar. Pero si una tecnología, formato o método propuesto no ha sido aprobado por algún organismo oficial de estandarización, se trata de una especificación. Aunque, en algunos casos, una especificación puede considerarse un estándar de facto si su uso es extendido y entretanto se ratifica como estándar (Varlamis & Apotolakis, 2006).

Una especificación es un trabajo en evolución y progreso, en qué varias personas y organizaciones pueden ofrecer sus recomendaciones, y un estándar acreditado, que está idealmente basado en implementaciones reales y ofrece criterios muy claros y sin ambigüedades para la implementación y conformidad.

A continuación se describirán algunas de las iniciativas más importantes y sus estándares e-learning más significativos.

Pero antes de comenzar que describamos con más detalle a los Metadatos. Son datos estructurados que proporcionan un resumen sobre cualquier recurso de información, impresa o digital y facilita la ubicación, identificación o descubrimiento de ese recurso.

La Figura 5.2 muestra como pueden clasificarse los Metadatos.

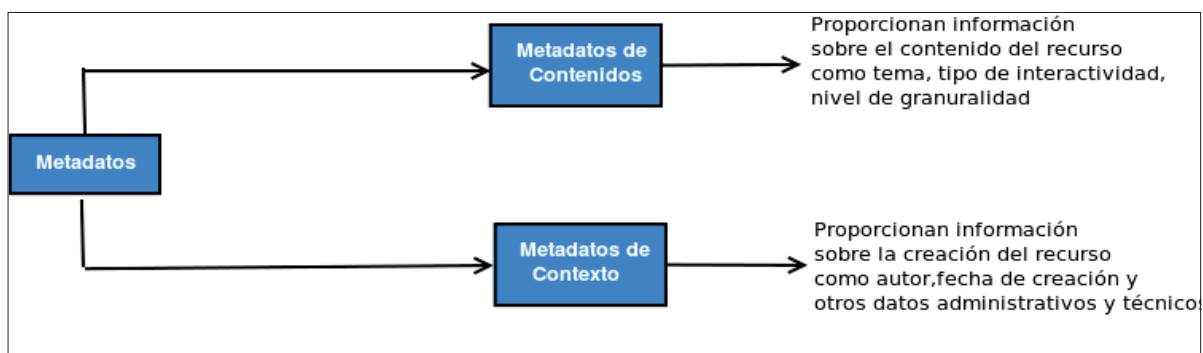


Figura 3.2. Clasificación de Metadatos

3.1.1. AICC (Aviation Industry CBT Comitee)

El primer grupo de estándares es el AICC (<http://www.aicc.org>), creado en 1989 por el Comité CBT (Computer Based-Training) de la Industria Aeronáutica de Estados Unidos con el objetivo de contar con un código que permitiese intercambiar módulos elaborados con diferentes autores o herramientas para crear cursos interactivos de autoinstrucción que se ejecutaban en máquinas individuales o en redes de área local o LAN.

Las especificaciones del AICC cubren nueve áreas principales, que van desde OA que denomina Unidad Asignable (UA) hasta los Learning Management System (LMS).

3.1.2. IEEE learning object meta-data / IMS learning resource metadata specification (LOM)

El LTSC (Learning Technology Standards Committee, <http://ltsc.ieee.org/wg12/>) del IEEE desarrolló el estándar de metadatos para los OA LOM (Learning Object Meta-data). Este estándar se encarga de definir un esquema conceptual de datos (XML), que permite describir OA. Se facilita entonces la búsqueda, evaluación, adquisición y utilización de los objetos educativos (Anido Riffon y Rodríguez Artacho, 2002).

El estándar LOM de IEEE es un estándar de jure que goza de mayor aceptación (LOM 2002).

El objetivo de LOM es la creación de descripciones estructuradas de recursos educativos. Su modelo de datos especifica qué aspectos de un OA deberían ser descritos y qué vocabularios se pueden utilizar en dicha descripción.

Esta es una descripción jerárquica con nueve apartados principales que agrupan el resto de campos (Figura 3.3). A continuación describimos cada una de estas categorías (Anido Riffon y Rodríguez Artacho, 2002):

1. General: información general que describe un objeto educativo de manera global.
2. Ciclo de Vida: características relacionadas con la historia y el estado actual del objeto educativo y aquellas que le han afectado durante su evolución.
3. Meta-Metadatos : información sobre la propia instancia de Metadatos, (en lugar del objeto educativo descrito por la instancia de Metadatos) .
4. Técnica: agrupa los requerimientos y características técnicas del objeto educativo.
5. Educacional: características educativas y pedagógicas del objeto.
6. Derechos: derechos de propiedad intelectual y las condiciones para el uso del objeto educativo.
7. Relación: características que definen la relación entre este objeto educativo y otros objetos educativos relacionados.
8. Anotación: comentarios sobre el uso educativo del objeto e información sobre cuándo y por quién fueron creados dichos comentarios.
9. Clasificación: descripción en relación a un determinado sistema de clasificación.

Conjuntamente, estas categorías forman el Esquema Base LOM v1.0. La última categoría, Clasificación, permite clasificar un OA de acuerdo con una estructura de clasificación arbitraria. Como puede hacerse referencia a cualquier sistema de clasificación, esta categoría se proporciona como un mecanismo de extensión (Anido Riffon y Rodríguez

Artacho, 2002).

Los elementos que componen la categoría educacional son los que comúnmente se consideran para diseñar una unidad de aprendizaje, la descripción de ellas permite a los docentes conocer el recurso para saber si es pertinente a sus requisitos, aspecto clave para la reutilización de los OAs, por tanto, dentro del ámbito educativo se podrían mencionar las siguientes ventajas:

- Descripción de recursos educativos: la categoría educacional permite incluir datos que son importantes a considerar para la conformación de unidades de aprendizaje; es decir, aspectos pedagógicos que no eran considerados por el modelo de los OAs y que son importantes para el diseño instruccional de unidades didácticas.
- Facilita el ensamblaje de los OAs desde un punto de vista instructivo: de acuerdo a las características pedagógicas de cada OA, es posible conocer qué otros OAs son necesarios para el ensamblaje, de manera que tenga sentido para una situación concreta de aprendizaje.
- Permite obtener información de valor educativo: gracias a la posibilidad de obtener OAs de diversas fuentes, la información extraída es mucho más enriquecedora.
- Reutilización de los recursos educativos: La descripción que llevan los datos a través de los Metadatos, permite reutilizar los recursos para adaptarlos a nuevas situaciones de aprendizaje.

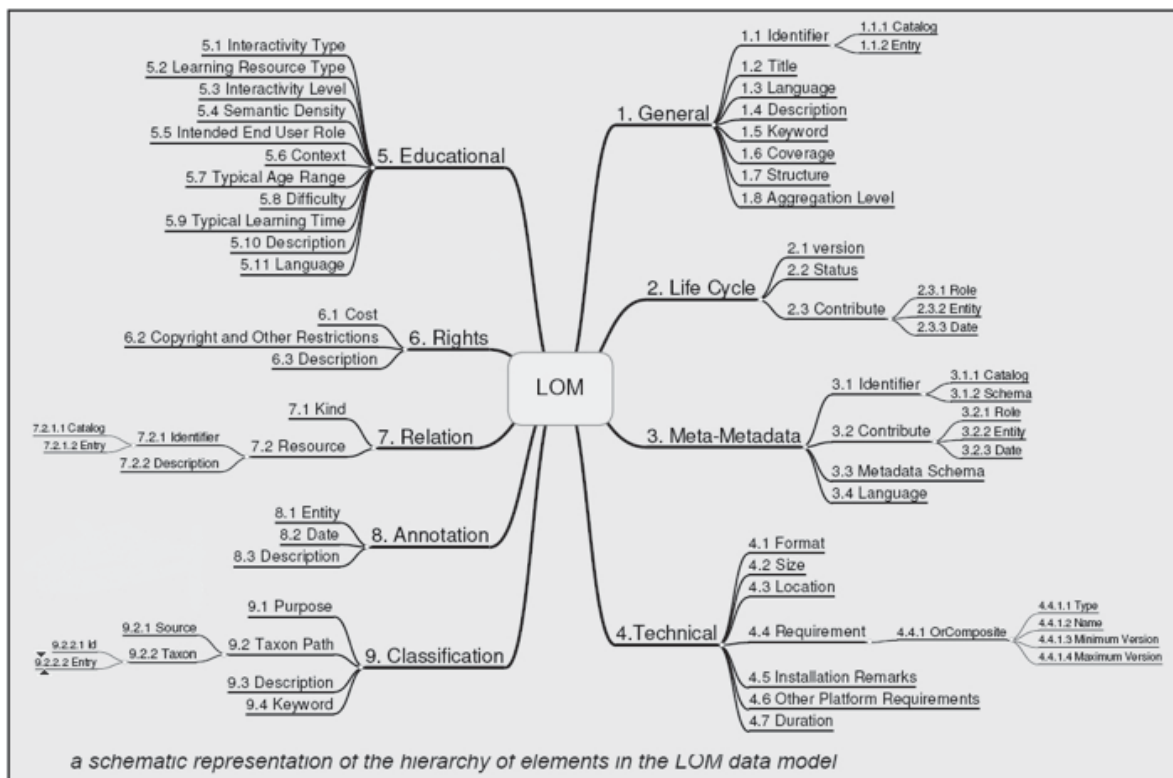


Figure 3.3: Representación de la jerárquica de elementos en Modelo de datos LOM. Fuente: Barker (2005).

3.1.3 Dublin Core

Esta especificación nació con el objetivo de describir recursos de carácter genérico en la Web y también ha sido adoptado por la comunidad educativa con el fin de adjuntar información complementaria a los recursos educativos.

Aunque en la actualidad LOM y sus ramificaciones constituyen el cuerpo de metadatos para materiales educativos con mayor reconocimiento y dedicación de esfuerzo en la comunidad de e-learning internacional, también existen otras propuestas de metadatos. De éstas, quizá la más conocida sea Dublin Core (<http://dublincore.org/>).

Frente a los más de 70 campos de LOM, los 15 metadatos básicos de Dublín Core para un recurso educativos son: título, autor, tema o palabras clave, descripción, editor, otros colaboradores, fecha, tipo de recurso, formato, identificador, fuente, idioma, relación con otros recursos, cobertura, y derechos.

La principal ventaja de Dublin Core frente a LOM es su mayor sencillez. No obstante, dicha sencillez se deriva, también, de su menor nivel de detalle. Efectivamente, LOM es mucho más exhaustivo que Dublin Core (por ejemplo, Dublin Core no introduce Metadatos específicos para caracterizar los propósitos educativos de los contenidos).

No obstante, en contextos donde no se demande excesiva flexibilidad semántica a la hora de catalogar los materiales, Dublin Core puede resultar una opción interesante.

Dado que Dublin Core propugna una organización más plana de los metadatos, la codificación es también más sencilla.

3.1.4. IMS learning resource meta-data specification

IMS Global Learning Consortium (<http://www.imsglobal.org>), usualmente conocido como Instructional Management System (IMS) es el principal promotor y desarrollador de especificaciones abiertas orientadas al e-learning (actualmente tiene 16 especificaciones). Su objetivo es que a partir de estas especificaciones se consiga la interoperabilidad de aplicaciones y servicios en e-learning para que los autores de contenidos y de entornos puedan trabajar conjuntamente. No obstante, ningún estándar puede cubrir todas y cada una de las necesidades que la gran diversidad de aplicaciones y contextos educativos exigen. Se considera que estas especificaciones son un marco general de interoperabilidad que proporcionan un margen de adaptación a las necesidades concretas de cada dominio o aplicación (lo que se denominan perfiles de aplicación).

Como se identificó al comienzo de este capítulo, IMS desarrolla especificaciones pero no aprueba estándares. A continuación se describen las especificaciones más importantes de IMS.

3.1.4.1. IMS Content Packaging (Empaquetado de contenidos IMS-CP)

La especificación IMS-CP v1.2 describe las estructuras de datos que pueden ser utilizadas para intercambiar datos entre sistemas que desean importar, exportar, agregar y desglosar paquetes de contenido. Esta especificación permite exportar contenido desde un LMS o repositorio digital e importarlo en otro (IMS-CP, 2007).

Ofrece una forma de empaquetar (en un archivo comprimido tipo zip) los contenidos educativos tales como cursos individuales, conjuntos de cursos, o cualquier tipo de recurso necesario en el proceso educativo (por ejemplo, evaluaciones o exámenes).

La estructura de un paquete de contenidos IMS v1.2 (Figura 3.4), está conformada por:

a) Un documento fundamental: el Manifiesto. Se describe la estructura de los contenidos incluidos en el paquete en dos niveles:

- Se describen cada uno de los Recursos del paquete. En una primera aproximación se puede hacer una relación casi directa entre un Recurso y un fichero con contenidos visualizables (e.g. un OA) como pueden ser ficheros

HTML, animaciones en Flash, etc. En cada Recurso se puede incluir información sobre los ficheros que componen dicho Recurso, el tipo de los mismos (que puede ser uno de los tipos ya definidos por el estándar o una extensión de los propuestos) y, opcionalmente, metadatos con información adicional sobre dicho Recurso.

- Se describe la organización de dichos Recursos, es decir, cómo se estructura el contenido del paquete. Esto se implementa mediante las Organizaciones. Una Organización es una vista (o recorrido) de una posible ordenación jerárquica (actualmente en forma de árbol) de los Recursos de un paquete. El estándar permite que un Manifiesto contenga distintas Organizaciones sobre los Recursos del paquete, dando así lugar a distintas vistas o cursos a partir de los mismos contenidos. El elemento básico de estructuración que se usa al definir las organizaciones son los ítems. A cada ítem se le puede asociar un Recurso, de modo que el árbol de Ítems es, efectivamente, una estructuración de los Recursos del paquete.

En síntesis, el Manifiesto es un fichero XML que describe y organiza los contenidos de un paquete, añadiendo información adicional en forma de metadatos que pueden ser procesados y aprovechados en tareas de catalogación de contenidos. Los metadatos proporcionan descripciones, propiedades e información sobre los Recursos (sean OA, partes de un curso o cursos completos).

Finalmente, para la distribución e intercambio efectivo de los Recursos se crea es un Archivo de Intercambio de Paquetes (Package Interchange File, o simplemente PIF). El PIF es un archivo que alberga en su interior el manifiesto y los recursos que se referencian en dicho manifiesto. En definitiva es un paquete comprimido con formato de intercambio zip.

La funcionalidad de exportación a PIF o de importación de un PIF se encuentra en muchos de los LMS tanto comerciales (e.g. WebCT) como de software libre (e.g.: Moodle, Dokeos, Claroline).

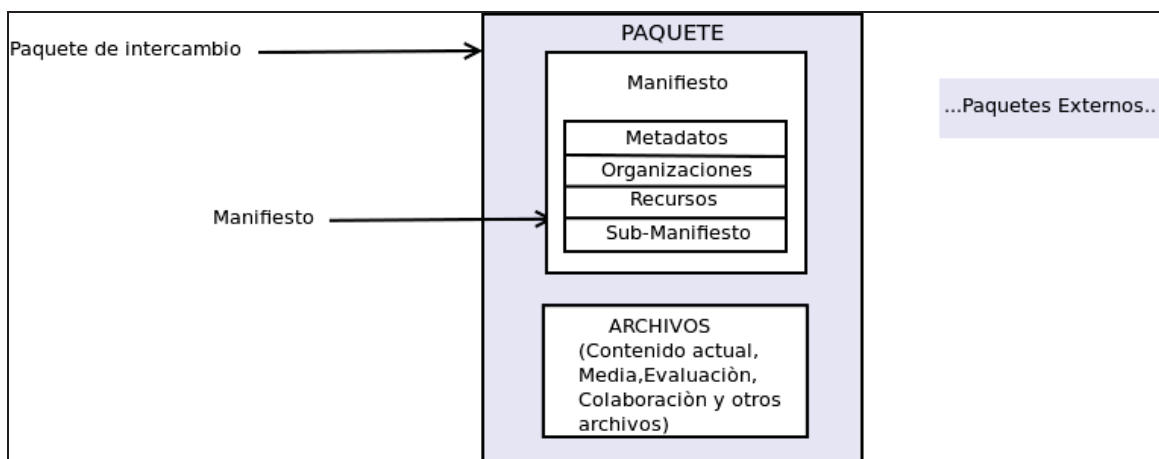


Figura 3.4. IMS Content Packaging. Adaptado de IMS-CP (2007).

3.1.4.2. IMS Learning Design (Diseño de aprendizaje IMS-LD)

Esta especificación ha sido el resultado de la integración dentro de IMS de la especificación Educational Modeling Language (lenguaje de modelado educacional) desarrollada inicialmente en la Universidad Abierta de Holanda. Se ocupa de describir y codificar el diseño pedagógico, es decir las metodologías educativas implícitas en un proceso de enseñanza de forma que sean procesables por un LMS.

El resultado final ha sido el desarrollo de una nueva especificación adaptada en aquellas partes en las que existía un solapamiento con el resto de especificaciones propuestas por IMS, junto a una adaptación de la propia especialización para poder dividirla en varios niveles, al estilo del modelo de capas en una arquitectura de software, para hacerla más comprensible.

Una de las adaptaciones más relevantes llevada a cabo ha sido la adopción de la especificación IMS-CP como formato y medio de transmisión e intercambio entre distintos LMSs y herramientas.

En el subsección anterior sobre la especificación IMS-CP se mostró la estructura de un paquete, en esta ocasión, la misma estructura muestra la incorporación de Learning Design para formar una unidad de aprendizaje, esto es posible al incluir el elemento “learning-design” como un tipo de organización dentro del componente <organización> de IMS-CP (IMS-LD, 2003) . La Figura 3.5 ilustra la integración de Learning Design dentro de una estructura de Content Packaging.

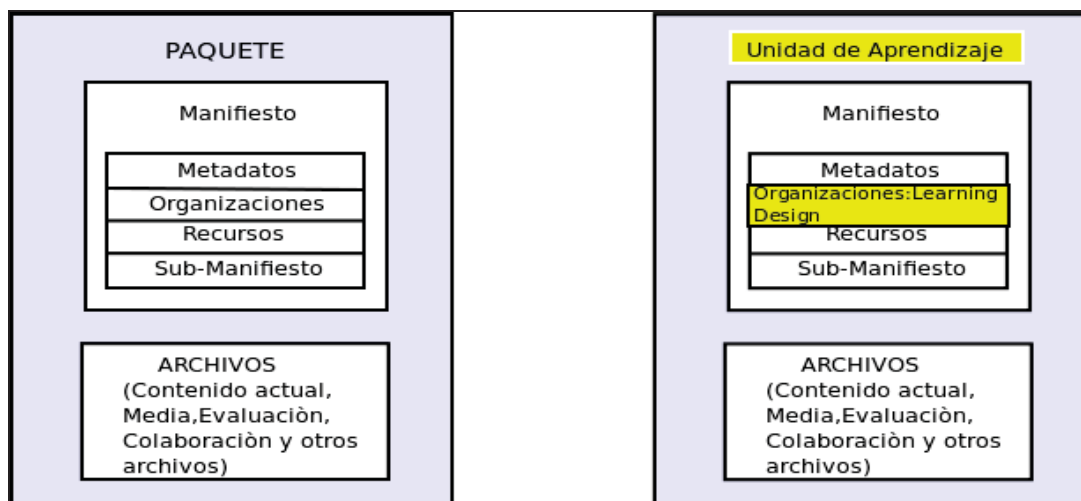


Figura 3.5. Integración de IMS-LD dentro de IMS-CP. Adaptado de IMS-LD (2003).

Para disminuir la complejidad que supone su implementación, IMS-LD está dividida en tres niveles:

- Nivel A: Contiene el vocabulario básico que soporta la diversidad pedagógica. Cuando se crea una Unidad de Aprendizaje (UDA), se debe especificar un modelo estático y un modelo dinámico.
- Nivel B: Añade propiedades y condiciones al nivel anterior. Las propiedades definen un modelo de datos para el alumno y las condiciones se utilizan para personalizar las UDAs en base a los conocimientos previos de los alumnos y a su interacción con dichas UDAs.
- Nivel C: agrega al Nivel B la posibilidad de realizar notificaciones.

En la Figura 3.6 puede observarse la estructura de alto nivel de una Uda expresada en IMS-LD.

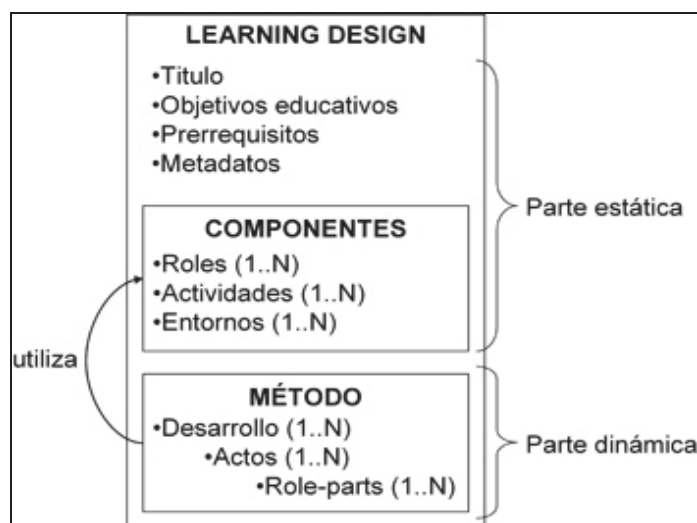


Figura 3.6. Estructura de alto nivel de una UDA en IMS-LD .Fuente: Fernández Manjón et.al. (2009).

El modelo estático de IMS-LD permite definir qué es lo que se va llevar a cabo en la Uda, qué tipos de usuarios (e.g. profesores, alumnos, etc.) participan en la UOL y con qué recursos se llevarán a cabo las actividades (e.g. páginas HTML, PDF, etc.).

El modelo dinámico permite definir qué tipo de usuario (rol) llevará a cabo una actividad concreta y también permite definir la sincronización y las dependencias que existirán entre las distintas actividades que componen la Uda.

Desde que se aprobó IMS-LD se han ido desarrollando un cierto número de herramientas: editores, reproductores y motores de ejecución.

Podemos mencionar:

- Ejemplos de herramientas de código abierto: Coppercore (<http://coppercore.sourceforge.net/>), RELOAD (<http://www.reload.ac.uk>), Eduplone (<http://eduplone.net/>) y MOT+ (<http://www.Licef.teluq.quebec.ca/francais/real/demot.htm>).
- Implementaciones comerciales : Perot Systems Edubox (<http://elearning.surf.nl/edubox/home>), LAMS, eLive LD (http://www.elive-ld.com/content/index_eng.html suite , and GTK Press (http://www.gtkpress.com/default.aspx?pid=m_1013).

La mayoría de estos editores requieren que los autores tengan un buen conocimiento de la especificación, otros editores de alto nivel que no tienen esta exigencia están en la actualidad en proceso de desarrollo.

El sistema LAMS dispone de una interfaz drag and drop, muy fácil de usar que permite combinar diferentes actividades de aprendizaje dentro de la estructura de una lección. Actualmente, cumple con la especificación IMS-LD nivel A.

Moodle y IMS-LD están trabajando desde el año 2005 la posibilidad de integración (Berggren et.al, 2005; Figueroa Martínez, Lluvia Morales, y Castillo, 2008). En este sentido, Moodle y la Open University de Holanda (OUNL) mantienen un grupo de trabajo con el propósito de: a) Que un curso generado por Moodle pueda ser exportado a la notación de IMS-LD; b) que una UDA generada bajo parámetros de IMS-LD pueda ser importada como curso dentro de Moodle; c) que una UDA IMS-LD pueda ser ejecutada dentro de Moodle.

3.1.4.3. IMS Question & Test Interoperability Specification (Interoperabilidad test y cuestiones IMS-QTI)

La especificación IMS-QTI describe un modelo de datos para la representación de preguntas individuales (assessment Item) y evaluaciones completas o test (assessment) con sus correspondientes informes de resultados. Así, esta especificación permitirá el intercambio de preguntas, test y resultados entre LMS.

QTI trata de ser pedagógicamente neutral y proporciona un gran conjunto de preguntas que habitualmente se utilizan en las evaluaciones tales como elección verdadero/falso, elección múltiple con respuesta única, elección múltiple con varias respuestas válidas, rellenar campos en blanco, ordenar objetos, relacionar objetos, etc. Además permite definir nuevos tipos de preguntas si fuera necesario (IMS-QTI, 2008).

En la especificación de QTI se ha producido un gran cambio entre la versión anterior la 1.2 y la versión final 2.1 ya que en esta última se ha tratado de sistematizar más los exámenes evitando muchas de las dificultades de interpretación y tecnológicas que existían en versión anterior. Por ejemplo, la versión 2.0 se ha centrado en simplificar el aspecto más conflictivo en las especificaciones anteriores, que es el concepto de ítem o pregunta individual, dejando inalterados aspectos como la agrupación de preguntas en secciones o exámenes que estaban claramente definidas en la versión 1.2 (IMS-QTI, 2008)

Un examen para IMS QTI es simplemente un grupo de preguntas. Durante el proceso de creación podemos estructurar el examen en distintas partes (testPart). Por ejemplo, si hemos impartido un módulo de un curso en el cual se discuten distintos temas, podemos crear un examen en el que se crean distintas partes por cada uno de esos temas. Asimismo,

estas partes pueden dividirse en distintas secciones (sections) que representan diferentes secciones dentro de una parte del examen. Tanto las partes como las secciones de un examen pueden contener materiales que serán presentados a los estudiantes durante la realización de dicha parte o sección (Figura 3.7) (Fernández Manjón et.al. 2007).

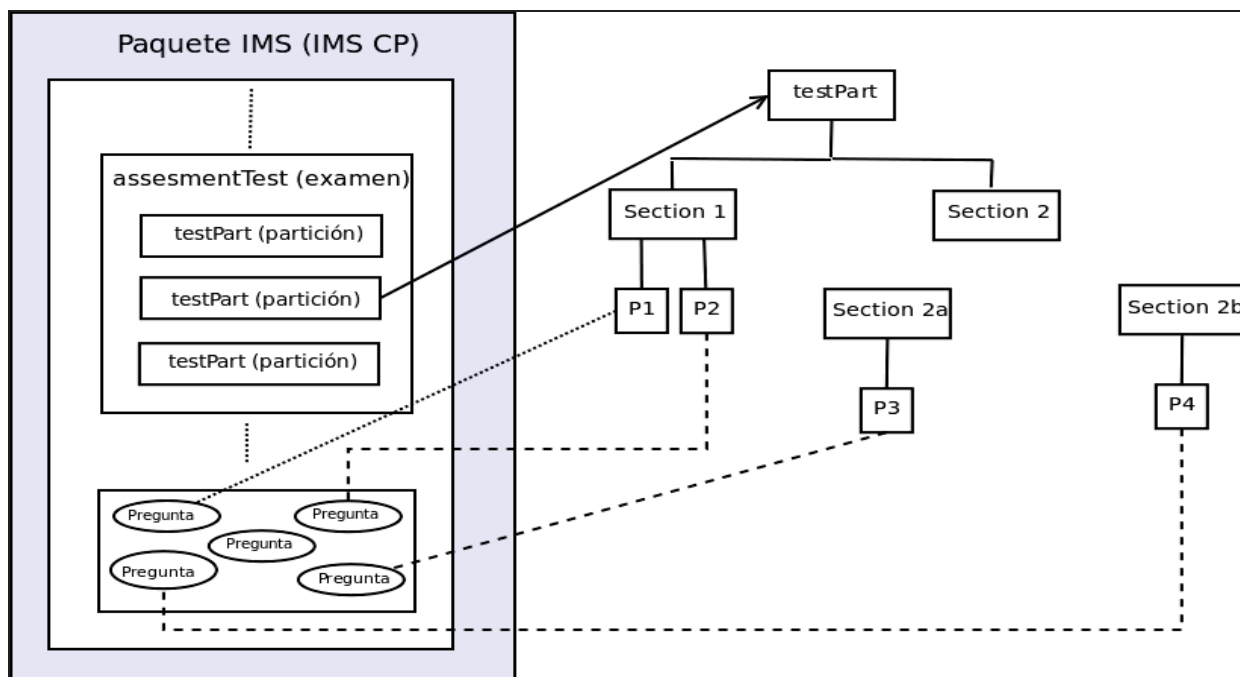


Figura 3.7. Estructura de un examen dentro de un paquete IMS. Adaptado de: Fernández Manjón et.al. (2007).

3.1.4.4. IMS Common Cartridge specification (Cartucho común IMS-CC)

Esta especificación define un formato abierto para la distribución de contenido enriquecido basado en web (html, xml, enlaces Web, ficheros multimedia, ficheros de aplicaciones que permita el intercambio de paquetes entre los LMS que soporten dicha especificación (Figura 3.8). Para ello define un perfil de las siguientes especificaciones:

- IEEE LOM, Dublin Core
- Paquete de contenidos IMS o IMS-CP v1.2
- Interoperabilidad test y cuestiones IMS o IMS-QTI
- Servicios web de autorización IMS v1.0

El objetivo de esta especificación es definir una estructura que permita el intercambio de paquetes con información relativa a cualquiera de los implicados en el sistema de enseñanza.

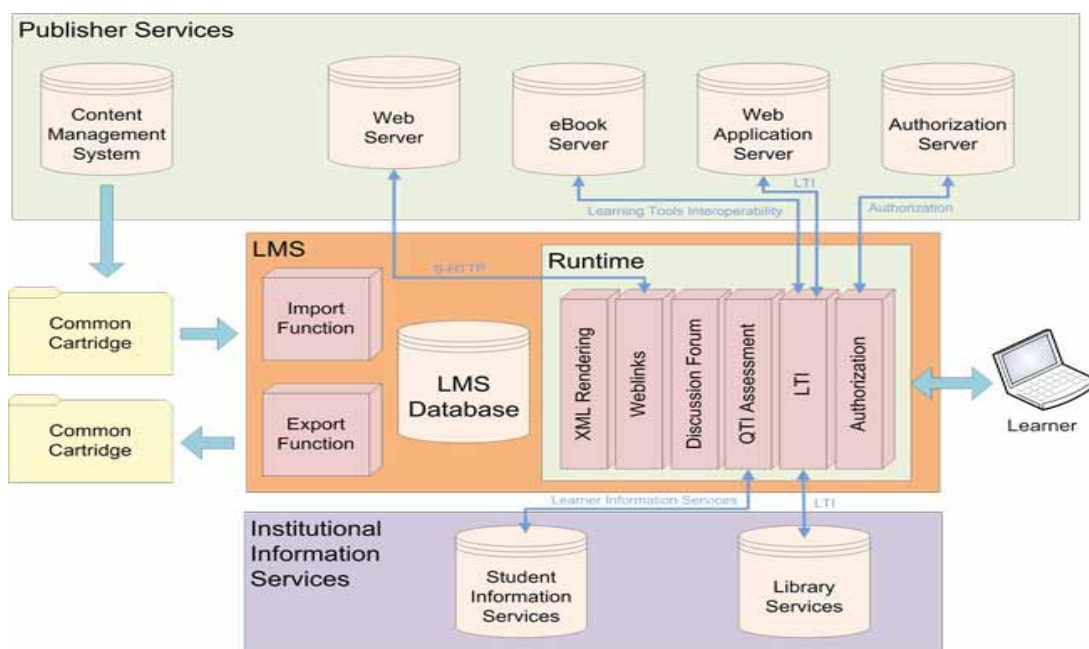


Figura 3.8. Formato de intercambio IMS-CC. Fuente :IMS Global.

A continuación se describirán los tipos de contenidos de IMS-CC (IMS-CC, 2011) :

- **Item-Folder:** una carpeta o folder representa una unidad de organización. Una carpeta es una colección de elementos o ítems y subcarpetas que siguen un orden.
- **Recurso – Contenido Web:** los archivos de contenido Web incluyen cualquier fichero que puede ser entregado sobre la Web.
- **Recurso – Enlace Web:** representa un objeto de aplicación de aprendizaje. Es enlace HTTP estándar.
- **Recurso – Tema de Discusión:** es un objeto de aplicación del aprendizaje que es utilizado para iniciar una discusión. El LMS espera generar un nuevo tema de discusión usando solo sus herramientas internas.
- **Recurso – Evaluación:** representa una instancia de una evaluación QTI. Una evaluación puede contener un número de atributos como: número de intentos, límite de tiempo, etc.
- **Recurso – Contenido Asociado:** es una colección de archivos usados exclusivamente por un objeto de aplicación de aprendizaje. Referencia interna de paquetes. Esto permite que los objetos de aplicación de aprendizaje o ficheros que se encuentran en el paquete puedan referenciar a otros archivos que también están dentro del paquete.

- Paquete de Metadatos IMS-CC: son metadatos al nivel del paquete IMS-CC y puede contener información acerca de: accesibilidad, descripción, etc.
- Banco de Preguntas: puede incluirse, de forma opcional, un banco de preguntas dentro de IMS-CC.

3.1.5. ADL/SCORM

ADL (Advanced Distributed Learning) (ADL, 2005) es un programa formado por el Departamento de Defensa de Estados Unidos y la Oficina de Política Científica y Tecnológica de la Casa Blanca con el objetivo de proporcionar formación de alta calidad, según sus necesidades.

ADL se ha centrado desde un principio en el aprendizaje sobre la Web. Actualmente es el modelo más utilizado en la industria y que cuenta con mayor cantidad de herramientas que lo soportan. Es un perfil de aplicación, ya que combina muchas especificaciones y las particulariza para un caso concreto. Las especificaciones, por su generalidad, dejan sin fijar aspectos que son necesarios para facilitar la implementación final, y SCORM (Sharable Courseware Object Reference Model) trata de ser más preciso para lograr una mayor compatibilidad.

En particular SCORM integra las siguientes especificaciones de grupos como IMS Global Consortium, ARIADNE, AICC o IEEE-LTSC (Fernández Manjón et. al., 2007):

- IEEE Learning Object Meta-data 1484.12 (IEEE LOM 2002): empleado en el Modelo de Agregación de Contenido para definir los Metadatos de los objetos de contenido.
- IEEE ECMAScript API for Content to Runtime Services Communication 1484.11.2 (IEEE EACRSC 2003): utilizado por el Entorno de Tiempo de Ejecución para definir el mecanismo de comunicación entre el contenido y el LMS.
- IEEE Data Model for Content Object Communication 1484.11.1 (IEEE DMCOOC 2002): empleado por el Entorno de Tiempo de Ejecución para definir el modelo de datos empleado en la comunicación entre el contenido y el LMS.
- AICC/Web-Based CMI Guidelines (AICC WBCMIG 1998): utilizado para definir la estructura del contenido en el Modelo de Agregación de Contenido.
- IMS Content Packaging (IMS-CP): empleado en el Modelo de Agregación de Contenido para agrupar objetos de contenido.
- IMS Simple Sequencing (IMS-SS): utilizado para el secuenciamiento de

actividades en un curso.

SCORM propone un entorno de ejecución, un modelo de metadatos y un modelo de la estructura de los cursos (modelo de agregación de contenidos). En su versión 2004 este modelo ha pasado a incluir también la secuenciación y navegación (Sequencing and Navigation) de los contenidos.

La definición del modelo SCORM, así como su evolución y las distintas decisiones de diseño tomadas durante el proceso de especificación, se basan en 6 principios esenciales (ADL, 2011):

- **Accesibilidad:** localizar y acceder a componentes instruccionales desde una ubicación remota y su envío a otras muchas localizaciones.
- **Interoperabilidad:** tomar componentes instruccionales desarrollados en una ubicación determinada y empleando unas herramientas y plataformas determinadas para su posterior aplicación en otra ubicación y otro conjunto de herramientas y plataformas.
- **Durabilidad:** resistir la evolución de la tecnología y futuros cambios sin incurrir en rediseños, reconfiguraciones o recodificaciones excesivamente costosas.
- **Reusabilidad:** flexibilidad para incorporar componentes instruccionales en múltiples contextos y aplicaciones.

Los elementos más característicos del modelo son (Fernández Manjón et. al., 2007, SCORM 2004 4th Edition Specification, 2009):

a) CAM (Content Aggregation Model): El modelo de Agregación de contenidos:

- Describe cómo deben ser los objetos de aprendizaje.
- Especifica cómo describir los objetos de aprendizaje para facilitar su búsqueda y localización.
- Define cómo agruparlos y empaquetarlos para crear unidades más complejas que puedan ser transportadas entre diferentes sistemas.
- Especifica las reglas para establecer una secuencia de objetos de aprendizaje que conformen unidades más complejas (tales como cursos).

Para entender el modelo de datos SCORM, es necesario conocer cada uno de los elementos de SCORM que construyen un recurso de aprendizaje:

- **Asset.:** es el bloque básico de construcción de un recurso de aprendizaje. Los assets son una representación electrónica de un medio de comunicación: texto, imágenes, sonidos o cualquier dato que pueda ser mostrado en un cliente Web. Es importante indicar que un asset puede ser construido por varios assets y que pueden describirse mediante metadatos, facilitando la búsqueda y reutilización de estos.
- **Sharable Content Object (SCO):** es el nombre que reciben los OA de SCORM. A través de los SCOs los estudiantes acceden directamente a los contenidos de aprendizaje, según el resultado de esta interacción cada SCO se comunica con el LMS quien se encarga de gestionar y administrar los SCOs.
Para mejorar la reutilización de los SCO, éste debe ser todo lo independiente posible del contexto de aprendizaje. Si mantiene relaciones muy estrechas con otros objetos, será muy difícil poder aprovecharlo en otros entornos (Figura 3.9)

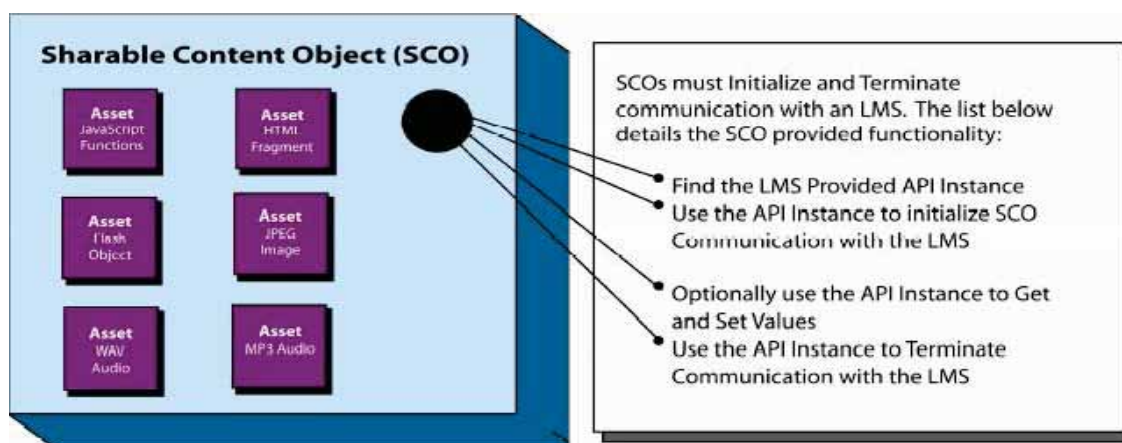


Figura 3.9. Arquitectura conceptual de SCO. Fuente: SCORM 2004 4th Edition Specification (2009).

- **Actividades:** puede ser descrita como una unidad de instrucción. Conceptualmente, es una acción que el estudiante realiza mientras progresa en su instrucción. Una actividad puede proporcionar al estudiante un SCO o asset. También puede estar compuesta de otras actividades (Figura 3.10).

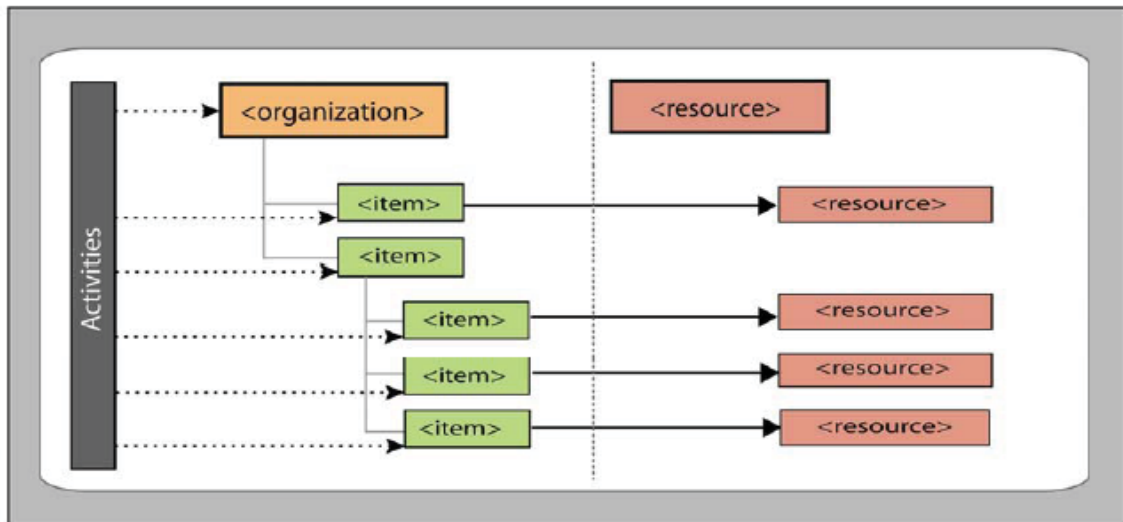


Figura 3.10. Representación conceptual de las actividades. Fuente : SCORM 2004 4th Edition Especification (2009).

- Organización de contenidos: es una representación o un mapa que define el uso previsto de los contenidos a través de unidades estructuradas de instrucción (actividades). El mapa muestra cómo las actividades se relacionan entre sí. La Figura 3.11 muestra un ejemplo de una organización de contenidos.

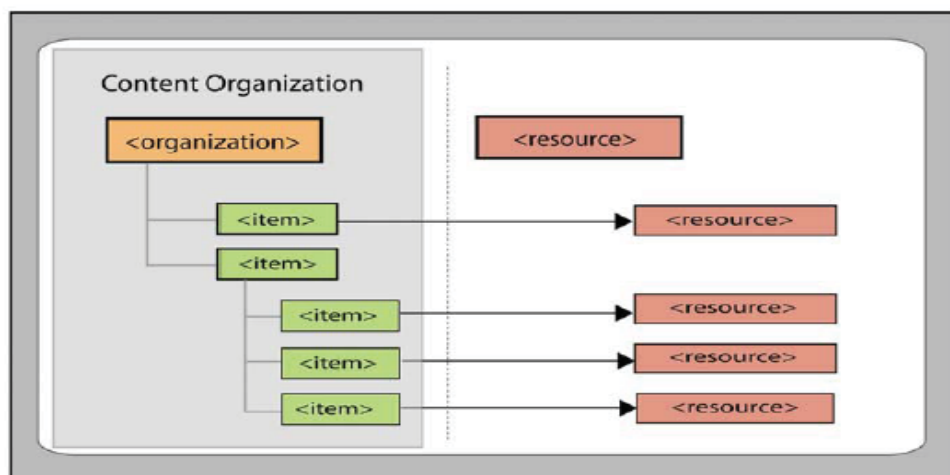


Figura 3.11. Ilustración conceptual de una Organización de contenidos. Fuente SCORM 2004 4th Edition Specification (2009).

- Agregación de contenidos: describe la acción o proceso para componer un conjunto de funcionalidades relacionadas con los OA o contenido, de tal forma que estos pueden ser aplicados en una experiencia de aprendizaje (Figura 3.12).

En términos del modelo de contenido SCORM, una agregación de contenidos también se utiliza para describir la entidad creada como parte de este proceso o acción. A veces el término se utiliza libremente para describir el paquete de contenido. La agregación de contenidos puede ser utilizados para entregar el contenido y la estructura de los contenidos prescritos, transferidos entre sistemas, o incluso almacenar en un repositorio.

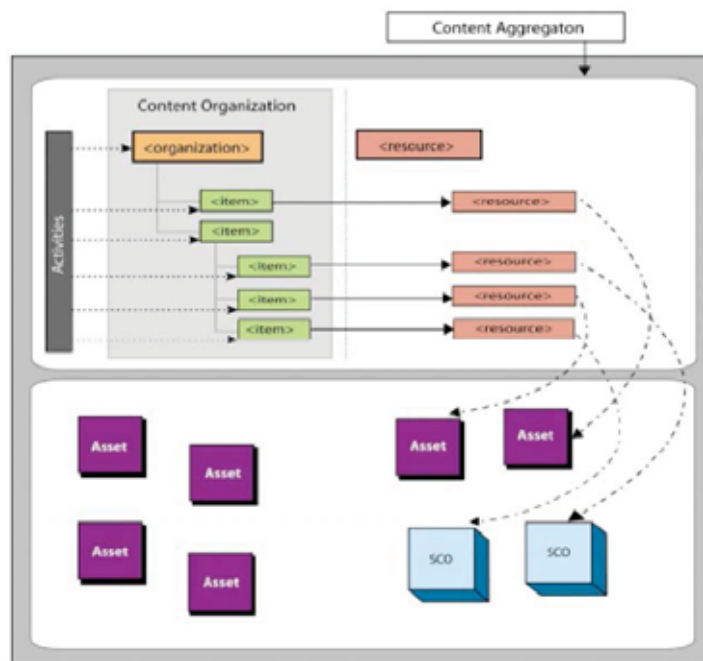


Figura 3.12. Ilustración conceptual de Agregación de Contenido .Fuente: SCORM 2004 4th Edition Specification (2009).

b) Entorno de Tiempo de Ejecución (Run-Time Environment, RTE): detalla los requisitos para la puesta en marcha de los objetos de contenido y los mecanismos de comunicación que tanto el LMS como el propio contenido deben emplear. Como se mencionaba en la descripción del Modelo de Agregación de Contenidos, en SCORM se distinguen dos tipos de material educativo: Los SCOs y los recursos: Assets.

El proceso de lanzamiento y ejecución debe estar estandarizado ya que el SCO, al ser lanzado, debe establecer un canal de comunicación con un LMS desconocido a priori. Esta comunicación se establece a través de un elemento enviado junto con el SCO al navegador del usuario. Este elemento consiste en una implementación de la interfaz de programación de aplicaciones (API) definida en el estándar IEEE 1484.11.2 y su provisión es responsabilidad del LMS, siendo el SCO completamente independiente del mecanismo de comunicación.

SCORM también define un modelo de datos y el comportamiento, es decir como deben ser

almacenados, procesados y utilizados por el LMS.

En definitiva, RTE establece (Figura 3.13):

- Una manera común de lanzar en ejecución los recursos educativos (Launch)
- Un mecanismo para la comunicación (API)
- Un vocabulario común (Modelo de Datos)

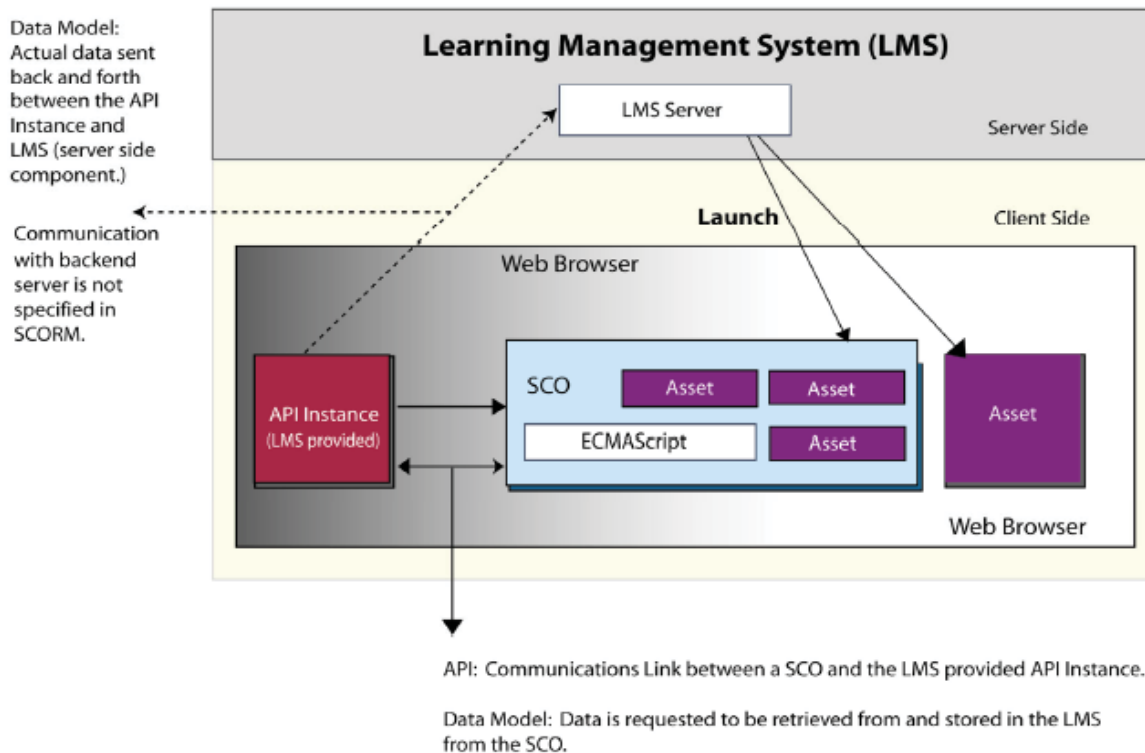


Figura 3.13. Entorno de Ejecución (RTE). Fuente: SCORM 2004 3rd Edition Specification (2006).

c) Modelo de Secuenciación y Navegación (SN): define la secuencia de contenidos para un usuario (generada por el LMS o fruto de la interacción explícita del usuario) y cómo se interpretan las reglas de secuenciación asociadas a los contenidos.

Para realizar esto, ADL SCORM se basa en el estándar de secuenciación de IMS (IMS-SS) (IMS-SS, 2003).

Para realizar esta tarea, SCORM emplea la estructura de contenido que es una herramienta utilizada para describir la relación jerárquica de una experiencia de aprendizaje (Figura 3.14). IMS-SS describe esta estructura como árbol de actividad y permite a SCORM utilizarlo para describir requisitos de información y de procesamiento, como: algoritmos de secuenciación y comportamientos.

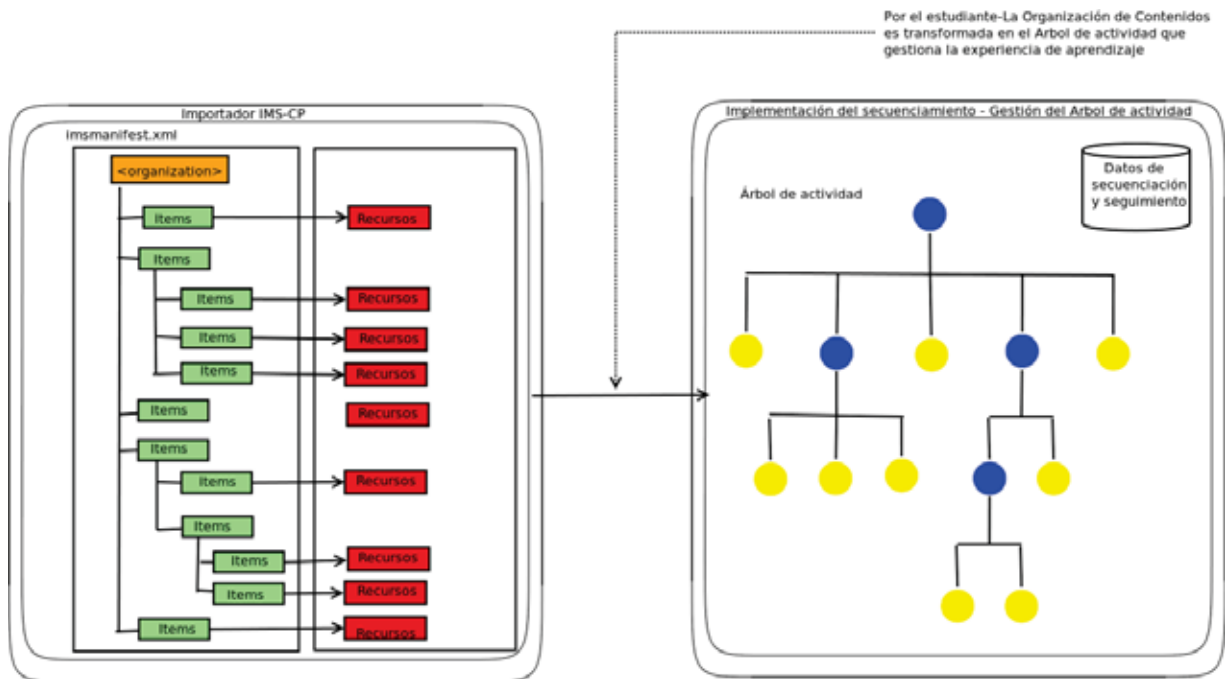


Figura 3.14. Relación entre una Organización de Contenidos y el Árbol de actividad. Adaptado de SCORM 2004 4th Edition Specification (2009).

Cabe destacar que la especificación SCORM no detalla como se deben implementar tecnológicamente estos objetos limitándose a especificar un mecanismo de comunicación independiente de la tecnología. Los fabricantes pueden escoger como realizar las implementaciones sin preocuparse en las cuestiones que se ocupa SCORM. Para ello deberán emplear una herramienta certificada por SCORM e indicar qué información desea que el SCO le transmita al LMS.

A modo de síntesis, la Figura 3.15 ilustra los distintos modelos del perfil SCORM.

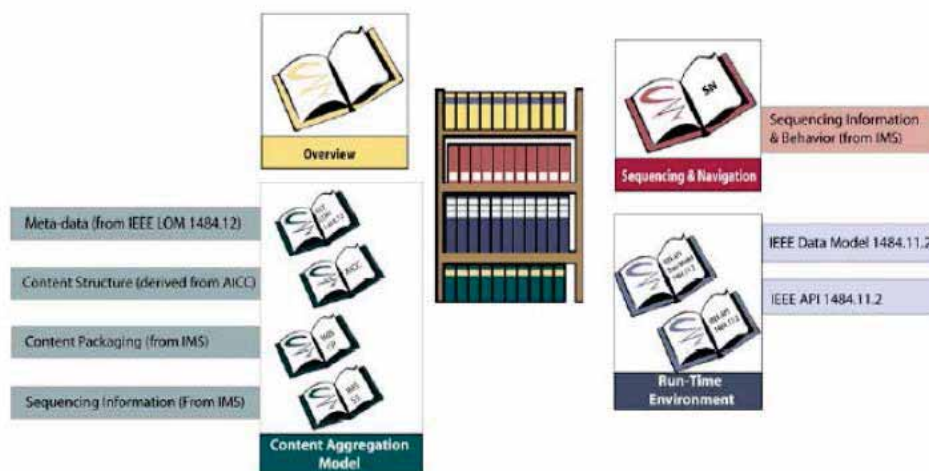


Figura 3.15. La biblioteca SCORM. Fuente : SCORM 2004 4th Edition Specification (2009).

3.1.6. ISO/IEC 19788 MLR

Actualmente, la Organización ISO/IEC, está desarrollando un nuevo estándar de definición de metadatos para objetos educativos, denominado ISO/IEC 19788 Metadata Learning Resource (2011), cuyas siglas son MLR. Se ha publicado la parte 1 del estándar y el resto se encuentran en fase de borrador, y toma como referencia principal para su creación otros estándares de metadatos ampliamente aceptados, tales como IEEE 1484.12 LOM y ISO/IEC 15836:2009 The Dublin Core Metadata Element Set.

Aunque LOM es uno de los estándares más utilizado, algunos elementos LOM pueden ser calificados como imprecisos por ser difícil encontrar una relación del nombre del elemento con los valores almacenados (ISO/IEC 19788, 2011)

Por otra parte LOM adolece de elementos pedagógicos como pueden ser el tipo de alumno, tipo de enseñanza o estrategias didácticas. La parte 5 del estándar tiene una relación directa con un registro de metadatos enfocado a recursos educativos. . Estos son:

- Método educacional: Proceso para generar conocimiento, capacidades y habilidades. Como ejemplo de un valor posible podría ser “aprendizaje colaborativo”.
- Anotación: Comentario textual de un usuario o colaborador sobre el recurso educativo.
- Colaborador: Agente responsable de realizar aportaciones para la descripción del recurso educativo.
- Nombre: Nombre de la persona individual que realiza la descripción pedagógica.
- Rol del colaborador: Función del colaborador que realiza la descripción pedagógica.
- Organización: Organización o servicio que aporta la descripción pedagógica.
- Audiencia: Clase o entidad a la que se dirige el recurso educativo.
- Nivel de la audiencia: Nivel de progresión en una secuencia educativa. Ejemplo: educación primaria.
- Rango de edad: Grupo de edad de los usuarios del recurso educativo

La Organización ISO/IEC pretende aprovechar los resultados de las experiencias en la implantación de estándares de metadatos, en especial del estándar IEEE LOM, y plantea en la definición de MLR diseñar un nuevo estándar con un nuevo enfoque hacia la utilidad y adaptabilidad de los metadatos para recursos educativos, que además admita integración y compatibilidad con anteriores estándares de metadatos.

3.1.7. Perfiles de aplicación LOM

Dado que ningún estándar puede cubrir las necesidades de todas las aplicaciones y contextos educativos son diversos, se considera en la actualidad que las especificaciones son un marco general de interoperabilidad que proporcionan un margen de adaptación a las necesidades concretas de cada dominio o aplicación (Fernández Manjón et.al. ,2007).

Duval et. al. (2002) lo define como una adaptación de la especificación, la cual se implementa mediante una extensión del mismo.

Un perfil de aplicación también posibilita la creación de extensiones de usuario para realizar ampliaciones, ajustes y adaptaciones al estándar es decir:

- a) Para añadir un nuevo elemento de datos que sólo tenga un interés en cierto contexto. Por ejemplo “nivel de accesibilidad” en un contexto educativo de educación especial para estudiantes con limitaciones físicas, visuales, auditivas, etc.
- b) Añadir nuevos valores posibles a un elemento de datos. Por ejemplo en el elemento “nivel educativo”, definido en la parte 9 del estándar dedicada a la clasificación elementos pedagógicos, se podría establecer un conjunto cerrado de posibles valores a seleccionar entre ellos, adecuando estos valores al nombre concreto de los niveles de enseñanzas de un país.

No obstante, la creación de perfiles de aplicación específicos dificulta la interoperabilidad de aplicaciones fuera de los contextos a los que están dirigidos, ya que dichas aplicaciones externas no estarán obligadas a *entender* las particularidades del perfil de aplicación.

En particular, dado el reconocimiento de IEEE LOM como estándar, algunas iniciativas lo utilizan de base para la definición de sus especificaciones como el caso de IMS y SCORM. Esto es posible pues IEEE LOM puede extenderse, es decir, si la aplicación lo requiere se pueden agregar elementos.

No hay duda de la complejidad de IEEE LOM, es por ello que muchas organizaciones realizan adaptaciones que según Friesen, Fischer & Roberts (2004) pueden clasificarse en cuatro grupos principales:

1. Los que combinan LOM con elementos de otras especificaciones o estándares de metadatos.
2. Los que se enfocan en la definición de elementos de extensión y otras adaptaciones de LOM.
3. Los que hacen énfasis en la reducción de los elementos de LOM.

4. Los que combinan la reducción de los elementos LOM y, además, definen elementos de extensión.

A continuación, se describen y analizan algunas de las iniciativas que se están desarrollando.

a) CanCore

El perfil de aplicación CanCore (Canadian Core Learning Resource Metadata Application Profile, <http://www.cancore.ca/en/index.html>) se creó seleccionando un subconjunto de los metadatos LOM en forma minuciosa precisando el significado de estos metadatos a un nivel de granularidad muy fina.

Según Friesen et. al. (2004), CanCore agrega valor a IEEE LOM simplificándolo y redefiniéndolo a través del desarrollo de vocabulario adecuado a los sectores educativos canadienses.

CanCore representa la adaptación de un estándar para atender las necesidades específicas de una particular comunidad de implementación con una aplicación de requisitos comunes (Lynch, 1997). Estas comunidades están construidas por público educativo de forma convencional y a distancia en Canadá, incluyendo la educación primaria, secundaria y universitaria.

Es así como el perfil de CanCore está siendo utilizado para diversos proyectos Canadienses: CANARIE-supported Portal for Online Objects for Learning liderado por TeleLearning (POOL; <http://www.newmic.com/pool/>) y el Broadband Enabled Lifelong Learning Environment (BELLE; <http://www.netera.ca/belle/>), así como también el proyecto de aprendizaje Alberta (<http://www.learnalberta.ca/>). CanCore soporta además la iniciativa de repositorio enfocado a educación post-secundaria a través del Campus Alberta Repository of Education Objects (<http://www.careo.org>).

b) LOM-ES

El perfil de aplicación LOM-ES (2006) se encuentra actualmente en desarrollo, a través del Subcomité 36 de la Asociación Española de Normalización y Certificación (AENOR, <http://www.aenor.es/>). A partir de los esquemas de definición de IEEE LOM v1.0 se han obtenido los esquemas de LOM-ES v1.0. Los cambios más relevantes, en términos generales, son los siguientes (Canabal, 2008):

- Se han añadido nuevos elementos que no aparecían en los esquemas originales de IEEE LOM.

- Se han añadido nuevos términos en los vocabularios controlados de IEEE LOM.
- Se han modificado cardinalidades de elementos de IEEE LOM.
- Se han modificado restricciones de obligatoriedad en algunos términos.
- En LOM-ES es necesario mantener la ordenación, tanto de categorías como de elementos, que marcan los esquemas. Esta restricción no existe en IEEE LOM.

El desarrollo del estándar LOM-ES significa un importante aporte para mejorar la gestión de la especificación de metadatos LOM, el cambio de opcionalidad por obligatoriedad para algunos de los metadatos ayuda a que exista uniformidad en relación al tipo de información a introducir y la definición de vocabularios específicos permite especificar concretamente esa información, de esta manera se facilita la búsqueda y recuperación de los OAs. Por otra parte, la traducción al español de los metadatos de LOM promueve de forma significativa la posibilidad de uso de metadatos para los hispano hablantes (Canabal, Sarasa y Sacristán, 2008)

3.2 Dimensiones de análisis de un OA

El tratamiento de los OAs desde la concepción presentada en esta tesis y conceptualizada en el capítulo 2 hacen posible que se analice desde dos perspectivas: pedagógica y tecnológica. Es por ello que a continuación se presentan por un lado las distintas teorías educativas y por otro las teorías del diseño instruccional que han influido en la creación de los OA.

3.2.1. Aproximación pedagógica

Como señala García Aretio (2007), la preocupación por disponer elementos de aprendizaje reutilizables no es nueva. En este mismo sentido Downes (2002) señala que tanto los profesores como las instituciones reutilizan apuntes, esquemas didácticos, libros y otros recursos docentes que se emplean conjuntamente en múltiples contextos, con estudiantes y entornos pedagógicos muy distintos.

La preocupación actual está en cómo compartir esos elementos en un contexto digital, en el que es necesario diseñar con antelación los procesos donde tendrán cabida los recursos (García Aretio, 2005).

Los actuales postulados sobre la forma de utilizar OA en procesos de aprendizaje están

fundamentados en tres teorías de aprendizaje: Conductismo, Cognitivismo y Constructivismo que detallan a continuación

a) Conductismo: se basa en los cambios observables en la conducta del sujeto. Se enfoca hacia la repetición de patrones de conducta hasta que estos se realizan de manera automática (Schuman, 1996).

La teoría del conductismo se concentra en el estudio de conductas que se pueden observar y medir (Good y Brophy, 1990). Ve a la mente como una “caja negra” en el sentido de que las respuestas a estímulos se pueden observar cuantitativamente ignorando totalmente la posibilidad de todo proceso que pueda darse en el interior de la mente. Algunas personas claves en el desarrollo de la teoría conductista incluyen a Pavlov, Watson, Thorndike y Skinner.

b) Cognoscitivismo: se basa en los procesos que tienen lugar atrás de los cambios de conducta. Estos cambios son observados para usarse como indicadores para entender lo que está pasando en la mente del que aprende. (Schuman, 1996).

Good & Brophy (1990) afirman:

Los teóricos del cognoscitivismo reconocen que una buena cantidad de aprendizaje involucra las asociaciones que se establecen mediante la proximidad con otras personas y la repetición. También reconocen la importancia del reforzamiento, pero resaltan su papel como elemento reafirmador para corrección de respuestas y sobre su función como un motivador. Sin embargo, inclusive aceptando tales conceptos conductistas, los teóricos del cognoscitivismo ven el proceso de aprendizaje como la adquisición o reorganización de las estructuras cognitivas a través de las cuales las personas procesan y almacenan la información. (p.187).

c) Constructivismo: se sustenta en la premisa de que cada persona construye su propia perspectiva del mundo que le rodea a través de sus propias experiencias y esquemas mentales desarrollados. El constructivismo se enfoca en la preparación del que aprende para resolver problemas en condiciones ambiguas.

El que aprende construye su propia realidad o al menos la interpreta de acuerdo a la percepción derivada de su propia experiencia, de tal manera que el conocimiento de la persona es una función de sus experiencias previas, estructuras mentales y las creencias que utiliza para interpretar objetos y

eventos (...) lo que alguien conoce es aterrizado sobre las experiencias físicas y sociales las cuales son comprendidas por su mente.(Jonassen, 1991, p.10).

Coincidiendo con Cesar Coll (1990; 1996) y Diaz Barriga y Hernández Rojas (2002) en cuanto a que:

.....la postura constructivista en la educación se alimenta de las aportaciones de diversas corrientes psicológicas: el enfoque psicogenético de Piaget, la psicología sociocultural de Vigotsky. la teoría de Ausubel de la asimilación y el aprendizaje significativo, entre otras. A pesar de que los autores se sitúan en encuadres distintos, comparten el principio de la importancia de la actividad constructiva del alumno. (Diaz Barriga y Hernández Rojas, 2002, p. 28)

Piaget sostiene que, a partir de unas capacidades generales con las que se nace, los sujetos van construyendo su inteligencia, al mismo tiempo que construyen todo su conocimiento sobre la realidad. Esto lo realizan actuando sobre el mundo físico y social, experimentando con los objetos y situaciones, y transformándolos. Pero Piaget no se limita a proponer que el sujeto es activo cuando adquiere el conocimiento, que lo tiene que construir y que no se reduce a reproducir lo que se le transmite (Delval, 1997)

En la teoría del constructivismo de Vigotsky (1979) o constructivismo social, llamada también constructivismo situado, el aprendizaje tiene una interpretación audaz: sólo en un contexto social se logra aprendizaje significativo. El constructivismo social no niega nada de las suposiciones del constructivismo psicológico, sin embargo considera que está incompleto. Lo que pasa en la mente del individuo es fundamentalmente un reflejo de lo que pasó en la interacción social y la cultura, tanto así, porque, según él, plantea que los procesos psicológicos superiores (lenguaje, razonamiento, comunicación etc.) se adquieren en interrelación con los demás, lo que un individuo puede aprender, de acuerdo a su nivel real de desarrollo, varía ostensiblemente si recibe la guía de un adulto o puede trabajar en conjunto con otros compañeros.

Otros de los conceptos esenciales en la obra de Vygotsky (1978) según sus propios términos son:

- La Zona de Desarrollo Próximo (ZDP): "No es otra cosa que la distancia entre el nivel real de desarrollo, determinado por la capacidad de resolver independientemente un problema".
- El Nivel de desarrollo potencial: es determinado a través de la resolución de un problema bajo la guía de un adulto o en colaboración con un compañero más capaz.

Uno de los autores que más ha influido en la elaboración y divulgación de las ideas que se expusieron es Ausubel (1983). Para que realmente sea significativo el aprendizaje (concepto acuñado por Ausubel), éste debe reunir varias condiciones: la nueva información debe relacionarse de modo no arbitrario y sustancial con lo que el alumno ya sabe, dependiendo también de la disposición (motivación y actitud) de éste por aprender, así como de la naturaleza de los materiales o contenidos de aprendizaje.

Ausubel, postula que el aprendizaje implica una reestructuración activa de las percepciones, ideas, conceptos y esquemas que el aprendiz posee en su estructura cognitiva (Díaz Barriga y Hernández, 2002).

El modelo constructivista está centrado en la persona, en sus experiencias previas de las cuales realiza nuevas construcciones mentales, por lo tanto, se considera que la construcción se produce: cuando el sujeto interactúa con el objeto del conocimiento (Piaget); cuando esto lo realiza en interacción con otros (Vigotsky); cuando es significativo para el sujeto (Ausubel) (Méndez Casanova, 2010).

3.2.2. Aproximación tecnológica: las teorías del aprendizaje y el diseño instruccional

Coincidiendo con García Aretio (2007), un segundo camino de aproximación a los OA es la tecnológica. Esta línea de desarrollo parece ser la elegida en la actualidad.

Los primeros desarrolladores de objetos en e-learning partieron de un enfoque claramente centrado en los contenidos. Este planteo está aún presente en buena parte de los informáticos que trabajan en e-learning, se resume en la siguiente concepción:

Los objetos de aprendizaje son los elementos de un nuevo tipo de instrucción basada en el ordenador y fundamentada en el paradigma computacional de "orientación al objeto". Se valora sobre todo la creación de componentes (llamados objetos) que pueden ser reutilizados en múltiples contextos. Ésta es la idea fundamental que se esconde tras los objetos de aprendizaje: los diseñadores instruccionales pueden construir pequeños componentes de instrucción (en relación con el tamaño de un curso entero) que pueden ser reutilizados varias veces en contextos de estudio diferentes. (Wiley, 2000, p. 3).

Desde hace un tiempo se ha abordado la necesidad de establecer un puente entre el aprendizaje y la práctica educativa; diversos autores han propuesto diferentes alternativas para lograr la transformación de la teoría en práctica es decir, generar una "función

puente”(Ertmer y Newby, 1993) que logre traducir aspectos relacionados con las teorías de aprendizaje en acciones óptimas de instrucción a través del denominado Diseño Instruccional . En ese orden de ideas, los diseñadores instruccionales se han dedicado a incorporar los principios del aprendizaje y la instrucción en especificaciones para materiales y actividades de instrucción (Smith y Ragan, 1993; citados por Ertmer y Newby, ob. cit.)

En este punto, como señala Garcia Aretio (2007) los pedagogos y los tecnólogos vienen empleando un lenguaje común con el fin de enlazar entre sí la calidad didáctica y la accesibilidad tecnológica en los procesos de creación de OAs.

El recorrido histórico del diseño instruccional se asemeja al de las teorías de aprendizaje y a continuación se presenta una revisión de la puesta en práctica de dichas teorías, finalizando con una perspectiva ecléctica a la cual se adhiere en esta tesis.

a) **Conductismo**

Seattler (2004) identifica seis áreas que mostraron impacto del conductismo en la tecnología educacional de los Estados Unidos: el movimiento de objetivos conductistas; la fase de la máquina de enseñanza; el movimiento de la instrucción programada; la aproximación de la instrucción individualizada; el aprendizaje asistido por computadora y la aproximación de sistema para la instrucción.

Para el logro de objetivos conductistas, una tarea de aprendizaje debe segmentarse mediante el análisis hasta lograr tareas específicas medibles. El éxito del aprendizaje se determina mediante la aplicación de pruebas para medir cada objetivo.

Ertmer y Newby (1993) señalan los supuestos o principios básicos de esta teoría que consideran son pertinentes al diseño instruccional:

- Énfasis en producir resultados observables y mensurables.
- Evaluación previa de los estudiantes para determinar dónde se debe iniciar la instrucción.
- Énfasis en el dominio de los primeros pasos antes de progresar a niveles más complejos de desempeño.
- Uso de refuerzos para impactar al desempeño.
- Uso de pistas o indicios, modelaje y práctica para asegurar una fuerte asociación estímulo-respuesta.

b) Cognoscitivismo

El objetivo de instrucción es mantener la comunicación o transferencia de conocimiento hacia el que aprende en la forma más eficiente y efectiva posible (Bednar et al., en Anglin, 1995). En el caso del conductismo, el instructor que busca un método más eficiente a prueba de fallas para que su aprendiz logre su objetivo, subdivide una tarea en pequeñas etapas de actividades. El investigador cognoscitivista analizaría una tarea, la segmentaría en pequeñas partes y utilizaría esa información para desarrollar una estrategia que va de lo simple a lo complejo.

En este caso, Ertmer y Newby (1993) señalan los principios básicos de esta teoría que consideran pertinentes al diseño instruccional:

- Retroalimentación.
- Análisis del estudiante y la tarea.
- Énfasis en la participación activa del estudiante en el proceso de aprendizaje.
- Uso de análisis jerárquico para identificar e ilustrar relaciones.
- Énfasis en la estructuración, organización y secuencia de la información para facilitar su óptimo procesamiento.
- Creación de ambientes de aprendizaje que permitan y estimulen al estudiante a hacer conexiones con material previamente aprendido.

c) Constructivismo

Promueve experiencias de aprendizaje más abierto, en los que los métodos y resultados del aprendizaje no son tan fácilmente medibles y podrían ser diferentes entre cada estudiante.

El conductismo y el constructivismo son totalmente diferentes desde sus perspectivas teóricas, pero el cognoscitivismo comparte algunas similitudes con el constructivismo. Un ejemplo de su compatibilidad es el hecho de que comparten la analogía de comparar el procesamiento mental de la información con el de las computadoras.

En este sentido, Perkins expresa:

Los modelos de procesamiento de la información han adoptado el modelo de computadora de la mente como un procesador de información. El constructivismo agrega que este procesador de información debe verse justamente como un sorteador de datos, pero manejando su flexibilidad durante el aprendizaje – haciendo hipótesis, probando las interpretaciones tentativas, etc.(Perkins, 1991, p.21 en Schwier, 1998).

En cuanto a lo señalado por Ertmer y Newby (1993), estos autores consideran que los supuestos pertinentes de esta teoría para el diseño instruccional son:

- Énfasis en la identificación del contexto en el cual las habilidades serán aprendidas y aplicadas.
- Énfasis en el control por parte del estudiante para manipular la información.
- Necesidad de que la información se presente en una amplia variedad de formas.
- Apoyar el uso de las habilidades de solución de problemas que permitan al estudiante ir más allá de la información presentada.
- Evaluación enfocada hacia la transferencia de conocimiento y habilidades.

Jonassen (1991) señala que el diseño instruccional basado en el conductismo y el cognitivismo tiene salidas predeterminadas e interviene en el proceso de aprendizaje para crear esquemas predeterminados de la realidad de un concepto en la mente del que aprende; mientras que el constructivismo se reserva porque las salidas del aprendizaje generalmente son impredecibles, la instrucción debe reforzar, más no moldear el aprendizaje.

En síntesis, Jonassen (1991) indica que un proceso de diseño instruccional constructivista se debe relacionar con el diseño de ambientes que favorezcan la construcción del conocimiento que:

- Proporcione múltiples representaciones de la realidad y la complejidad del mundo natural, evitando (sobre) simplificaciones de la instrucción.
- Permita al aprendiz realizar actividades reales auténticas, que estén contextualizadas.
- Proporcione un mundo real: ambientes de aprendizaje basados en casos, en lugar de instrucciones secuenciales predeterminadas.
- Refuerce la práctica de reflexionar.
- Faculte contextos, contenidos y conocimientos dependientes de la construcción.
- Soporte la construcción colaborativa de conocimientos a través de la negociación social, y no ponga a competir a los estudiantes por el reconocimiento.

Mergel (1998) resume una serie de principios para el diseño instruccional desde una perspectiva constructivista:

- El diseñador produce estrategias y materiales de naturaleza mucho más facilitadora que prescriptiva.

- Los contenidos no se especifican, la dirección es determinada por el que aprende y la evaluación es mucho más subjetiva, ya que no depende de criterios cuantitativos específicos, sino que se evalúan los procesos y el aprendiz realiza autoevaluaciones.
- La prueba de dominio estándar a base de papel y lápiz no se usa en un diseño instruccional constructivista; en su lugar se realizan evaluaciones basadas en resúmenes o síntesis, productos acabados y publicaciones.

d) Una Aproximación Ecléctica para una Teoría en el Diseño Instruccional

Luego de este recorrido por las aplicaciones en el Diseño Instruccional de las teorías de aprendizaje, cabe preguntarse cual es la más adecuada. Coincidiendo con Ertmer y Newby(1993), el aprendizaje es un proceso complejo que parece estar fuertemente influenciado por nuestros conocimientos previos, quizás la mejor respuesta a esta pregunta es “depende”.

En la medida en que las personas adquieren más experiencia con un determinado contenido, progresan en su conocimiento desde que son capaces de reconocer y aplicar las reglas, hechos y operaciones estándares de una profesión (saber que); pasando luego a pensar como un profesional y extrapolan a partir de estas reglas generales a los casos problemáticos particulares (saber cómo) y luego desarrollan y verifican nuevas formas de comprensión y de acción cuando las categorías y formas de pensamiento familiares fracasan (reflexión en acción) (Schön, 1987, citado por Ertmer y Newby,1993) .

Jonnassen, Mayes & McAleese,(1993, citado por Mergel, 1998) en su trabajo “Manifiesto for a Constructive Approach to Technology in Higher Education” , identifica los siguientes situaciones de aprendizaje y los relaciona con la teoría que él considera más adecuada.

- Aprendizaje introductorio: los aprendices tienen muy poco conocimiento previo transferible directamente o habilidades acerca de los contenidos. Se encuentran al inicio del ensamble e integración del esquema. En esta etapa el diseño instruccional clásico es el más adecuado porque está determinado, es restringido, es secuencial y se usan referencias. Esto permitirá a los estudiantes desarrollar sus propias anclas que les sirvan como referencia para futuras exploraciones.
- Adquisición de conocimientos avanzados: los siguientes conocimientos introductorios y los conocimientos más especializados posteriores, se pueden lograr mediante una aproximación constructivista no muy intensa.
- La adquisición de conocimientos expertos : la etapa final, en la que el aprendiz

es capaz de tomar decisiones inteligentes dentro del ambiente de aprendizaje, la aproximación constructivista funcionará espléndidamente bien.

En este mismo sentido, Ertmer y Newby(1993), relacionan las tareas a realizar con las estrategias más adecuadas:

Conductismo

Las tareas que requieren un bajo grado de procesamiento (por ejemplo, las asociaciones de pares, discriminación, la memorización) parecen facilitarse mediante estrategias más frecuentemente asociadas con las salidas conductuales (por ejemplo, estímulos respuesta, continuidad de retroalimentación y reforzamiento).

Cognitivismo

Las tareas que requieren un nivel superior de procesamiento (por ejemplo, clasificación, reglas o ejecuciones de procedimientos) están principalmente asociadas con las estrategias que tienen un fuerte énfasis cognoscitivista (por ejemplo, organización esquemática, razonamiento analógico, solución de problemas algorítmicos).

Constructivismo

Las tareas que demandan altos niveles de procesamiento (tales como: solución de problemas heurísticos (invención), selección de personal y monitoreo de estrategias cognitivas) frecuentemente son aprendidas mejor con estrategias avanzadas desde una perspectiva constructivista (ejemplo, aprendizajes ubicados, aprendizajes cognitivos, negociación social)

(Ertmer & Newby, 1993, pp. 23-27).

3.3. Creación de OA

Como se señaló al comienzo de este capítulo, además de las herramientas y los conocimientos sobre estándares y especificaciones e-learning, es necesario seguir una serie de pasos para la creación de OA. Esquemáticamente y siguiendo a Hernández (2005), consideramos cinco pasos que pueden ser ejecutados por diferentes organizaciones o incluso es posible obviar algunos de estos pasos (Figura 3.16). A continuación se explicarán cada uno de ellos.



Figura 3.16. Pasos a seguir para la creación de OAs. Fuente: Hernández (2005).

Con respecto a la construcción de los OA, nos referiremos con más detalle en el Capítulo 5. A continuación se explicarán las etapas de Empaquetado y Etiquetado que se asocian a lo desarrollado en este capítulo.

3.3.1. Empaquetado de OA

Una vez que el contenido de aprendizaje ha sido diseñado y construido, es necesario que dicho contenido esté disponible para: estudiantes, herramientas de autor, repositorios de contenido y LMS.

Para esta tarea, se creó el IMS-CP, el cual, fue diseñado para proporcionar una estructura estándar para el intercambio de contenidos. SCORM ha creado su propio paquete de contenido basándose en el paquete IMS, pero añadiendo información para poder empaquetar assests, OA y Organización de aprendizaje.

Como se explicó en el punto 3.1.3.1, la especificación IMS-CP, indica el formato en el que deben agruparse las colecciones de archivos con material educativo y, fundamentalmente, detalla la sintaxis de un archivo (Manifiesto) en el que se describen y estructuran los contenidos de un determinado paquete de contenido. Esta especificación no es estricta y permite la particularización de la sintaxis del Manifiesto para adaptar la especificación a distintos entornos y satisfacer las necesidades de las distintas organizaciones.

La especificación SCORM define dos perfiles de aplicación sobre IMS-CP para atender dos tipos de mecanismo de interoperabilidad: Los Paquetes de Contenido formados por Recursos y los Paquetes de Contenido formados por Agregaciones de Contenido.

a) Paquetes de Contenido formados por Recursos

Se emplea para empaquetar conjuntos de recursos y SCOs sin tener que especificar una organización o un contexto de aprendizaje. Este tipo de empaquetamiento es el mecanismo recomendado por SCORM para la interoperabilidad de contenidos entre distintos entornos

de aprendizaje. Al no definirse ningún tipo de estructura, los paquetes creados mediante este perfil de aplicación son transportables entre sistemas pero carecen de estructura y por tanto no son paquetes diseñados para ser consultados por los alumnos. En términos sencillos, estos paquetes son meras colecciones de recursos educativos sin diseño instruccional ninguno (Fernández Manjón et.al, 2007).

b) Paquetes de Contenido formados por Agregaciones de Contenido

Se emplea para agrupar los distintos contenidos educativos (recursos o SCOs) y la descripción de la estructura de estos contenidos.

En la Figura 3.17 se muestra una organización dentro de las organizaciones y un recurso dentro de los recursos. Un archivo Manifiesto real puede contener múltiples recursos y varias organizaciones. El Manifiesto también contiene un elemento de metadatos y uno o más sub-manifiestos (archivos Manifiestos anidados), elementos que describen los componentes del paquete. Ciertos atributos del Manifiesto (e.g., el identificador) han sido omitidos.

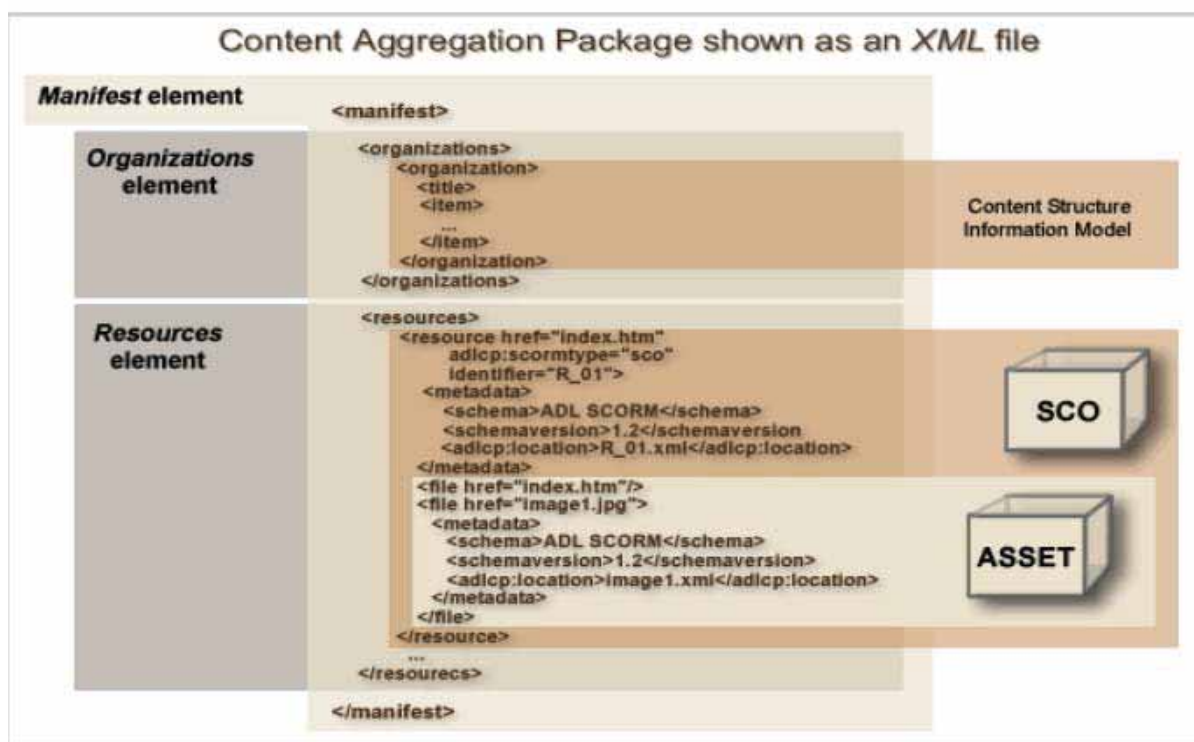


Figura 3.17. Ejemplo de un Paquete de Agregación de Contenidos y sus componentes. Fuente: SCORM 2004 4th Edition Specification (2009).

3.3.2. Etiquetado de OA

El etiquetado corresponde al proceso de agregación de metadatos a los OAs. Este proceso se puede llevar a cabo una vez que ya ha sido creado el OA, sin embargo podría considerarse como parte del proceso de creación porque para que un recurso se considere como OA debe tener incorporado metadatos que permitan su gestión.

Los metadatos se deben describir a través de un archivo XML independiente del recurso. Para facilitar su carga se han desarrollado una serie de herramientas que presentan una interfaz gráfica que facilita esta tarea y el código XML se genera sin que los usuarios intervengan.

A continuación se describen algunas de las herramientas más utilizadas.

a) LomPad

LomPad Version 1.0 (<http://helios.liceft.ca:8080/LomPad/en/index.htm>) es un editor de metadatos de código abierto, licencia GNU, diseñado por el LICEF Research Centre de Canada . Soporta el modelo SCORM y los perfiles LOM y SCORM, CanCore. Se encuentra disponible en los idiomas inglés y francés.

b) El proyecto Reload

Reload (Reusable eLearning Object Authoring & Delivery, <http://www.reload.ac.uk>), es un proyecto del JISC (Joint Information Systems Committee) de código abierto.

Contiene varios desarrollos:

- Reload Editor: Es un editor de metadatos y un empaquetador de contenido de código abierto desarrollado por JISC (Joint Information Systems Committee) . La versión 2.0.2 soporta IMS Metadata, IEEE LOM, IMS Content Packaging 1.1.4, SCORM 1.2, and SCORM 2004.
- Reload SCORM 1.2: se trata de una especie de LMS pero desprovisto de las herramientas de gestión de usuario y de discusión que habitualmente facilitan las plataformas de e-learning.
- Reload Learning Design Editor: permite diseñar actividades de aprendizaje según la especificación IMS-LD (Learning Design de IMS) (niveles A, B y C).
- Reload Learning Design Player: es un reproductor de IMS-LD, basado en CooperCore,

c) Exelearning

El proyecto Exe (<http://exelearning.org/>) es un programa de Código Abierto creado por la Auckland University of Technology y la Tairawhiti Polytechnic. Está financiado por el Tertiary Education Commission de Nueva Zelanda. A diferencia de los editores anteriores, exelearning, conocido también como (eXe) no sólo permite editar metadatos, sino que además es una herramienta de autoría para el diseño, desarrollo y publicación de materiales de enseñanza y aprendizaje a través de la Web. Posee las siguientes características:

- Proporciona capacidades profesionales de publicación web que pueden ser fácilmente referenciadas o importadas por LMS conformes a los estándares.
- Está desarrollado como herramienta de autoría "offline",
- Posee un ambiente intuitivo WYSIWYG.
- Permite agregar metadatos de Dublin Core, empaquetar el recurso con las especificaciones SCORM 1.2 o IMS-CP.

CAPITULO 4. METODOLOGÍAS DE DISEÑO DE UN OBJETO DE APRENDIZAJE (OA)

En este capítulo se analizan en la sección 4.1, las consideraciones propuestas por varios autores en cuanto al desarrollo de software educativo. Luego en la sección 4.2, a partir de la investigación bibliográfica, se presentan las distintas metodologías de desarrollo de la Ingeniería de Software (IS) adaptables a la construcción de OA. Se destaca además el marco de trabajo de esta tesis, el cual integra las actividades del Diseño Centrado en el usuario (DCU) con la metodología de desarrollo de software de sistemas interactivos.

4.1. Ingeniería de software educativo

Uno de los problemas más importantes con los que se enfrentan los ingenieros de software y los programadores en el momento de desarrollar un software de aplicación educativa, es la falta de marcos teóricos interdisciplinarios entre las áreas de trabajo (Galvis, 1996, Galvis, 2000, Ellis, Wagner & LongMire (1999), Barajas, Muñoz y Álvarez (2007).

Coincidimos con Borges (2002) en cuanto a que el desarrollo de software educativo integra tanto los aspectos ergonómicos (estudiados en la Ingeniería de Usabilidad (IU)) y los aspectos educativos (estudiados por las teorías educativas), como los aspectos funcionales (estudiados por la IS).

En el caso de la IS tomamos la definición de la IEEE Std. 610.12 -1990 (1990) de la Ingeniería de Software:

1. La aplicación de un enfoque sistemático, disciplinado y cuantificable para el desarrollo, operación y mantenimiento de software, es decir, la aplicación de la ingeniería al software,
2. el estudio de enfoques relacionados con (1)

Es decir, la IS, no sólo cubre los aspectos puramente tecnológicos de la producción del software, sino que conlleva además la gestión de los presupuestos, de proyectos y de los equipos de desarrollo, así como también de la planificación de dichos proyectos.

Comprende tres elementos clave (Granollers, 2004):

- Los modelos de proceso o métodos (para construir técnicamente software).
- Las herramientas (proporcionan el soporte para el desarrollo de los métodos).

- Los procedimientos (definen las secuencias de aplicación de los métodos y de enlace métodos-herramientas).

Coincidiendo con Pressman (2002), la calidad es la base de estos elementos, como lo podemos observar en la Figura 4.1



Figura 4.1 Capas de IS. Fuente: Pressman (2002).

Según Pressman (2002) la calidad del software es “la concordancia con los requerimientos funcionales y de rendimiento explícitamente establecidos, con los estándares de desarrollo explícitamente documentados y con las características implícitas que se espera de todo software desarrollado profesionalmente” (p.135).

La calidad debe ser tenida en cuenta en los desarrollos de software educativo (como los OA), pues como en todo proyecto de software, el ciclo de vida consiste en convertir los requerimientos planteados por los usuarios en un producto de software con funcionalidades específicas.

Aycart, Ginestá y Hernández (2007) señalan que cualquiera que sea la técnica o conjunto de técnicas que se utilicen para asegurar la calidad del software, existen un conjunto de principios básicos que se deben tener siempre presentes. A continuación enumeramos los principales:

- a) Es imperioso disponer en detalle de los requisitos del sistema y las necesidades de los usuarios. Los procesos de calidad se basan en verificar que el software cumple con los requisitos. A todas las pruebas se les debería poder hacer un seguimiento hasta los requisitos del usuario.
- b) Los procesos de control de calidad deben estar integrados a las distintas fases del proyecto de desarrollo del software.
- c) El equipo que prueba el software debe ser distinto al equipo de desarrollo. El grupo que desarrolla el software no debe ser en ningún caso el mismo que se responsabiliza del control de las pruebas de calidad. Para ser más eficaces, las

pruebas deberían ser realizadas por un equipo independiente.

En el ciclo de vida de desarrollo del software, el testing demarca ciertas especificidades para el software educativo, pues este tipo de software involucra aspectos didácticos, comunicativos, pedagógicos y ambientales que no siempre son fáciles de formalizar, condición esta que es fundamental en las metodologías (Parra Castrillón, 2011).

De acuerdo con Barajas, Muñoz y Álvarez (2007) se presentan varios problemas, que se puede englobar generalmente en los siguientes cuatro puntos:

1. Los expertos en contenidos no han sido provistos de herramientas sencillas e intuitivas que automaticen la producción masiva de contenidos.
2. Los productores de OA no están capacitados con los fundamentos teóricos que sustenten la estructuración o diseño de los objetos de aprendizaje, o la experiencia en el área de conocimiento en la que aplica el OA.
3. Para la elaboración de un OA no existe un proceso estructurado ni basado en IS para la producción de estos productos de software, que garantice la consistencia y estandarización de los procesos para aumentar la calidad de los productos.
4. No existe ningún modelo de madurez de procesos que garantice la calidad del proceso de producción de OA.

En síntesis, los proyectos de desarrollo de OA deberían considerar aspectos de pedagogía, tecnología y protocolización de procesos. Considerando esto, para su construcción de una manera estructurada, dentro de cánones de la IS.

A partir de la investigación bibliográfica, se presentan a continuación distintas metodologías de desarrollo adaptables a la construcción de OA.

4.2. Metodologías de desarrollo de la IS

4.2.1. Modelo de proceso RUP (Rational Unified Process)

El RUP es un proceso de desarrollo de software que está basado en componentes, es decir, el sistema se forma con componentes de software que se interconectan a través de interfaces bien definidas. Utiliza el Lenguaje de Modelado Unificado (UML), para preparar todos los esquemas de un sistema de software.

Las características distintivas del RUP se resumen en: Dirigido por casos de uso, Centrado

en la arquitectura, iterativo e incremental. (Jacobson, Booch y Rumbaugh, 2000).

Kruchten (2003) señala que los Casos de Uso son una técnica de captura de requisitos que fuerza a pensar en términos de importancia para el usuario y no sólo en términos de funciones que sería bueno contemplar. Un caso de uso describe la secuencia de interacciones que se producen entre el sistema y los actores del mismo para realizar una determinada función. Un actor es un papel que las personas (usuarios) o dispositivos juegan cuando interactúan con el software.

Según Jacobson, Booch y Rumbaugh (2000), en RUP los Casos de Uso no son sólo una herramienta para especificar los requisitos del sistema, sino que también guían su diseño, implementación y prueba. Los Casos de Uso constituyen un elemento integrador y una guía del trabajo. Además de utilizarlos para guiar el proceso; se presta especial atención al establecimiento temprano de una buena arquitectura que no se vea fuertemente impactada ante cambios posteriores durante la construcción y el mantenimiento.

El proceso tiene dos estructuras o dimensiones (Figura 4.2):

- Un eje horizontal que representa el tiempo y muestra los aspectos del ciclo de vida del proceso.
- El eje vertical representa los flujos de trabajo del proceso, los cuales agrupan actividades de acuerdo a su naturaleza.

La primera dimensión representa la parte dinámica del proceso, y está expresada en términos de ciclos, fases e iteraciones.

La segunda dimensión representa la parte estática del proceso, cómo se describe en términos de componentes, actividades, flujos de trabajo, artefactos y actores.

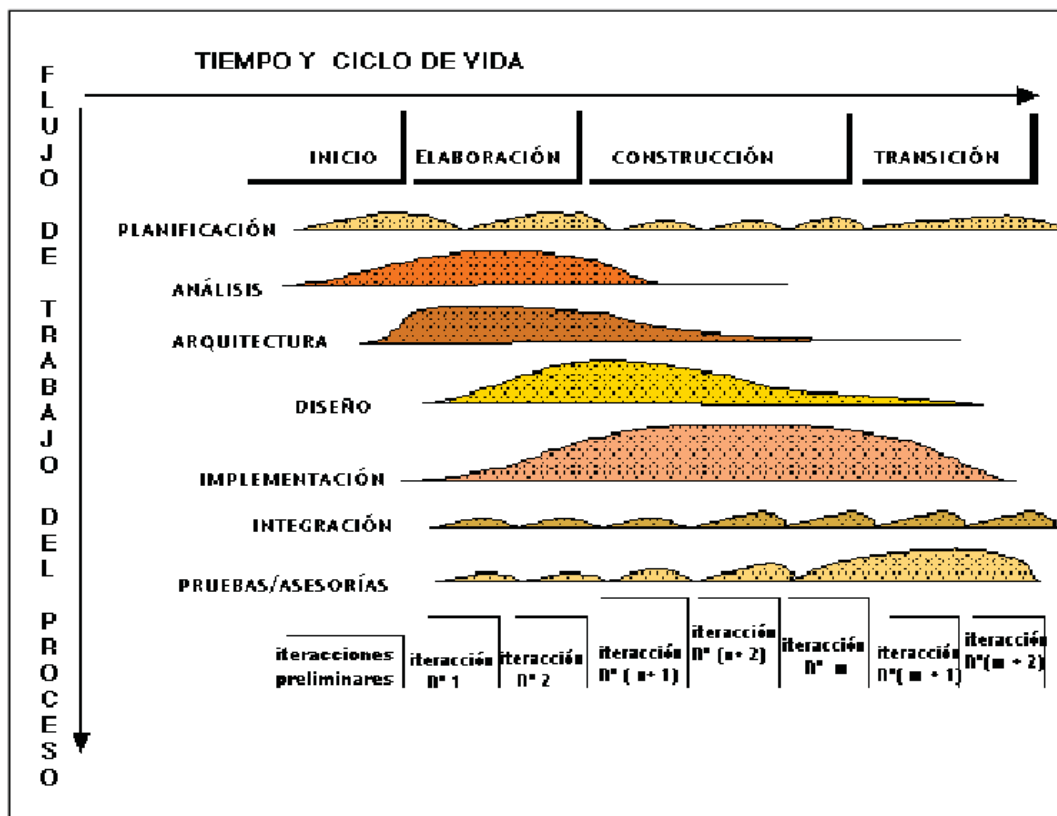


Figura 4.2. Dimensiones del proceso RUP. Fuente: Kruchten (1996).

Como señalan Rodríguez y Cardona (2005) aunque los OA son artefactos de software, se escapan un poco del mismo, pues su construcción involucra múltiples facetas: educacionales, psicológicas, sociológicas, gráficas, colaborativas, entre otras. Es por ello que se debería plantear una metodología que tome de RUP las fortalezas necesarias para el desarrollo de proyectos de software y se incorporen condiciones adicionales que se requieren para construir OA con altos niveles de calidad sistémica. Para estos autores, este enfoque de producción permite integrar altos estándares de calidad y mejoramiento de producto.

En el caso particular de los OAs y del software educativo, existen investigaciones que han incluido la metodología RUP en su desarrollo:

- a) Díaz (2002) del Laboratorio de Información y Sistemas (LISI), Universidad Simón Bolívar: extensión de la Metodología RUP para el desarrollo de software educativo.
- b) Rodríguez y Cardona (2005) de la Universidad de Aguas Calientes, Mexico: Cursos on-line que incluyen OA.
- c) Medina y López (2006) de la Universidad Central de Venezuela: Metodología LOCOME para la construcción de OA.

A continuación se presenta un Modelo de Proceso RUP adaptado para software educativo. El mismo está sustentado en los conceptos de calidad total sistémica del modelo de Calidad del Producto (Ortega et al., 2000) y en el modelo de Calidad del Proceso de desarrollo (Pérez et al., 2001).

4.2.1.1. Modelo de proceso RUP para software educativo

Díaz (2002) plantea una propuesta de una metodología de desarrollo de software educativo bajo un enfoque de calidad sistémica el cual realizan una adaptación y extensión de la Metodología RUP. El control de la calidad de producto se sustenta en el Modelo Sistémico de Calidad (MOSCA).

Díaz (2002) establece un modelo de desarrollo de software educativo que integra la Metodología RUP y MOSCA con sus respectivas adaptaciones para software educativo, ya que estos modelos son originalmente formulados genéricamente.

Para adaptar el Modelo MOSCA, Díaz (2002) seleccionó un conjunto del total de las características, categorías y métricas de MOSCA que se ajustan a la evaluación de software educativo. Por último añadió un nuevo conjunto de medidas que involucran tanto un nuevo parámetro de medición (sub-características), como nuevas métricas que no están presentes en el modelo original.

MOSCA consta de cuatro niveles: dimensiones, categorías, características y métricas; con un total de 587 métricas. Además proporciona el algoritmo para evaluar la calidad sistémica. El algoritmo contempla tres fases:

Fase 1: Calidad del producto de software con un enfoque sistémico.

Fase 2: Calidad del proceso de desarrollo de software con un enfoque sistémico.

Fase 3: Integración de las mediciones de los submodelos de la calidad del producto y la calidad del proceso.

Por otra parte, las cuatro fases que especifica Díaz (2002) como adaptación de la metodología RUP contienen actividades agregadas para el desarrollo del software educativo. Dichas fases son las siguientes:

a) Fase de Comienzo o Inicio:

Esta fase está principalmente dirigida a la comprensión de los requerimientos y el alcance del esfuerzo de desarrollo.

Se analiza la problemática de manera que puedan detallarse las necesidades en el desarrollo del software educativo, y todos los requerimientos necesarios para especificar la

interfaz deseada en el software educativo.

Las actividades agregadas en esta fase según la metodología RUP para el desarrollo del software educativo son las siguientes:

- Análisis de las necesidades educativas y del entorno educativo.
- Estudio sobre las teorías de aprendizaje y diseño instruccional que definen el formato del programa.
- Listado de requerimientos pedagógicos relacionados con el contenido y la población estudiantil a la que va dirigida el programa.
- Revisión de los objetivos y contenidos del material educativo del programa.
- Encuadre de las áreas educativas que se van a desarrollar.
- Determinación de un diseño instruccional para el proyecto multimedia, incluyendo los instrumentos de evaluación del usuario sobre lo aprendido.
- Estudio sobre las pautas de diseño de interfaz adecuadas a la población estudiantil a la que va dirigida el programa.
- Determinación de los criterios de evaluación del software educativo basados en las características de funcionalidad, usabilidad y fiabilidad (Adaptación del modelo MOSCA)

b) Fase de Elaboración:

En esta fase se realiza principalmente el diseño e implementación. Se lleva a cabo por medio de una serie de iteraciones. Para cada iteración se seleccionan Casos de Uso, se refinan su análisis y diseño y se procede a su implementación y pruebas.

Las actividades que propone para este caso, según la Metodología RUP, son las siguientes:

- Refinar los modelos de diseño instruccional que se utilizan en el proyecto.
- Refinar los requerimientos de diseño gráfico y aspectos comunicacionales en base a las pautas pedagógicas establecidas.

c) Fase de Construcción:

En esta fase se realiza principalmente el diseño e implementación. Se lleva a cabo por medio de una serie de iteraciones. Para cada iteración se seleccionan Casos de Uso, se refinan su análisis y diseño y se procede a su implementación y pruebas.

Se agrega la siguiente actividad:

- Probar el diseño instruccional, comunicacional y gráfico, contra los criterios de evaluación previamente establecidos (se utiliza la adaptación de MOSCA).

d) Fase de Transición:

En esta fase lo importante es asegurar que el sistema posee el nivel adecuado de calidad para alcanzar los objetivos.

Se libera el sistema y se entrega al usuario para un uso real. Se incluyen tareas de marketing, empaquetado atractivo, instalación, configuración, entrenamiento, soporte, mantenimiento, etc.

La actividad agregada a la Metodología RUP que es la siguiente:

- Realizar la evaluación del producto por parte del docente y del estudiante utilizando la adaptación del modelo MOSCA

La metodología RUP es apropiada para proyectos grandes de largo plazo con equipos de desarrollo con numerosas personas que pueden estar dispersas geográficamente. Se trata de un proceso de implementación complejo, para el cual el número de personal especializado en el grupo de trabajo debe aumentar con respecto a otras metodologías.

En el caso de los OA consideramos que esta metodología no es la adecuada pues es muy compleja para proyectos y equipos pequeños. Se deben repartir 32 roles y generar muchos artefactos finales, los cuales pueden ser aprovechados en una reutilización de productos, modelos o procesos pero también significa un incremento de tiempos y costos.

4.2.2. Modelo de prototipos evolutivos con refinamientos sucesivos

El uso de prototipos se centra en la idea de ayudar a comprender los requisitos que plantea el usuario, sobre todo si este no tiene una idea muy acabada de lo que desea. También pueden utilizarse cuando el ingeniero de software tiene dudas acerca de la viabilidad de la solución pensada.

Esta versión temprana de lo que será el producto, con una funcionalidad reducida, en principio, podrá incrementarse paulatinamente a través de refinamientos sucesivos de las especificaciones del sistema, evolucionando hasta llegar al sistema final (Cataldi, 2006).

Según Pressman(2002), un prototipo es aquel que describe una interacción entre la máquina y el hombre, facilitando una comprensión de la forma en cómo se da esa interacción, dando también una visión de las funciones que el sistema puede contener para así lograr una función deseada al finalizar el desarrollo del mismo.

En el modelo incremental la entrega al cliente está centrada en un producto operacional

con cada incremento. Los primeros incrementos son versiones «incompletas» del producto final, pero proporcionan al usuario la funcionalidad que precisa y también una plataforma para la evaluación (Pressman, 2002)

En el caso particular de los OA o del software educativo, como afirma Cataldi (2006), la elección de este método puede estar fundamentada en la ventaja de la realización de los cambios en etapas tempranas y la posibilidad de construcción de varios prototipos evaluables durante el desarrollo, obteniéndose de este modo paralelamente una metodología integral también para el proceso de evaluación del programa.

Según García Álvarez (2006) este modelo resulta menos rígido que el modelo de cascada puesto que el cliente (en este caso los educadores) pueden ir haciendo los refinamientos o las aportaciones necesarias, inclusive pueden probar el primer prototipo con sus estudiantes además de realizar una breve evaluación con sus demás colegas, sin duda alguna que esto mejorará el diseño del software y de manera especial la parte comunicativa o educativa.

Existen investigaciones que han desarrollado productos con este modelo:

- a) Cataldi (2000 y 2006) de la Universidad de Buenos Aires. Refinamiento del Modelo de prototipos evolutivos basado en dos dominios: pedagógico y técnico para el desarrollo de software educativo.
- b) Barajas et. al. (2007): se propone el modelado del proceso en base a las siguientes metodologías, 1) el ciclo evolutivo de prototipos o 2) la metodología de Programación Extrema.
- c) Borrero Caldas et.al. (2010) de la Universidad del Valle (Colombia): desarrollo de OA.

A continuación se presenta la adaptación del Modelo de prototipos evolutivos con refinamientos sucesivos para software educativo de Cataldi (2002) del Laboratorio de Sistemas Operativos y Bases de Datos. Departamento de Computación. Facultad de Ingeniería de la Universidad de Buenos Aires

4.2.2.2. Modelo de prototipos evolutivos con refinamientos sucesivos para software educativo

La propuesta de Cataldi (2002) incrementa el nivel de detalle en el desarrollo de Software Educativo. El refinamiento consiste en un ciclo de vida que define todos los procesos y actividades a efectuar en cada etapa del desarrollo (Figura 4.3).

En el ciclo de vida de prototipo incremental se definen las siguientes etapas:

1. Factibilidad (FAC)

2. Definición de requisitos del sistema (RES)
3. Especificación de los requisitos del prototipo (REP)
4. Diseño del prototipo (DPR)
5. Diseño detallado el prototipo (DDP)
6. Desarrollo del prototipo (codificación) (DEP)
7. Implementación y prueba del prototipo (IPP)
8. Refinamiento iterativo de las especificaciones del prototipo (aumentando el objetivo y/o el alcance).Luego, se puede volver a la etapa 2 o continuar si se logró el objetivo y alcance deseados. (RIT)
9. Diseño del sistema final (DSF)
10. Implementación del sistema final (ISF)
11. Operación y mantenimiento (OPM)
12. Retiro (si corresponde) (RET)

A continuación se describen cada una de las etapas del ciclo de vida elegido que formarán parte de la matriz de actividades:

1. **Factibilidad (FAC):** definición del producto de software y su factibilidad desde la perspectiva de la relación costo –beneficio, como así las ventajas y desventajas respecto de otros productos.
2. **Requisitos del sistema (RES):** definición de las funcionalidades, interfaces y el tipo de diseño.
3. **Especificación de requisitos del prototipo (REP):** especificación de las funcionalidades, las interfaces y el rendimiento para el prototipo. Aquí se considerarán incrementos en porcentajes de la funcionalidad total del sistema.
4. **Diseño del prototipo (DPR):** ejecución del plan del prototipo. Análisis de las funciones y módulos del mismo.
5. **Diseño detallado del prototipo (DDP):** Se deben traducir los requisitos a una representación del software de manera que se obtenga la calidad requerida antes de que comience la codificación.
6. **Desarrollo del prototipo (codificación) (DEP):** codificación o diseño detallado en un lenguaje
7. **Implementación y prueba del prototipo (IPP):** Consiste en lograr un funcionamiento adecuado del producto software en el sistema informático.
8. **Refinamiento iterativo de las especificaciones del prototipo (RIT):** aumento de

la funcionalidad del sistema, para luego volver REP a fin de aumentar la funcionalidad del prototipo o continuar, si se logró el objetivo y alcance deseados.

9. **Diseño del sistema final (DSF):** ajustar las restricciones o condiciones finales e integrar los últimos módulos.
10. **Implementación del sistema final (ISF):** instalación y capacitación del personal.
11. **Operación y mantenimiento (OPM):** puesta en funcionamiento del sistema informático.
12. **Retiro (RET):** transición adecuada de las funciones realizadas para el producto y sus sucesores.

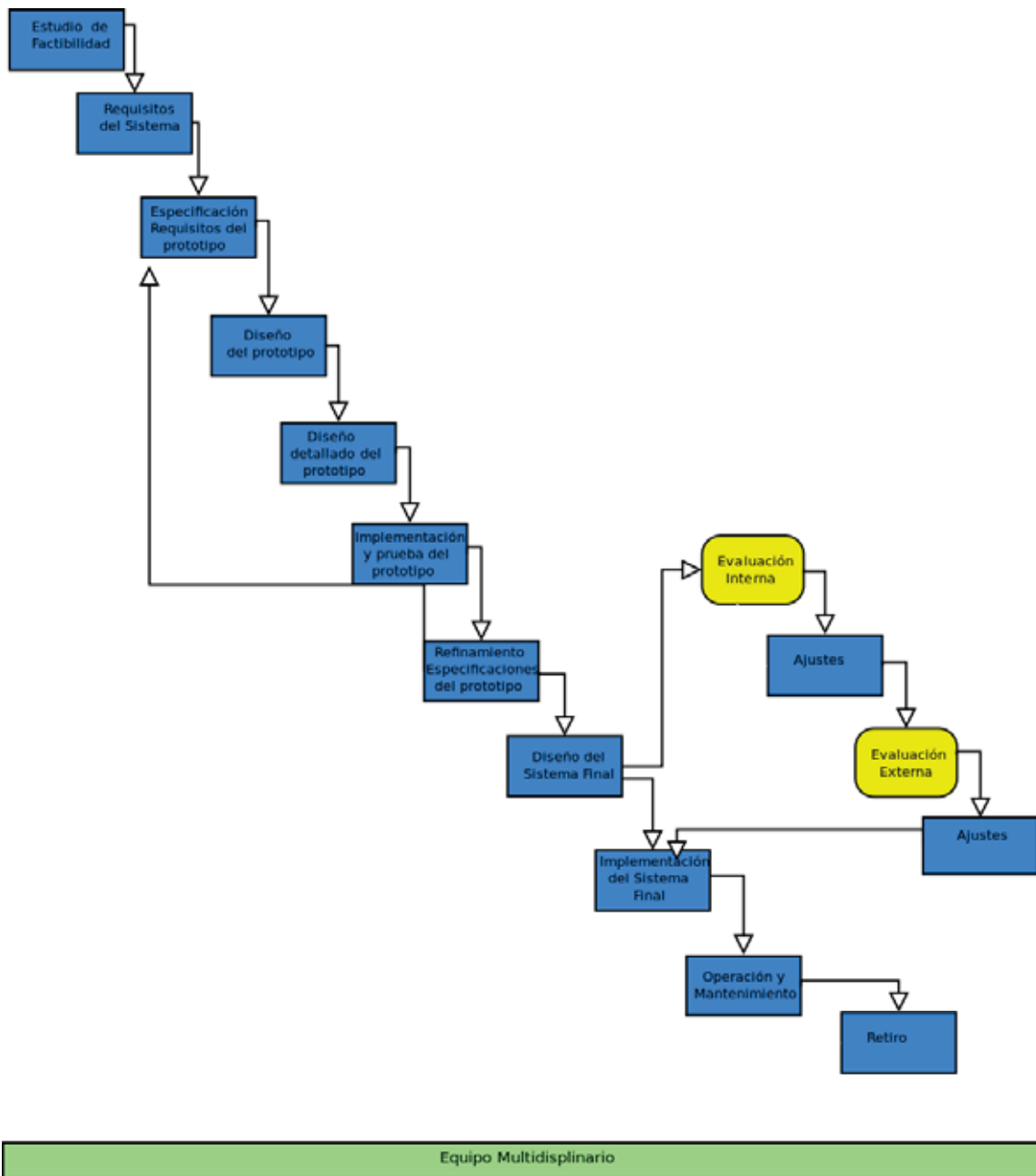


Figura 4.3: Esquema del ciclo de vida del prototipado evolutivo

Cataldi identificó que las diez etapas debían extenderse e incorporó procesos nuevos que contemplaron las necesidades pedagógicas-didácticas.

Los procesos nuevos que se incorporaron así como sus actividades se describen a continuación:

- Identificación de la necesidad educativa del software y selección de la teoría educativa.
- Análisis de los requisitos educativos. Definición de objetivos educativos, las características del grupo destinatario, los contenidos, las estrategias didácticas, las actividades mentales a desarrollar, el nivel de integración curricular, el tipo de uso del programa y el nivel de interactividad, los efectos motivadores, los posibles caminos pedagógicos, el tiempo y modo de uso del programa y el hardware asociado.
- Evaluación de los prototipos de software.
- Evaluación interna y externa del software: a partir de la evaluación se identifican los cambios y ajustes a realizar
- Documentación didáctica: planificación y elaboración de la guía e información didáctica.

La metodología extendida para la creación de software educativo desde una visión integradora propuesta por Cataldi, hace hincapié en que las metodologías propias de la Ingeniería del Software no toman en cuenta aspectos pedagógicos-didácticos del producto software a desarrollar, pocas metodologías toman en cuenta los aspectos de comunicación con el usuario que en principio son un componente elemental y básico en el desarrollo de software educativo (Cataldi, 2003).

Consideramos que esta propuesta no es la adecuada para el desarrollo de un OA pues no se enfoca a aplicaciones Web ni permite la reutilización de materiales educativos, características inherentes de un OA.

En cuanto a la evaluación de los prototipos está centrada en uno de los usuarios: el docente, lo cual no garantiza que el modelo se enfoque a las mejoras en el proceso de enseñanza y de aprendizaje.

4.2.3. Sistemas interactivos

Los sistemas interactivos se caracterizan por la importancia del diálogo con el usuario y la interfaz determina en gran medida la percepción e impresión que el usuario posee de una aplicación. En general, el usuario no está interesado en la estructura interna de una

aplicación, sino en cómo usarla, para él no existe nada más que la interfaz, no le importa cómo éstas se llevan a cabo, sólo está interesado en las funcionalidades que utiliza y no es consciente de aquellas que la interfaz le esconde. Tan sólo interacciona con el ordenador para poder realizar una serie de tareas y solamente está interesado en los resultados de las mismas, mostrados precisamente por dicha interfaz.

En ese sentido, Moran (1981) afirma que la interfaz de usuario de un sistema consiste en aquellos aspectos del sistema con los que el usuario entra en contacto, física, perceptiva o conceptualmente, mientras que los aspectos del sistema que están escondidos para el usuario se denominan implementación.

En la misma línea, Negroponte define la interfaz como el sitio donde los bits y las personas se encuentran (Negroponte, 1994) y Van der Veer (1990) quien define la interfaz como el conocimiento que los usuarios pueden y deberían tener para poder utilizar satisfactoriamente el sistema.

En particular en el caso de los OAs, éste es considerado un objeto interactivo cuando se construye no sólo con un contenido sino que además tiene la capacidad de registrar el progreso de los estudiantes y las diferentes interacciones que dicho usuario (estudiante) realiza sobre una unidad de contenido concreta. La interactividad se puede definir a partir del desarrollo de ejercicios, simulaciones, cuestionarios, diagramas, gráficos, diapositivas, tablas, exámenes, experimentos, etc.

La interactividad por otra parte, adquiere un valor agregado en el caso de los OA pues se constituye en un efectivo mediador (virtual) entre lo que se representa y el usuario que interpreta, o entre la teoría y el usuario, con dos posibles propósitos: que el usuario intervenga sobre lo representado (interactividad) y que infiera y deduzca la teoría que sustenta (efectividad); es decir, promueve su conocimiento significativo (Proyecto Canals).

La Interacción Persona-Ordenador (IPO) es una disciplina relacionada con el diseño, implementación y evaluación de sistemas informáticos interactivos para uso de seres humanos y con el estudio de los fenómenos más importantes con los que están relacionados (Hewett et.al, 1997).

Puesto que los OA, se consideran sistemas interactivos, coincidimos con Mor, Garreta y Galofréy (2007) en cuanto a que deberían diseñarse también considerando los conceptos y metodologías propios de la IPO. Se basa en un Proceso de Diseño Centrado en el Usuario (DCU) en donde el usuario tiene un grado de implicación en todos los puntos del desarrollo del sistema. Dicho diseño se explica en la subsección siguiente.

4.2.3.1. Diseño centrado en el usuario (DCU)

El DCU se caracteriza por asumir que todo el proceso de diseño y desarrollo debe estar conducido por el usuario, sus necesidades, características y objetivos. Centrar el diseño en los usuarios implica involucrar desde el comienzo a los usuarios en el proceso de desarrollo del sitio; conocer cómo son, qué necesitan, para qué usan el sitio; testar el sitio con los propios usuarios; investigar cómo reaccionan ante el diseño, cómo es su experiencia de uso; e innovar siempre con el objetivo claro de mejorar la experiencia del usuario (Hassan, Fernández y Iazza , 2004).

Coincidimos con Sanchez (2011) en cuanto a que la usabilidad es la calidad de los productos que se pretende obtener mediante el DCU; es decir, el objetivo principal del DCU es obtener productos más usables.

Aunque, tal como lo afirman Tuya, Ramos y Dolado (2007), durante mucho tiempo, en el desarrollo de sistemas interactivos se ha olvidado la importancia de la usabilidad de dichos sistemas, relegando ésta exclusivamente a actividades de evaluación del producto final. Estos autores sostienen que puede deberse por un lado, al hecho de basar el desarrollo de los sistemas fundamentalmente en la tecnología disponible y por otro, a la relación errónea que los desarrolladores han establecido entre usabilidad y apariencia de la interfaz de usuario en cuanto a sus características estéticas.

Otros estudios como los realizados por Ferré y Moreno (2004), Ferré (2003) y Granollers(2004) detallan cuales pueden ser los motivos por los cuales las organizaciones de desarrollo de software no apliquen ninguna aproximación metodológica que incluya la usabilidad en todo el ciclo de desarrollo de software:

- Los modelos propuestos, implican un cambio radical en el proceso, que los ingenieros de software no creen suficientemente justificado.
- Los modelos propuestos por la IU son complejos de entender y de aplicar.
- Los directivos de las organizaciones software creen que la usabilidad no está económicamente justificada.
- Existen diferencias de conceptos y terminología entre ambas disciplinas (IS e IU).
- No existe una integración adecuada de las actividades de usabilidad en el proceso de desarrollo de la Ingeniería del Software.

El término usabilidad se define en la norma ISO 9241-11 como “el grado en el que un producto puede ser utilizado por usuarios especificados para conseguir objetivos concretos

con efectividad, eficiencia y satisfacción, en un determinado contexto de uso” (ISO 9241, 1998). La usabilidad de los sistemas interactivos se trata en profundidad en el Capítulo 5.

La norma ISO 9241-11 explica cómo identificar la información que se necesita considerar en el momento de especificar o evaluar la usabilidad en términos de medidas de funcionamiento y de satisfacción del usuario.

El estándar ISO 13407 (1999) constituye un marco de referencia en el desarrollo de sistemas interactivos usables incorporando el DCU durante el ciclo de vida del desarrollo. En ese sentido, el estándar ISO 13407 internacional revela cuatro actividades de DCU (Figura 4.4),

- Entender y especificar el contexto de uso.
- Especificar los requisitos de usuario y de la organización.
- Producir soluciones de diseño.
- Evaluar los diseños en base a los requisitos.

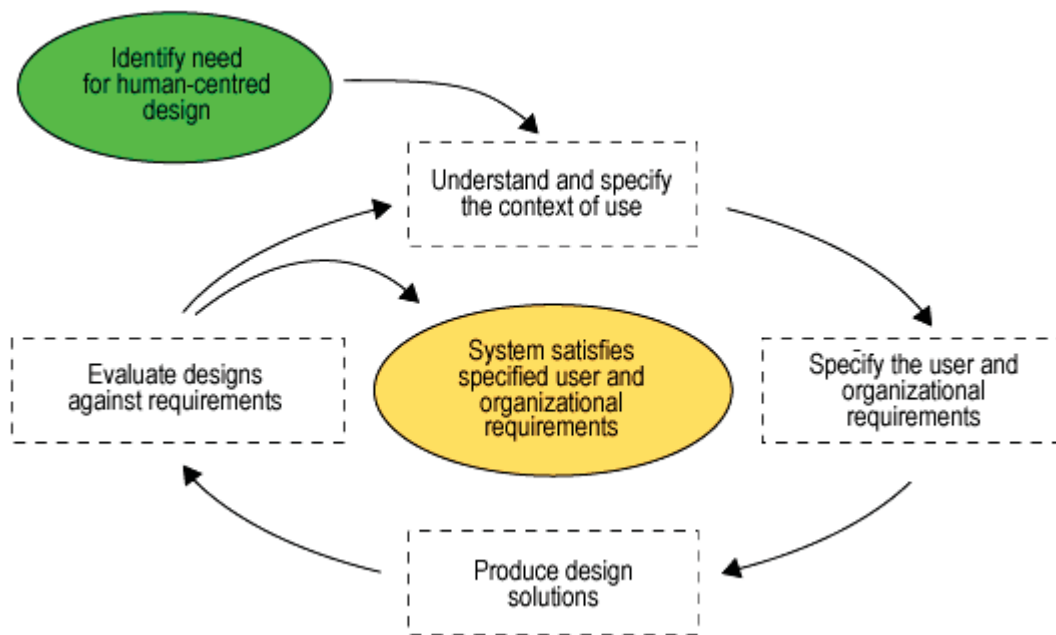


Figura 4.4. Proceso iterativo del DCU según ISO 13407. Fuente: Sergeev (n.d.).

Este proceso tiene como meta que el producto final responda a las necesidades y características del usuario, que en el caso del desarrollo de OA facilita a los estudiantes una experiencia de aprendizaje positiva.

Cabe destacar, como señala Mor, Garretay Galofréy (2007), que la aplicación de técnicas y métodos de DCU en entornos y contenidos de aprendizaje virtual como los OA, requiere de adaptaciones ya que los principales usuarios son estudiantes y consecuentemente tienen un objetivo concreto: aprender.

En ese sentido, el estándar ISO 13407 internacional revela cuatro actividades de DCU, que en el caso particular del diseño de OAs toman las siguientes particularidades:

- Comprender y especificar el contexto de uso: Se deben analizar las características de los implicados, las tareas que estos van a desarrollar en y con el OA en un entorno virtual de enseñanza aprendizaje (EVEA)
- Especificar los requerimientos del usuario y de la organización: Se establecen los objetivos del OA, se identifican los roles y perfiles de usuarios y la estructura y los contenidos de la aplicación.
- Producir soluciones de diseño: A partir de la información recabada en los pasos anteriores y los aportes de los equipos multidisciplinares, se construyen soluciones de diseño utilizando algún tipo de prototipado. Estos prototipos son evaluados por los usuarios, retroalimentando el proceso y modificando el diseño.
- Evaluar los diseños respecto de los requerimientos: Proporciona información que contribuye a mejorar el diseño, determina si se han alcanzado los objetivos especificados y verifica el uso a largo plazo del producto.

Recientemente, el ISO 13407 ha sido actualizado y renombrado como ISO 9241-210:2010 – Ergonomics of human-system interaction – Part 210: Human-centred design for interactive systems (ISO 9241-210 (2010) para adaptarlo a las tendencias actuales e integrarlo con otros estándares relacionados (Travis; 2011).

En general, las características del DCU que hemos visto anteriormente contrastan con las metodologías tradicionales de desarrollo (Pressman, 2002), en las que el diseño es una de las etapas dentro de un desarrollo lineal, donde los usuarios finales participan básicamente en la fase de requisitos (cuando lo hacen), y donde el desarrollo del proyecto tiene una componente predominantemente técnica, orientada a que el sistema cumpla los requisitos definidos en un principio.

En resumen, podemos decir que se han realizado muchos esfuerzos por hacer compatibles diferentes metodologías de desarrollo con los principios del DCU, resultando en propuestas interesantes pero muy específicas de la metodología en concreto, lo que hace que no puedan utilizarse, en la práctica, como referentes genéricos de lo que es el DCU (Sanchez, 2011)

En esta línea, se han planteado diferentes propuestas para el establecimiento de métodos, técnicas y herramientas con el objetivo de orientar a los desarrolladores sobre las actividades a seguir durante el proceso de desarrollo de software que garanticen un nivel de

usabilidad previamente establecido. En Ferré y Moreno (2004) se presenta un estudio de la integración de la usabilidad en el proceso de desarrollo. En Granollers (2004) se presenta un resumen de algunos de los Modelos de Procesos propuestos por la IU para el desarrollo de sistemas interactivos así como una nueva propuesta de Modelo de Proceso Centrado en el Usuario que integra la IS, la IPO y la accesibilidad en el contexto de equipos de desarrollo multidisciplinares. Kreitzberg (2008) presenta un marco de trabajo para la integración de actividades de DCU en las metodologías de desarrollo de software y proporciona un marco en el que aplicar las mejores prácticas de diseño.

En el caso particular de los OAs y del software educativo, existen investigaciones que han incluido el Diseño Centrado en el Usuario en su proceso de desarrollo como:

- a) Ferran, Guerrero-Roldan, Mor y Minguillon (2009) y Mor, M., Garreta, M, y Galofréy (2007) de la Universidad Oberta de Cataluña: Repositorios de OA.
- b) Churchill,& Hedberg (2008) de la University of Hong Kong) y Bradley et. al (2009) : OA para dispositivos móviles.
- c) Moasil, Domnariu, Rogozea (2009): Sitio web de Educación para la Salud
- d) Costa et al.(2009) : Couseware sobre Educación Ambiental
- e) Menéndez, Prieto y Zapata (2010): Plataforma LMS AGORA (Ayuda para la Gestión de Objetos Reusables de Aprendizaje) de la Universidad Autónoma de Yucatán.
- f) Santos et.al (2009) de la UNED (Universidad Nacional de Educación a Distancia): Diseño de navegación de un LMS.

El desarrollo de OA supone una mirada desde las problemáticas educativas estudiadas y estudiables, intervenidas e intervenibles por comunidades científicas y desde la acción de los usuarios sobre dichos objetos.

Desde la perspectiva de la IPO, el estudiante no se encuentra solo realizando su tarea con la computadora, sino que se encuentra en una organización social y para que ello sea posible existe un complejo proceso de construcción del OA en el que cada uno de estos componentes debe ser abordado con igual grado de implicación y no caer en el error frecuente de centrarse solamente en la parte tecnológica y obviar la parte humana.

Consideramos que la incorporación de los principios del DCU en la construcción de los OA con las adaptaciones pertinentes es la más adecuada ya que permitirá relevar elementos significativos en el diseño de estos objetos proporcionando a los estudiantes mejores experiencias a partir de un mayor grado de implicación de todos los actores del proceso de enseñanza y aprendizaje.

A continuación se presenta el Modelo de Proceso de la Ingeniería de la usabilidad y de la accesibilidad (MPIu+a) que es la culminación de una de las líneas de investigación prioritarias del grupo de investigación GRIHO de la Universitat de Lleida.

4.2.3.2. Modelo de proceso MPIu+a

La Figura 4.5., presenta las diferentes fases en las que se divide el modelo y la relación entre ellas.

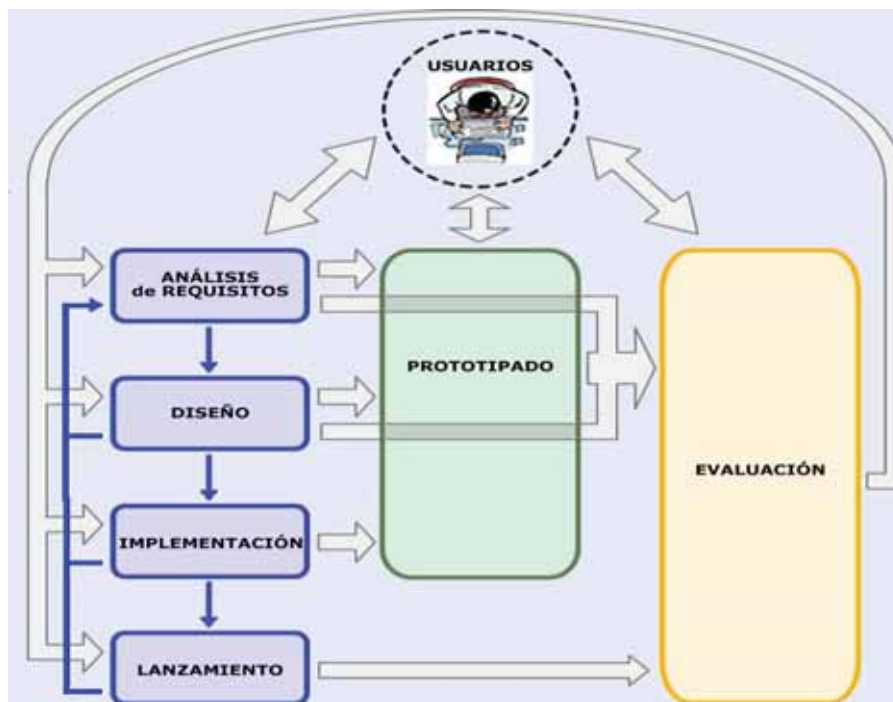


Figura 4.5. Esquema MPIu+a. Fuente:Granollers (2004).

a) Análisis de requisitos

La principal diferencia que aporta el modelo de proceso MPIu+a, consiste en el cambio de enfoque en la especificación del diseño de las interfaces que pasa a ser el aspecto principal, dejando para más adelante el resto de actividades de diseño.

Para poder realizar estas capturas de requisitos, MPIu+a contempla la realización de las siguientes actividades o aplicación de técnicas:

- Análisis etnográfico: observación del entorno (contexto) del usuario (relaciones entre persona-objeto, organización de tareas).
- Análisis de implicados: identificación de las personas involucradas y el grado de implicación.
- Clasificar a los usuarios: determinación de perfiles de usuario y roles.

- Análisis contextual de tareas: determinación de tareas que el sistema debe llevar a cabo.
- Objetos: Identificación de cualquier elemento (físico o conceptual) que pueda parecer, e intervenga de alguna manera durante la interacción del usuario con el sistema.
- Plataforma: Restricciones y ventajas en función de la plataforma escogida para el sistema.
- Objetivos: Especificación en base a necesidades.

b) Diseño

Esta fase se realizará de manera repetida tomando como referencia los requisitos obtenidos en la fase anterior, hasta que el diseño sea el deseado y éste pueda proporcionar la información necesaria al equipo de desarrollo para proceder a su codificación.

El diseño de la interacción se divide en dos actividades:

- Diseño de la actividad: Estudio de la funcionalidad y las tareas que permiten llevarla a cabo. Estudio de los modelos mentales de los usuarios.
- Diseño de la información: Relacionado directamente con los aspectos físicos de la interacción, el lenguaje, la ubicación de la información y los elementos, y la coherencia y consistencia de manera global en el sistema.

MPIu+a destaca principalmente dos maneras de abordar este diseño:

- Aproximación empírica: Basada en la experiencia del diseñador, estándares y recomendaciones internacionales. Para evaluar el resultado de su aplicación, el usuario a través de test de usabilidad, determina el nivel de satisfacción producido.
- Aproximación metodológica: aplicación de los pasos de una metodología de diseño fundamentada teóricamente.

c) Prototipado

Se centra en la realización de diversos prototipos con el objetivo de representar partes o la totalidad de un sistema, para poder evaluar las mismas y determinar mejoras, cambios y aciertos del sistema.

Dado que durante el proceso de prototipado se deben tener en cuenta dos aspectos muy importantes como: presentación del sistema y la interacción que éste proporcionará, se considerarán cada uno como una fase diferente.

Grannollers (2004) presenta un resumen de las diferentes técnicas de prototipado, incluyendo costos y tiempo de desarrollo que facilita la elección de la técnica según las necesidades.

d) Evaluación

Para MPIu+a la fase de evaluación constituye un factor crucial para garantizar un sistema interactivo usable y accesible. Mediante la aplicación de diferentes técnicas de evaluación se obtendrá la retroalimentación necesaria de los usuarios y/o evaluadores expertos para la mejora del sistema. Esta fase es aplicable en todo el proceso de desarrollo del sistema.

e) Implementación

El modelo de proceso MPIu+a no detalla específicamente esta fase debido al hecho que corresponde propiamente a un proceso de la IS.

Destaca la necesidad de seguir realizando evaluaciones de manera continua de prototipos y/o versiones preliminares del sistema final, en las que la participación de los usuarios e implicados esté presente de manera activa.

f) Lanzamiento

Granollers (2004) destaca que el éxito del producto dependerá de dos factores:

- La satisfacción y comodidad del usuario con el sistema (relacionado con la usabilidad y accesibilidad del sistema).
- Obtención de los resultados esperados por los responsables del proyecto (relacionado con la funcionalidad ofrecida por el sistema).

El concepto más importante en esta fase es el feedback del usuario. Instalado el sistema, se establece un período de prueba en el que los usuarios finales se familiarizarán con el sistema, y pueden presentar las dudas, mejoras, quejas, etc. A continuación se deberán realizar las correcciones necesarias en función de los comentarios obtenidos y que corresponden con las especificaciones iniciales del producto, favoreciendo de esta manera la calidad del producto y la satisfacción del usuario.

CAPITULO 5. PARÁMETROS DE CALIDAD DE UN OBJETO DE APRENDIZAJE (OA)

En este capítulo en la sección 5.1, se presenta una introducción sobre la perspectiva de calidad de un OA que se aborda en esta tesis. A continuación, en la sección 5.2, se introducen los conceptos de calidad de la IA en donde la usabilidad juega un papel importante. Luego, en la sección 5.3 se expone la visión de la tesis en cuanto a los criterios de evaluación de calidad de un OA. Finalmente en la sección 5.4 teniendo en cuenta las actividades necesarias para la evaluación de la calidad del OA, se presentan una clasificación de los métodos de evaluación de la usabilidad y una descripción de los mismos.

5.1. Introducción

Existen numerosas definiciones de los OAS que han generado diversas controversias en el campo. En este trabajo como destacamos en el Capítulo 2, adoptamos la siguiente definición:

La mínima estructura independiente que contiene un objetivo, un contenido, una actividad de aprendizaje, un metadato y un mecanismo de evaluación, el cual puede ser desarrollado con tecnologías de infocomunicación de manera de posibilitar su reutilización, interoperabilidad, accesibilidad y duración en el tiempo.

Dicha definición aporta el marco conceptual para el desarrollo y evaluación de los OAs.

La discusión acerca de la calidad de los OAs también ha tenido diversos enfoques a lo largo del tiempo. En ese camino encontramos que la mayoría de la investigaciones hasta 2005 se focaliza principalmente en discusiones acerca de la “calidad del metadato” (probabilidad de que un recurso pueda ser o no encontrado) relacionado a la concepción y definición técnica inicial de los OA.

Coincidimos con Leal Fonseca (2009) en que el proceso de aseguramiento de la calidad debería estar compuesto por tres diferentes niveles: uno en el proceso mismo del desarrollo, que corresponde a una concepción de la calidad más ligada al desarrollo de software, el segundo mediante la valoración de los materiales por comunidades

profesionales (expertos en contenido, pedagogos, diseñadores gráficos e ingenieros de sistemas, entre otros) , tiene que ver más con una concepción multidisciplinar de la calidad y el tercero corresponde a los usuarios finales de los OA, valoración desde la pertinencia y las posibilidades reales de uso que encuentren.

Durante la conferencia Open Education 2009: Crossing the chasm, Wiley (2009) realizó una presentación relacionada con la evaluación de recursos educativos abiertos, en la cual señaló lo siguiente:

La calidad no es una cualidad independiente de un Recurso Educativo Abierto (OER). De la misma manera que un problema de matemáticas es más fácil para una persona y más difícil para otra, un OER será de alta calidad, o muy útil para una persona, menos para otra, no tanto para una tercera... La idea de calidad no significa nada hasta que el OER entra en contacto con una persona que está tratando de aprender algo a partir de su uso. Es en ese momento en el que la calidad ‘ocurre’.

En síntesis, y como se destacó en el Capítulo 4, subsección 4.2.3.1 los OA deberían diseñarse también considerando los conceptos y metodologías propios de la Interacción Persona-Ordenador (IPO), permitiendo relevar elementos significativos con el propósito de proporcionar a los estudiantes mejores experiencias a partir de un mayor grado de implicación de todos los actores del procesos de enseñanza y aprendizaje.

5.2. La calidad en la Ingeniería de Software

Para que un proyecto software tenga éxito, es necesario que el resultado cuente con la calidad esperada por el cliente o los usuarios.

Pressman (2002) define la calidad del software como: “la concordancia con los requerimientos funcionales y de rendimiento explícitamente establecidos, con los estándares de desarrollo explícitamente documentados y con las características implícitas que se espera de todo software desarrollado profesionalmente” (p.135).

Por su parte, Sommerville (2005) proporciona otra definición simple pero precisa: “Calidad significa que un producto debe cumplir con sus especificaciones”

La norma ISO/IEC 9126 (2001) define el concepto de calidad de la siguiente manera: “El conjunto total de características de una entidad (producto, proceso o servicio) que le confieren la capacidad de satisfacer las necesidades establecidas y las necesidades

implícitas”

Las características definidas en dicho estándar sirven para las especificaciones de los requerimientos funcionales y también para los no funcionales (calidad interna y externa), tanto desde el punto de vista del cliente como del usuario (calidad en el uso).

La parte primera de dicho estándar define la calidad interna y externa de una aplicación basada en los atributos de funcionalidad, fiabilidad, usabilidad, eficiencia, mantenimiento y portabilidad y la calidad en el uso basada en las características de efectividad, productividad seguridad y satisfacción.

En particular, todos los atributos de calidad del software se aplican también a las aplicaciones WEB. Sin embargo, Pressman (2002), Olsina et.al. (1999) destacan que las características más relevantes -usabilidad, funcionalidad, fiabilidad, eficiencia y capacidad de mantenimiento- proporcionan una base útil para evaluar su calidad.

En el caso del software, muchas veces no está definido con precisión el dominio en el que va a ser utilizado, construyéndose de esta forma sistemas software con un nivel de calidad insuficiente para el dominio de aplicación.

Así, para evaluar la calidad del software es preciso comprender el propósito para el cual el sistema se va a usar.

La definición de usabilidad conforme según la norma ISO 9241-11(1998): “El grado en el cual un producto puede ser usado por unos usuarios específicos para alcanzar ciertas metas especificadas con efectividad, eficiencia y satisfacción en un contexto de uso especificado”.

La definición conforme a la norma ISO/IEC 9126-1 11 (2001): “Capacidad de un producto software de ser entendido, aprendido, usado y atractivo para el usuario, cuando es usado bajo unas condiciones específicas”.

La frase "cuando se utiliza bajo condiciones especificadas" (equivalente a "contexto de uso" en la norma ISO 9241-11 se añadió para dejar claro que un producto no tiene ninguna utilidad intrínseca, sólo una capacidad para ser utilizado en un contexto particular.

La norma reconoce que la usabilidad juega dos papeles (Bevan, 1999): una actividad de software de diseño detallado (que implica la definición de la usabilidad), y un objetivo general que el software satisface las necesidades del usuario (similar a la norma ISO 9241-11 , concepto de la facilidad de uso). ISO / IEC 9126-11 utiliza el término "calidad en uso" para este objetivo general: capacidad del producto de software para permitir a los usuarios especificados para alcanzar las metas especificadas con efectividad, productividad, seguridad y satisfacción en contextos de uso específicas. Es el efecto combinado de las seis

categorías de calidad del software, cuando el producto está en uso (<http://www.usabilitynet.org/>).

5.3. La calidad de un recurso digital educativo

Si pensamos en los OAs como recursos digitales que se diseñan mediante editores de páginas web, se podrían rescatar criterios de evaluación de calidad de sitios web en donde el concepto de usabilidad juega un papel importante. Si pensamos en los OAs como recursos pedagógicos, los criterios de evaluación de calidad deben referirse al destinatario, al contenido y los objetivos específicos entre otros aspectos.

Conciliando ambos criterios y extendiendo los atributos definidos por Nielsen (1993), denominamos “**usabilidad pedagógica**” a la facilidad de aprendizaje, eficiencia de uso pedagógico y la satisfacción con las que las personas son capaces de realizar sus tareas gracias al uso del producto con el que está interactuando. Entendemos eficiencia de uso pedagógico como la capacidad de propiciar aprendizajes significativos mediante interacciones generadas en la Zona de Desarrollo Próximo (ZDP).

Coincidimos con del Río Lugo (1999) en que la ZDP, es vista como un proceso de apropiación instrumental, un espacio estratégico para el desenvolvimiento del desarrollo humano en donde el agente promotor de desarrollo puede no ser necesariamente una persona. El papel de las herramientas culturales (un libro, la computadora, un software, la música, un video, etc.) pueden funcionar como agentes de desarrollo.

Desde esa perspectiva, tomamos la expresión acuñada por Bruner (1988), “amplificadores de la mente” para hacer referencia al software diseñado bajo los parámetros de usabilidad ya que en palabras de Velázquez y Sosa (2009, p.11): “La fuerza del potencial cognitivo radica en que la usabilidad exige que los sistemas se adapten a los usuarios y no a la inversa” y en el caso de los OA estos usuarios son estudiantes y consecuentemente tienen un objetivo principal y concreto: aprender.

5.4. Métodos de evaluación de usabilidad

5.4.1. Clasificación de los métodos de evaluación de usabilidad

Existen una amplia variedad de métodos de evaluación que utilizan diferentes medios y técnicas e intentan medir diferentes aspectos. Siguiendo la clasificación de Cañas,

Granollers y Lorés Vidal, (2005) en la Figura 5.1 observamos que los métodos pueden clasificarse de varias maneras

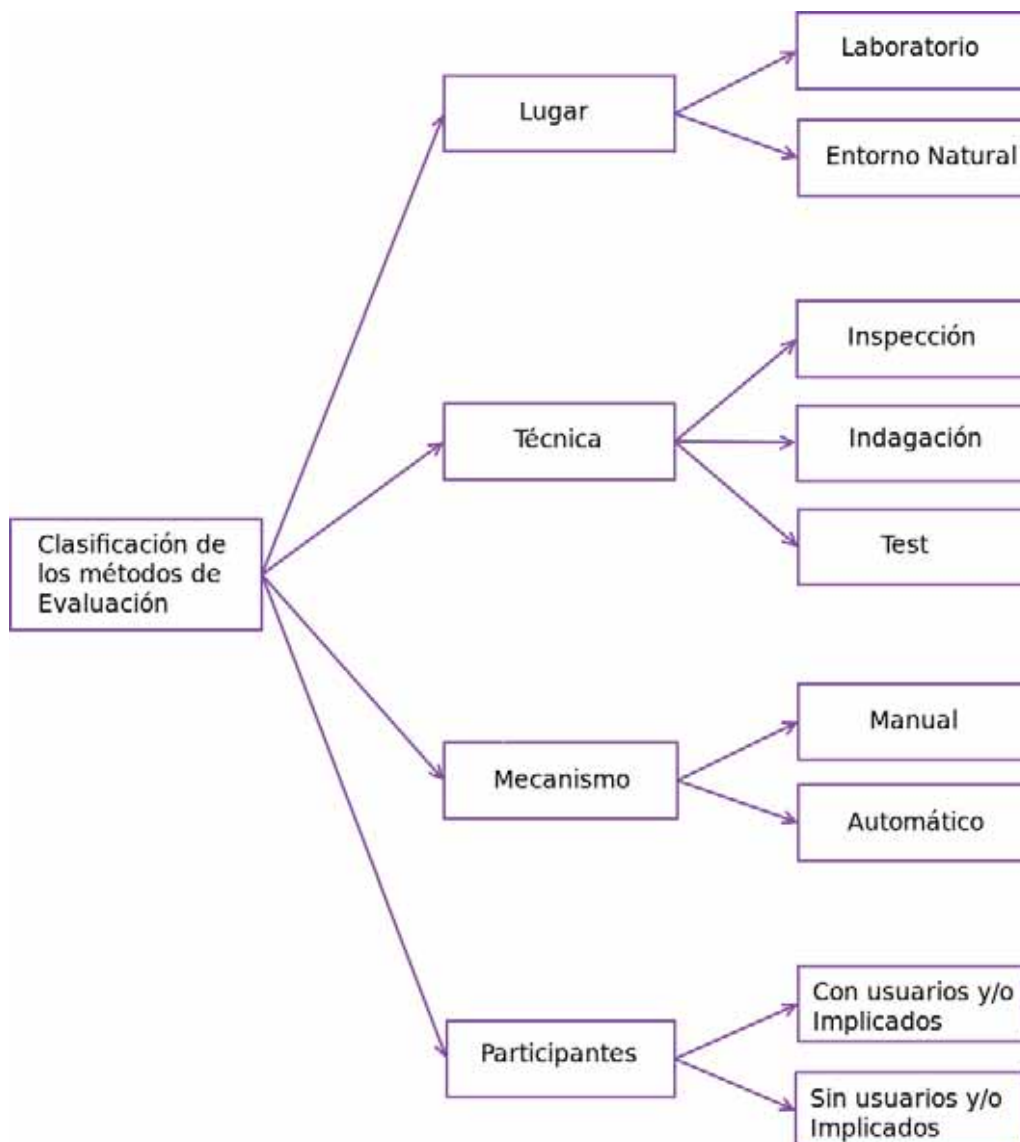


Figura 5.1. Clasificación de los Métodos de Evaluación. Fuente:Cañas, Granollers y Lorés Vidal, (2005).

1. Lugar de realización

- Laboratorio: el objetivo es comprobar aspectos sin que precise la ayuda de los usuarios, la realiza el evaluador y puede además traer usuarios para un estudio particular.
- Entorno natural: se produce cuando el evaluador realiza su trabajo en el lugar natural donde se realiza la escena, el lugar de interacción habitual del usuario.

2. Técnica de comprobación

- Métodos de inspección: los expertos, conocidos como evaluadores examinan (inspeccionan) aspectos de la interfaz del sistema relacionados con la usabilidad y la accesibilidad que la misma ofrece a sus usuarios.

Se comienzan a popularizar en las empresas productoras de software, pues permiten identificar, clasificar y contabilizar un gran número de errores potenciales de usabilidad a precio relativamente bajo (Bias & Mayhew, 2005), Nielsen, & Mack (1994), siendo el hecho de no utilizar usuarios uno de los factores que más contribuyen a dicha reducción económica.

- Métodos de indagación: permiten descubrir y aprender para generar ideas de diseño, especialmente para obtener información de usabilidad sobre un producto que se desea producir (Hom, 1998). Cuanto más se conozca a la audiencia, más adaptado será el diseño y más satisfactoria la experiencia del usuario final (Hassan, Fernández & Iazza, 2004).
- Métodos de test: usuarios representativos trabajan en tareas concretas utilizando el sistema (o el prototipo) y los evaluadores utilizan los resultados para ver cómo la interfaz de usuario da soporte a los usuarios con sus tareas.

3. Mecanismo

- Automáticos: disponen de mecanismos (hardware y/o software) que facilitan la comprobación de los aspectos a validar. Los métodos automáticos resultan altamente eficientes puesto que se realizan muy rápidamente y los resultados siempre provienen de los mismos parámetros, sin apreciaciones subjetivas.

El principal problema es que se basan en sistemas ya finalizados (o versiones preliminares de lanzamiento) y no pueden utilizarse en etapas tempranas del ciclo de vida del desarrollo.

Una de las características que hacen más atractivos a estos métodos es la posibilidad de realizar tests remotamente (Bartek & Cheatham, 2003), (Dray & Siegel, 2004) a través de la Web.

- Manuales: consumen más recursos: tiempo, evaluadores, etc., pero pueden evaluar aspectos que se salen del estándar y pueden realizarse en cualquier etapa del desarrollo y con cualquier tipo de prototipo.

4. Participantes:

- a. Con usuarios y/o implicados: resultan muy positivos por implicar la participación de usuarios en el proceso de diseño pero dificultan, en parte, el proceso debido a que no siempre resulta fácil reclutar usuarios para este tipo de actividades.
- b. Sin usuarios y/o implicados: los realizan sólo expertos evaluadores con la ayuda de guiones, pautas o documentos que permiten hacer un seguimiento de los que han hecho los usuarios mientras utilizaban el sistema. Se realizan con mayor velocidad pero carecen de la valiosa información que evaluar con dichos usuarios aporta al desarrollo del sistema. Permiten detectar errores funcionales, de consistencia, etc.

5.4.2. Métodos de evaluación de la usabilidad

Basándonos en la clasificación propuesta por el tipo de evaluación vista anteriormente (inspección, indagación y test) veremos en esta sección los métodos más destacados de cada una de las tres categorías (Figura 5.2).

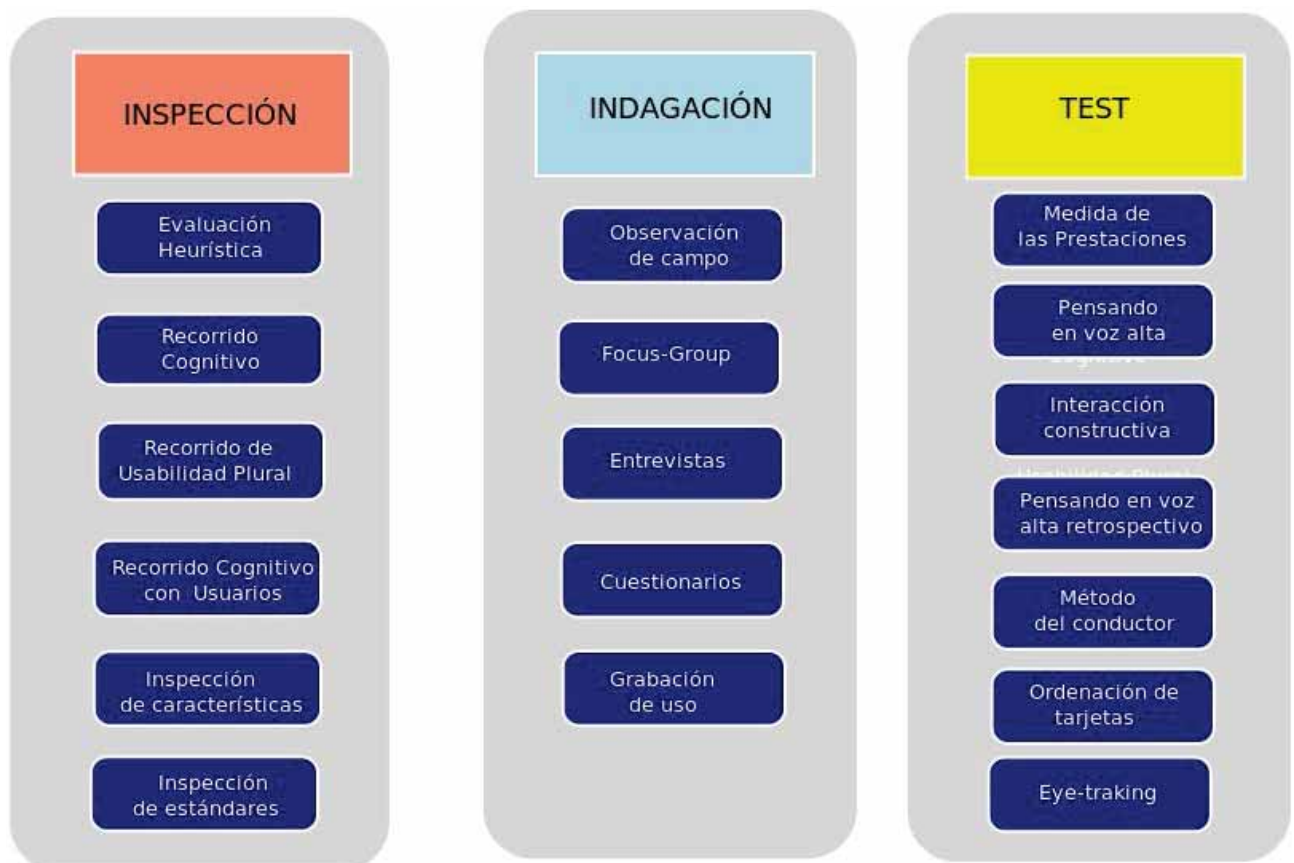


Figura 5.2. Métodos de evaluación de usabilidad

1. Inspección

a) Evaluación Heurística

El método fue desarrollado por Nielsen (Nielsen & Mack, 1994) y Molich (Molich & Nielsen, 1990) y consiste en analizar la conformidad de la interfaz con unos principios reconocidos de usabilidad (la "heurística") mediante la inspección de varios evaluadores expertos.

Para aplicar este método (Nielsen, 1994a), un conjunto de evaluadores (expertos en usabilidad) contrasta y valida individualmente las "10 reglas heurísticas de usabilidad". Tras las revisiones individuales los resultados son puestos en común y debatidos en una reunión entre los evaluadores y el responsable de la evaluación (denominado observador), quienes generan el informe final de la evaluación.



Según Nielsen (1994b), este método detecta aproximadamente un 42% de los problemas graves de diseño y un 32% de los problemas menores, dependiendo del número de evaluadores que revisen el sitio. Sin embargo, coincidimos con González, Lores y Pascual (2002) en que estas afirmaciones no pueden sostenerse en la actualidad, ya que la Ingeniería de la Usabilidad (IU) ha avanzado en la última década con lo que el contexto actual es más favorable: más conocimiento sobre la metodología, evaluadores más capacitados, más fuentes bibliográficas, mayor madurez científica y metodológica en la disciplina de la Interacción Persona-Ordenador (IPO).

b) Recorridos

b-1) El Recorrido Cognitivo (Cognitive Walkthrough) se centra en evaluar en un diseño su facilidad de aprendizaje, básicamente por exploración y está inspirado en la observación que muchos usuarios prefieren aprender software a base de explorar las posibilidades que este ofrece. Combina recorrido de software con un modelo cognitivo de aprendizaje por exploración. Es un método de revisión dónde los evaluadores especialistas construyen los escenarios de tarea de una especificación o prototipo inicial y después asumen el rol del usuario trabajando con esa interfaz (Lewis & Rieman, 1993), Blackmon (2004).

En todos los recorridos se realiza una revisión detallada de todas las acciones asociadas a la consecución de una o más tareas que el usuario debe poder satisfacer con el uso del sistema.

Según Granollers, Perdrix y Lorés (2004) la gran virtud de esta técnica es que al estar tan enfocada a la resolución de tareas concretas se consigue detectar un elevado número de

problemas, inconsistencias y mejoras.

Presenta una serie de problemas derivados de la ausencia de usuarios: siempre habrá aspectos que sólo pondrán de manifiesto los verdaderos interesados, el evaluador debe fiarse en la descripción que se le informa sobre los usuarios para interpretar si las tareas son adecuadas o no.

b-2) El recorrido pluralista (Pluralistic Walkthrough), se define como una reunión de grupos de usuarios, desarrolladores y expertos en factores humanos que discuten sobre los escenarios que ofrece o debe ofrecer un artefacto a la hora de posibilitar la realización de las tareas para las que está pensado (Bias,1994), (Riihiahho,2002).

b-3) El recorrido cognitivo con usuarios (Granollers, Perdrix y Lorés, 2004) se centra en evaluar en un diseño su facilidad de aprendizaje, básicamente por exploración y está motivado por la observación que muchos usuarios prefieren aprender software a base de explorar sus posibilidades (Wharton et. al.,1994).

Se debe hacer cuidadosamente ya cada persona tiene puntos de vista particulares e ideas propias. Así que en determinadas ocasiones las respuestas que un usuario puede dar serán menos



buenas que las de los expertos. El experto en evaluar interfaces de usuario puede conocer lo que los usuarios están pensando mejor que ellos mismos (Dix et. al, 2003).

c) Inspección de características

Analiza únicamente un conjunto de características determinadas del producto, proporcionándose escenarios de usuario para el resultado final a obtener del uso del producto.

Un experto en usabilidad analiza la interfaz de usuario de cada una de las páginas o secciones a las que afecta el módulo común, y la forma en que lo acceden. En una reunión de especificaciones con los distintos equipos de desarrollo se acuerdan las características definitivas del módulo (Hom, 1998).



Se utiliza en las fases intermedias del desarrollo, en las que ya son conocidas las funciones que deben estar implementadas en el producto.

d) Inspección de estándares

El experto realiza una inspección exhaustiva de la interfaz para comprobar que cumple en todo momento y globalmente todos los puntos definidos en el estándar establecido (Wixon et. al., 1994).

Lo más efectivo es realizarlo a partir de prototipos software y especialmente con una primera versión del sistema final donde estén implementadas las partes que deben confrontarse con el estándar (Granollers, 2004).

2. Indagación

a) Observación de campo (indagación contextual): su objetivo es comprender cómo los usuarios de los sistemas interactivos realizan sus tareas y más concretamente conocer todas las acciones que éstos realizan durante la realización de las mismas (Nielsen, 1993).

Utiliza una perspectiva naturalista que confía en el material de primera mano trazado por la experiencia de un usuario en su escenario de trabajo, en lugar de un escenario artificial o experimental (Rosenbaum, 2002).

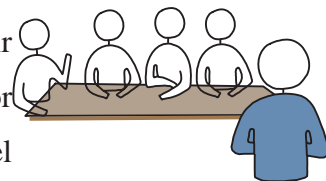
Un estudio de estas características aporta datos cualitativos que deben organizarse, comprenderse e interpretarse en fases tempranas, anteriores al proceso de diseño del producto (Hassan y Ortega, 2009), es decir para poder recoger requisitos necesarios para el diseño del sistema (Granollers, 2004).

b) Grupo de Discusión Dirigido (Focus Group)

El Focus Group (Nielsen, 1993) o grupo de discusión dirigido es una técnica de recopilación de datos donde se reúnen de 6 a 9 personas

(generalmente usuarios y también implicados) para discutir

aspectos relacionados con el sistema. En ellos un evaluador experto en usabilidad y/o accesibilidad (dependiendo del



objetivo de la evaluación) realiza la función de moderador. Éste preparará previamente la lista de aspectos a discutir y se encargará de recoger la información que necesita de la discusión.

Este método se utiliza a menudo en las primeras etapas de planificación de productos y la recopilación de requisitos para obtener información sobre los usuarios, productos, conceptos, prototipos, tareas, estrategias y entornos. También se puede utilizar para obtener un consenso sobre temas específicos (<http://www.usabilitybook.org/methods/p866>).

c) Entrevistas.

Las entrevistas pueden ser efectivas para una evaluación profunda. Son una poderosa herramienta cualitativa, pero no para evaluar la usabilidad de un diseño, sino para descubrir deseos, motivaciones, valores y experiencias de nuestros usuarios (Kuniavsky, 2003).



Suele realizarse una vez que el sistema ya ha sido puesto en marcha, el principal objetivo es captar la satisfacción del cliente o usuario con el producto (Hom, 1998).

d) Cuestionarios.

Un cuestionario es una lista de cuestiones o preguntas planteadas sobre algún tema con la finalidad de que alguien las responda. El cuestionario normalmente se distribuye en formato escrito y las preguntas plantean aspectos relacionados con el sistema o la aplicación a evaluar.

Las preguntas se elaborarán de manera que permitan obtener la información que precisemos sobre el sistema, la inmediatez en la interacción de los usuarios con el mismo y la percepción subjetiva de estos sobre la experiencia de su manejo (Hom, 1998).

d) Grabación del uso (análisis de logs)

Se basa en "grabar" o "recoger" todas las actividades realizadas por el usuario con el sistema para su posterior análisis. Es útil porque muestra cómo los usuarios realizan su trabajo real y porque es fácil recoger automáticamente datos de una gran cantidad de usuarios que trabajan bajo diversas circunstancias (Paganelli & Paternò, 2002).

Un registro de logs, contendrá típicamente, estadísticas sobre la frecuencia con la que cada usuario ha utilizado cada característica en el programa, información relativa a quien y cuando visita una página web determinada, la frecuencia con que los diversos eventos de interés (tales como mensajes de error) han ocurrido.

Su principal fortaleza es que, al contrario que otras técnicas cuantitativas, no se basan en muestras, sino en la monitorización del total de los usuarios que están haciendo uso del sistema (Hassan y Ortega, 2009).



Rovira (2007) afirma que existen diversas formas de utilizar los datos como: analizar dónde hacen clic los usuarios; comparar el número de abandonos y éxitos de una tarea entre dos páginas con la misma función pero diferente diseño; detectar en qué

campo de un formulario se produce mayor número de abandonos; o analizar las rutas de navegación que siguen los usuarios.

Estos informes se pueden utilizar para optimizar características usadas frecuentemente y para identificar las características que se no utilizan o se utilizan raramente. La estadística que muestra la frecuencia de las diversas situaciones de error y el uso de la ayuda se puede utilizar para mejorar la usabilidad del sistema (Granollers, 2004).

3. Test

a) Medida de las prestaciones

Está basado en la toma de medidas objetivas (rendimiento) u otro tipo de aspecto subjetivo que afecte a la usabilidad del sistema. Es necesario disponer del sistema ya implementado o de un prototipo que permita evaluar estos aspectos.

Las medidas de rendimiento objetivas son cuantitativas. Se pueden contar las acciones y los comportamientos. La mayor parte de las medidas de rendimiento requieren observaciones cuidadosas (Abascal et. al, 2006).

Algunos ejemplos de medidas de rendimiento son: tiempo para completar una tarea, tiempo consumido en menús de navegación, tiempo invertido en recuperarse errores, número de opciones de menú erróneo, etc.

Ejemplos de medidas subjetivas son: reflexiones y comentarios, percepciones, preferencias, etc.

Luego se analiza la información obtenida considerando los datos cualitativos y cuantitativos de los participantes con las observaciones propias y los comentarios del usuario.

b) Pensando en voz alta (thinking aloud).

Consiste en solicitar al usuario que exprese verbalmente qué está pensando, qué no entiende, por qué lleva a cabo una acción o duda mientras que interacciona con el sistema o un prototipo del mismo (Nielsen,1993).

Resulta altamente eficaz para capturar aspectos relacionados con las actividades cognitivas de los usuarios potenciales del sistema evaluado.



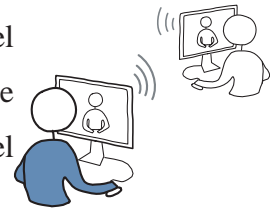
Sin embargo, es importante observar que con esta técnica el usuario no está ejecutando una tarea, sino más bien dos: realizando la tarea y verbalizando sus acciones, lo cual puede distorsionar el resultado de la investigación (Hassan y Ortega, 2009).

c) Interacción constructiva (aprendizaje por co-descubrimiento)

Es una derivación del pensando en voz alta pero con dos usuarios realizando conjuntamente cada test del sistema (O'Malley, Draper & Riley, 1984).

En este caso, la situación de la prueba es mucho más natural que el “thinking aloud” pues las personas normalmente verbalizan cuando tratan de resolver un problema conjuntamente y además hacen muchos más comentarios (aunque los usuarios pueden tener diferentes estrategias de aprendizaje) (Abascal et. al, 2006).

Las dificultades de usabilidad se suceden mucho antes, ya que el usuario se siente menos desamparado y a partir del intercambio de comentarios se obtiene más información sobre la percepción del interfaz (Hom, 1998).



Este tipo de prueba está especialmente indicada para evaluación de software en el trabajo cooperativo o colaborativo, groupware (García y Sicilia, 2006).

d) Pensando en voz alta retrospectivo

El usuario trabaja con la interfaz y sus acciones son grabadas en formato video. Una vez finalizada, expresa verbalmente cómo recuerda que ha sido su proceso interactivo (García y Sicilia, 2006). Es posible de esta manera conseguir información cuidadosa y sin problemas de tiempo (Wright & Monk, 1989).

e) Método del Conductor (coaching method)

Durante el test, el usuario puede preguntar al evaluador cualquier aspecto relacionado con el sistema y éste le responderá en la medida de su capacidad (Hom, 1998).

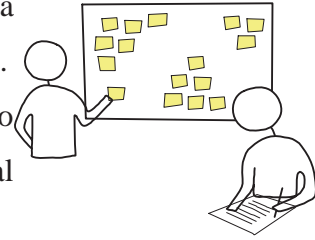
Está centrado en el usuario inexperto y su propósito es descubrir las necesidades de información de los usuarios de tal manera que se proporcione un mejor entrenamiento y documentación evitando la necesidad de preguntas en un nuevo rediseño de la interfaz (Abascal et. al, 2006).

f) Ordenación de tarjetas (card sorting)

Esta técnica consiste en solicitar a un grupo de usuarios representativos que agrupen los conceptos representados en cada tarjeta por su similitud semántica. Su propósito es identificar qué conceptos, de los representados en cada tarjeta, tienen relación semántica entre sí, e incluso cuál es el grado de esa relación (Hassan y Ortega, 2009).

Estas pruebas pueden ser realizadas de forma manual o virtual. En el caso manual, los

conceptos son representados en tarjetas reales (papel o cartón), y los participantes proceden a agruparlas sobre una mesa. En el caso virtual se emplean aplicaciones software específicas, mediante las que los participantes realizan la prueba facilitando la recogida de datos y su posterior análisis estadístico. Esta prueba debería realizarse en etapas tempranas del proyecto pues está destinada a adaptar la arquitectura de información al modelo mental del usuario.



Aunque se trata de una técnica simple (fácil de entender y de aplicar), barata, rápida (Spencer & Warfel, 2007), coincidimos con Cañas, Antolí & Quesada (2001) en que su uso inexperto o inadecuado puede producir resultados erróneos.

g) Eye-tracking

El Eye tracking (ET) –seguimiento de los ojos– es una técnica que permite documentar los puntos del sistema de software o sitio web que ha estado mirando el usuario en cada momento, la dificultad para encontrar aspectos concretos de la interfaz y el tiempo o el número de elementos analizados antes de determinar el adecuado. Esto permite que los evaluadores puedan saber, por ejemplo, si una de las causas de que los usuarios no perciban correctamente la Arquitectura de la Información es que no hayan fijado la mirada en algunos de los elementos relevantes de la misma (López-Gil et. al., 2010). Lo más reciente son los trabajos utilizando ET realizados por Nielsen & Pernice (2010).

Es una tecnología que permite seguir los movimientos oculares de una persona para inferir qué mira y qué ve. Esto se consigue actualmente mediante un eyetracker, un monitor especial que lanza rayos infrarrojos a los ojos de quien lo usa. Estos rayos rebotan en su pupila y vuelven al aparato, permitiendo así calcular con precisión dónde está mirando (Nielsen y Pernice, 2010).

Existen tres formas de visualizar los resultados: análisis de los videos del seguimiento de la mirada en cámara lenta (consume mucho tiempo), los mapas de calor y los diagramas de visión.

Los mapas de calor son la técnica de visualización más conocida para el estudio de ET. En ellos vemos una captura de pantalla con un código de colores que indica la cantidad de miradas que atrajo cada una de las partes. En un diagrama de visión, se representa la visita a una página de un único usuario mediante una serie de puntos azules cuyo tamaño representa la duración de dicha fijación (Figura 5.3).



Figura 5.3. Mapa de color y diagrama de visión

4. Una solución ecléctica: Combinación de métodos de evaluación

Nielsen y Pernice (2010) señalan que miden muchos de los resultados de sus trabajos de usabilidad en una forma cualitativa y además utilizan cuatro medidas cuantitativas básicas en los trabajos sobre usabilidad: tiempo de realización de la tarea, calificación del éxito errores cometidos y satisfacción subjetiva del usuario.

En palabras de Kort et. al. (2005):

La dirección más fructífera para el desarrollo de soluciones para la investigación y la evaluación en DCU de un sistema complejo ha de ser una combinación de métodos cuantitativos y cualitativos, que se puede aplicar de manera más discreta y en una escala mayor, in situ.(p.1).

- El método “Medida de las prestaciones” se puede utilizar en combinación con las “Pensando en voz alta retrospectivo “ o después de la entrevista o cuestionario , cruzando de esta manera los datos cuantitativos y cualitativos que se obtienen (Hom, 1998).
- Combinación de “Evaluación heurística” con el “Recorrido cognitivo con usuarios” u otro test con usuarios. Ejemplo de casos de estudio son los trabajos de Chisnell, Redish & Lee (2006), Molich, et. al (2004), Granollers (2004), Kort et. al.(2005), Conte et. al (2009), Reyes y Liberos (2011) entre otros.
- Autores como Redish (2007) recomiendan combinar “Evaluación heurística” con

“Pensando en voz alta retrospectivo”.

- Como expresamos anteriormente, el “análisis de logs” no responde a todas las preguntas de usabilidad y en algunos casos, es probable que incluso plantee otras nuevas.

Coincidimos con Gerken et. al.(2008) en que al analizar los resultados de estos registros, nos queda preguntarnos por qué un usuario se comportó de una cierta inesperada manera. Es decir, la interpretación de los datos puede ser realmente problemático y propenso a errores.

Consultores de usabilidad, tales como Nielsen Norman Group (<http://www.nngroup.com>), SirValuse (<http://www.sirvaluse.com>), Xexperience (<http://www.xperienceconsulting.com>) entre otros, ofrecen la integración de mecanismos de retroalimentación (experiencia de usuario) dentro de sus servicios de análisis de sitios web, de modo que los usuarios directamente pueden hacer comentarios sobre los aspectos de un sitio web o una aplicación web.

Gerken et. al. (2008) recomienda vincular el “análisis de logs” con métodos cualitativos tales como entrevistas u otro test de usuario, que ofrecen a los diseñadores la información necesaria para apoyar las preferencias de sus usuarios y aumentar su satisfacción al utilizar el sistema.

De acuerdo con Gerken et. al. (2008) , la combinación de estos dos métodos de evaluación - el análisis de logs y los métodos cualitativos deben proporcionar una evaluación más clara y más precisa acerca de la usabilidad de un sistema, al permitir la validación cruzada de los resultados obtenidos y la selección de una adecuada información del usuario.

- Recientemente se han presentado trabajos que emplean una combinación de métodos de inspección, indagación y test. Ejemplos de ello son los de Nielsen y Pernice (2010): “Evaluación Heurística”, eye-tracking y “Pensando en voz alta retrospectivo”; López-Gil et. al.(2010) : test de estrés de navegación, “Pensando en voz alta retrospectivo”, cuestionario de usabilidad y eye- tracking ; Aceves , Argüello y del Río (2009) : Evaluación heurística , “Recorrido cognitivo con usuarios”, Inspecciones de estándares, eye- tracking, “Pensando en voz alta retrospectivo” y encuestas ; Olsen, Smolentzov & Strandvall (2009) : eye- tracking, cuestionarios y “Pensando en voz alta retrospectivo” ; Bojko & Schumacher (2008) : “Recorrido cognitivo con usuarios”, eye-tracking, “Pensando en voz alta retrospectivo” y cuestionarios.

CAPITULO 6. METODOLOGÍA DE DESARROLLO Y EVALUACIÓN DE UN OBJETO DE APRENDIZAJE (OA)

En este capítulo se expone el aporte de esta tesis: una propuesta de Metodología de desarrollo y evaluación de OA. En la sección 6.2, se desarrolla el Modelo de Proceso para el desarrollo de Objetos de Aprendizaje (MPOBA) con la descripción de cada una de sus fases.

6.1. Introducción

El OA, como recurso digital, establece metas pedagógicas y en este sentido consideramos su evaluación desde ambos aspectos.

Si ampliamos el concepto de usabilidad al campo de la valoración de los OAs podríamos utilizar métodos de evaluación de usabilidad de productos de software considerando criterios que involucren dimensiones pedagógicas. Sicilia & García (2003), Mor, Garreta y Galofré (2007), Morales, Gómez y García Peñalvo (2008), entre otros, recomiendan aplicar estos métodos.

En cuanto al momento de la evaluación de la usabilidad, autores como Granollers (2004) y Hassan, Fernández y Iazza (2004), sostienen que la usabilidad debería ser considerada en todo momento, desde el mismo comienzo del proceso de desarrollo hasta las últimas acciones antes de librar el sistema a sus destinatarios.

En este sentido, apuntamos a realizar una evaluación formativa del OA. Se denomina evaluación formativa (Redish et. al., 2002) a una colección de métodos de Ingeniería de la Usabilidad (IU) que centra su actividad en un proceso iterativo de identificación de problemas de usabilidad y su correspondiente corrección (“identificar y corregir”) antes de que un producto esté completado.

Nielsen (1994c) declara que en la realización de pruebas con usuarios con el objetivo de aprender sobre el diseño para mejorar la nueva iteración y no tiene que ser confundida con la evaluación aditiva o sumativa, que es un método que juzga el valor de un programa al final de las actividades del mismo (Bhola, 1990), el enfoque está en el resultado mediante el cual ofrece una comparación cuantitativa entre un producto (a menudo completo) y un producto competitivo o uno estándar.

En relación a los métodos de evaluación, autores como Hom (1998), Kort et. al. (2005), Redish. (2007), Gerken et. al. (2008) y Nielsen y Pernice (2010), destacan que los métodos cuantitativos pueden enriquecerse con las expresiones de los usuarios durante la interacción con el sistema a través de registrar y resumir cualitativamente conceptos como satisfacción o los problemas generales de usabilidad. Es decir, desde una visión ecléctica de la evaluación, concluimos en realizar la evaluación del OA con varios métodos de evaluación de usabilidad.

6.2. MPOBA: Modelo de Proceso para el desarrollo de Objetos de Aprendizaje

En todo proceso de desarrollo de software existe una fase más o menos importante en la cual, a base de una serie de repeticiones, se pasa de una aproximación de la solución ideal a la solución definitiva.

Este proceso de repetición en la Ingeniería clásica del software se produce en una fase más tardía que en la Ingeniería de la Usabilidad (IU), y suele ser más costosa en cuanto a recursos y tiempo empleado.

En base a lo definido en los capítulos anteriores y de acuerdo a las actividades que define la ISO 9241-210 (2010) se describe el aporte de esta tesis: “**El Modelo de Proceso para el desarrollo de Objetos de Aprendizaje**”.

En la Figura 6.1, puede observarse esquemáticamente el Modelo **MPOBA**.

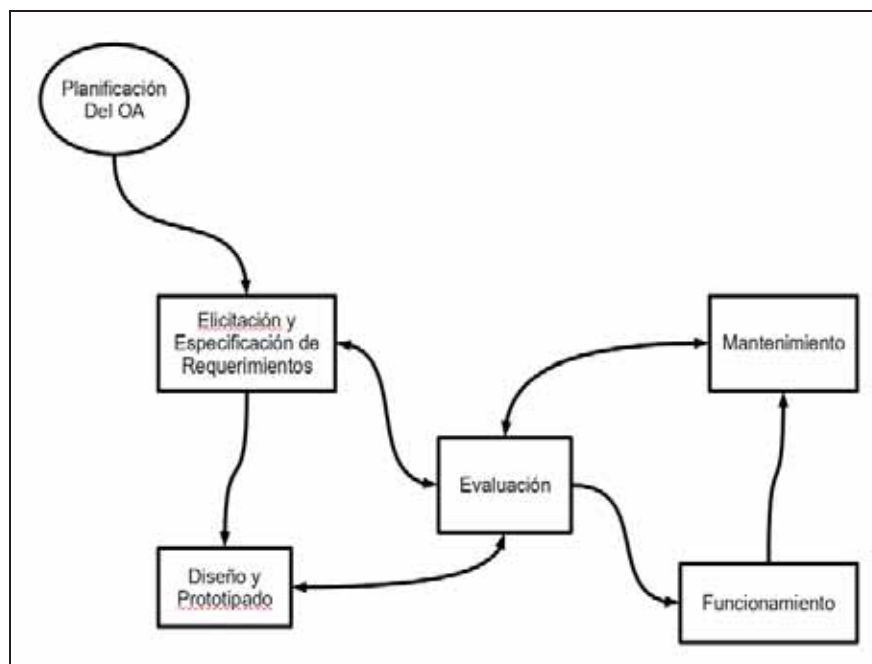


Figura 6.1. Modelo de proceso MPOBA

El esquema de la Figura anterior nos muestra las diferentes fases en las que se divide el modelo de proceso y cómo se relacionan cada una de ellas.

Podemos observar también que el modelo no sigue un sentido lineal ni restrictivo. Estas fases no deben verse nunca como si fueran fases estancas y sucesivas sino fases iterativas. El equipo de desarrollo del OA, junto con los requerimientos a implementar, son los que marcarán cuantas iteraciones son necesarias.

6.2.1. Fases del modelo MPOBA

El modelo está organizado en una serie de fases que se ejecutaran repetidamente durante el desarrollo de un OA determinado.

A continuación se presentarán conceptualmente cada una de las fases, sus objetivos y las actividades de cada una de ellas.

La Fase “Prototipado” se explicará primero pues es un aspecto primordial en todo diseño de sistemas centrado en el usuario e inevitablemente se hace referencia continuamente al explicar el resto de fases y sus actividades relacionadas.

Con respecto a la Fase “Evaluación”, en la subsección 5.4.2. del capítulo 5, se describió con profundidad cada una de las técnicas por lo que en las fases restantes se explicará su aplicación específica.

6.2.1.1. Prototipado

El prototipado es una metodología que engloba técnicas que permitirán la posterior fase de evaluación.

Los prototipos, son documentos, diseños o sistemas que simulan o tienen implementadas partes del sistema final.

El prototipado modela el producto final y permite efectuar una prueba sobre determinados atributos del mismo sin necesidad de que este disponible (Floría Cortés, 2000).

Desde la perspectiva del DCU, los prototipos constituyen mucho más que simples demostraciones del producto. Se utilizan para demostrar conceptos, informarse sobre los problemas y sus posibles soluciones y además recoger las impresiones del usuario para reflejarlas en el diseño de la interfaz Granollerss (2004), Preece et. al. (1994), Sommerville (2005) y Pressman (2002).

En cuanto al momento de su aplicación, Sommerville (2005), Pressman(2002) . Maner (1997), recomiendan:

- a) en el proceso de Ingeniería de Requerimientos, para obtener, refinar y validar los requerimientos.
- b) en el proceso de diseño, para explorar soluciones de software particulares y apoyar el diseño de interfaz.
- c) en el proceso de pruebas, para ejecutar pruebas back to back (de contraste) con el sistema que se entregará al cliente.
- d) son esenciales para la documentación, tanto de conceptos funcionales del sistema como de tareas concretas del mismo.

Cuando se utiliza en etapas tempranas del proceso de desarrollo, un prototipo alienta la participación e implicación del usuario, y permite a los desarrolladores observar el comportamiento de los usuarios y su reacción ante el prototipo (Hix & Hartson, 1993).

A continuación se detallan algunas de las técnicas para la construcción de prototipos. Algunas dan más importancia a la presentación (look) y otras a la interacción (feel).

a) Prototipos de papel

Es una técnica de baja fidelidad que se basa en la utilización de materiales sencillos como lápiz, el papel y las tijeras para la creación de prototipos simples (Figura 6.2), poco costosos pero enormemente versátiles (Sommerville, 2005).

Su utilización no precisa incorporar software; sólo es necesario que capture la funcionalidad del sistema y que comunique la información y sus interacciones adecuadamente (Granollers, 2004)



Figura 6.2. Prototipo en papel de Jack Dorsey del status de twitter. Tomado de <http://www.flickr.com/photos/jackdorsey/182613360/>

b) Storyboards

Tienen su origen en la industria cinematográfica. Consisten en una serie de dibujos o imágenes dispuestos en formato secuencial de viñetas que ilustran los distintos pasos de un sistema durante consecución de una determinada tarea (Figura 6.3).

Permiten crear diferentes vistas del sistema en las primeras etapas de su implementación de la manera más rápida y barata posible. Es un material de soporte para evaluar si el diseñador ha comprendido el sistema durante las reuniones con usuarios, implicados y responsables del proyecto. No es adecuado para comprobar aspectos referentes a la interactividad del sistema.

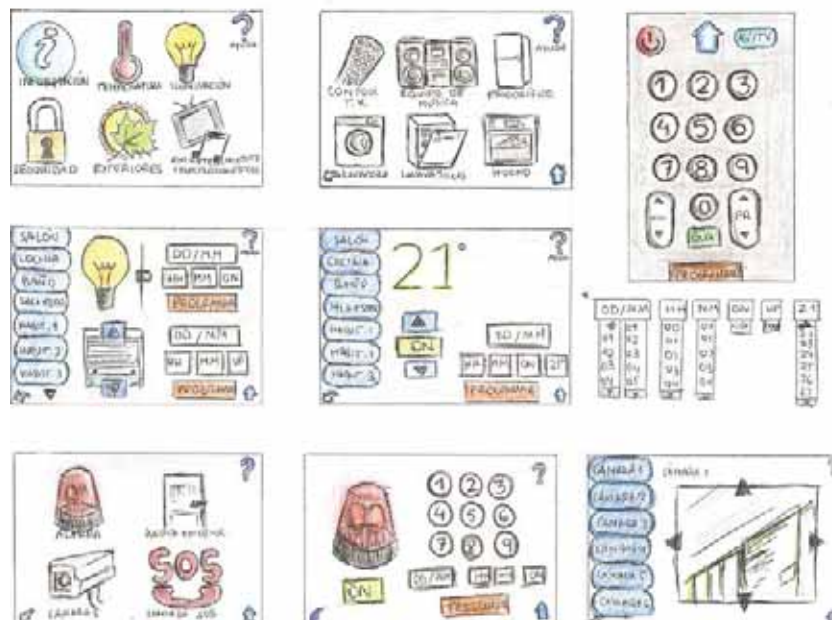


Figura 6.3. Storyboard de un sistema remoto de control de funcionalidades de una vivienda. Tomado de <http://proyectoipo.blogspot.com.ar/>

c) Maquetas digitales (wireframes)

En diseño web, un wireframe es una representación esquemática de una página web sin elementos gráficos que muestran contenido y comportamiento de las páginas (Figura 6.4). Sirven como herramienta de comunicación y discusión entre arquitectos de información, programadores, diseñadores, usuarios e implicados. También se pueden utilizar para comprobar la usabilidad de un sitio web.

Son representaciones de calidad en formato digital que por su mayor nivel de detalle permite visualizar de una manera muy aproximada a la versión final el diseño de la interfaz

(colores, estructura de navegación, botones, etc.).

Son percibidas por los usuarios como versiones finales que no se pueden cambiar, por lo que es más adecuado utilizarlas en la fase de diseño.

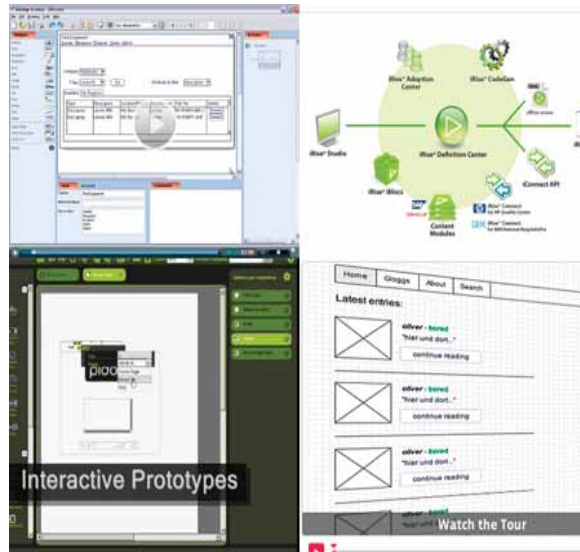


Figura 6.4. Maqueta digital .Tomado de <http://www.maspixel.com>

d) Storyboard navegacional

Es una técnica diferente del storyboard que consiste en desarrollar una serie de dibujos o imágenes que representan el espacio de navegación, bien sea de todo el sistema, de una parte de él o de una tarea concreta (Figura 6.5). Se representan en un espacio bidireccional (con papel, en una pizarra, con impresiones de pantalla y flechas con rotulador, etc.) todos los estados de las interfaces (pantallas) de la parte del sistema que se examinará y todas las posibilidades a nivel interactivo desde cada uno de estos estados para visualizar las posibles acciones o movimientos que el usuario puede realizar mientras interacciona con la interfaz (Granollers, 2004).

Componente	Descripción
Título	Identificación del escenario.
Objetivo	Meta a ser alcanzada en el dominio de la aplicación. El escenario describe la forma de lograr el objetivo.
Contexto	Describe las acciones previas necesarias para iniciar el escenario, las precondiciones, la ubicación física y temporal.
Recursos	Identifican los objetos con los cuales los actores trabajan.
Actores	Detalla las entidades que se involucran activamente en el escenario.
Set de episodios	Cada episodio representa una acción realizada por un actor, donde participan otros actores y se utilizan recursos. Los episodios se ejecutan secuencialmente. Un episodio también puede referenciar a un escenario. Se incluyen restricciones del escenario o episodio según corresponda.
Excepciones	Menciona los casos de excepción, que pueden corresponder a otros escenarios.
Dudas:	Puntos pendientes a clarificar con el usuario.

Tabla 6.1. Esquema de escenario de Leite

Los escenarios se utilizan en diferentes momentos del proceso de desarrollo del software. Estimulan, por una parte, la imaginación creativa de los diseñadores, mientras que por otra proporcionan herramientas ágiles que dan soporte al razonamiento del sistema en el proceso de diseño (Carroll ,2000). Ayudan a escenificar problemas existentes, favoreciendo la comprensión de los mismos y sus posibles vías de solución.

Es importante que el escenario contenga la mayoría de los aspectos que directa o indirectamente intervienen durante el proceso interactivo, destacando aquellos que son claves para que su consecución futura sea posible.

e) Prototipos de software

Son implementaciones realizadas con técnicas de programación del sistema interactivo propuesto que reproducen las funcionalidades del sistema o aplicación con el objetivo de probar determinados aspectos del sistema final (Figura 6.6). Es una visión preliminar del

modelo futuro, es un modelo operable, fácilmente ampliable y modificable, que tiene todas las características propuestas, pero realmente es un modelo básico que tiene que ser mejorado.

El prototipo de software se implementa después de varias iteraciones de Prototipado-Evaluación y su intención es empezar a ver realmente cómo responde el sistema.



Figura 6.6. Prototipo de software educativo. Fuente: Rodríguez Castro,Pino Roque, García Pino (2011).

6.2.1.2. Planificación del OA

Dado que los OA se incorpora a un Entorno Virtual de Enseñanza Aprendizaje (EVEA) en el que se pueden encontrar otros OAs, previo a la realización de las fases, se debe contar con el siguiente análisis preliminar:

- Caracterización de los usuarios finales (profesores y estudiantes).
- Plataforma (posibilidades/restricciones): selección de la Plataforma educativa (EVEA) en dónde se incorporará el OA.
- Determinación de los perfiles de usuario y roles.

6.2.1.3. Elicitación y especificación de requerimientos

La más notable de las definiciones pertenece a la IEEE 610 -12 (1990):

- Una condición o capacidad necesaria para un usuario para resolver un problema o alcanzar un objetivo.
- Una condición o capacidad que debe ser alcanzada o poseída por un sistema o componente de un sistema para satisfacer un contrato, estándar, especificación, u otro documento formalmente impuesto.

- Una representación documentada de una condición o capacidad dada en los puntos anteriores.

La Ingeniería de Requerimientos se define como el proceso sistemático de desarrollo de los requerimientos a través de un proceso cooperativo e iterativo de análisis del problema, documentando las observaciones resultantes en una variedad de representaciones y chequeando la certeza del conocimiento (Pohl, 1993).

Los requerimientos de los sistemas interactivos suelen enfocarse en el qué debe hacer el sistema y no en cómo debe hacerlo, por lo que suelen clasificarse básicamente en requerimientos funcionales (lo que el sistema debe hacer, que necesita ser mantenido y procesado) y no funcionales (restricciones globales de cómo debe construirse y funcionar el sistema), aunque también obedecen a criterios de prioridad (esenciales, deseados y deseados pero de baja prioridad —“estaría bien tenerlos”—) (Granollers (2004), Loucopoulos & Karakostas, (1995)).

Los modelos de Ingeniería de Requerimientos incorporan otros factores a tener en cuenta que, de realizarse, garantizarán el desarrollo de un sistema con un grado mucho mayor tanto desde el punto de vista funcional como de su usabilidad y de su accesibilidad.

Aun así, las aproximaciones al desarrollo de software “centradas en el usuario” reconocen que es imposible especificar todos los requisitos por adelantado (Brooks, 1995).

Los clientes no pueden apreciar sus necesidades reales hasta que no pueden ver e interactuar con las opciones de que disponen. En palabras de Brooks (1995) :“(..)muchos requerimientos son descubiertos una vez los usuarios interactúan con los prototipos; o incluso más, no aprecian las nuevas tecnologías hasta que no las prueban, simplemente porque las desconocen”.

No se sabe lo que se ha desarrollado hasta que no se ha realizado el primer versión (“wrong system”)(Rosson y Carroll, 2002)

Es imposible pues, determinar todos estos objetivos en una primera fase o visita con el cliente. Haciendo uso de una serie de técnicas y validaciones referenciadas por el modelo de proceso se tratará de definir toda la información lo más detallada y efectivamente posible para lograr un producto de calidad.

En este sentido, coincidimos con Presman (2002) en cuanto a que los requerimientos están indisolublemente vinculados con la calidad del sistema.

a) Elicitación de conocimiento para la generación de requerimientos

La Elicitación de requerimientos es el proceso de adquirir (“eliciting”) todo el conocimiento relevante, necesario para producir un modelo de requerimientos de un dominio de problema. Es un proceso que comprende una serie de actividades para establecer lo que requiere el cliente, y con esos requerimientos identificados y validados se lleva a cabo el desarrollo del sistema.

Para realizar esta tarea se aplican técnicas para facilitar las especificaciones de la aplicación, este enfoque es partidario de la creación de un grupo compuesto por clientes y desarrolladores que trabajaran en conjunto para identificar el problema, brindar diferentes enfoques y especificar un conjunto preliminar de requisitos (Pressman, 2002).

En esta tesis, seguimos el modelo de de Proceso de la Ingeniería de Requisitos de Pohl (1997). Es un modelo iterativo en el que se definen las cuatro actividades básicas: se asume una secuencia en la que los requisitos son *elicitados*, a continuación son *negociados* entre los participantes, se integran con el resto de la documentación y finalmente se *validan* y *verifican* para asegurar que no presentan conflictos con los demás requisitos y que se corresponden con las necesidades reales de los clientes y usuarios (Durán, 2000).

A continuación se detallan cada una de técnicas utilizadas en la etapa Elicitación de Requerimientos del Modelo de proceso MPOBA:

a-1) Entrevistas

La identificación de los objetivos del OA se realizará mediante entrevistas a profesores. Los objetivos deben ser formulados de una manera clara, concreta y sin lugar a ambigüedades y desde el punto de vista pedagógico y tecnológico. Serán estos, en armonía con las necesidades de los usuarios, los que guiarán todo el proceso de conceptualización, diseño e implementación.

a-2) Indagación Contextual

Consiste en un estudio de las tareas actuales de los usuarios (estudiantes y profesores) cómo las realizan, qué estrategias de aprendizaje y enseñanza utilizan dentro del contexto en el que se desarrollan dichas tareas (dentro y fuera del aula) y, con ello, llegar a especificar y entender el dominio del problema, los objetivos de los usuarios, el lenguaje propio y los esquemas mentales. Se realiza la investigación por medio de la observación de

campo.

a-3) Cuestionario o Encuesta

Destinado a estudiantes con el objeto de conocer sus hábitos de interacción y acceso a Internet y su percepción en cuanto al propio aprendizaje.

a-4) Focus Group con implicados

El objetivo es rescatar e interpretar los hechos relevantes que serán necesarios a tener en cuenta para que el OA cumpla los requerimientos funcionales y no funcionales. La idea básica es ir obteniendo los requerimientos como un refinamiento de los objetivos, de forma que la existencia de un requerimiento esté siempre justificada como una necesidad para alcanzar uno o más objetivos (Duran, 2000).

b) Especificación

La norma IEEE 610-12 (1990) define: “La Especificación de requerimientos es un documento que especifica los requisitos para un sistema o componente. Se incluyen típicamente requerimientos funcionales, de rendimiento, de interfaz, de diseño y normas de desarrollo”.

Una especificación (Loucopoulos & Karakostas, 1995):

- Describe objetos, procesos, reglas de negocios, agentes, componentes del sistema de información.
- Representa un conjunto de proposiciones que se consideran válidas hasta que se demuestre lo contrario.
- La especificación modeliza lo que necesita y formula el problema u objetivo.

En relación a la especificación de los requerimientos funcionales:

- deben estar redactados de tal forma que sean comprensibles para usuarios sin conocimientos técnicos avanzados ,
- deben especificar el comportamiento externo del sistema y evitar, en la medida de lo posible, establecer características de su diseño,
- deben priorizarse (al menos, se ha de distinguir entre requisitos obligatorios y requisitos deseables).

En cuanto a la especificación de los requerimientos no funcionales:

- han de especificarse cuantitativamente, siempre que sea posible (para que se

pueda verificar su cumplimiento).

A continuación se detallan cada una de técnicas utilizadas en la etapa de Especificación de Requerimientos del modelo de proceso **MPOBA**:

b.1) Definición de los objetivos del OA: desde el punto de vista pedagógico y tecnológico
El objetivo de esta tarea es conocer por qué se realizará el desarrollo, y por lo tanto conocer qué objetivos se esperan alcanzar mediante el sistema software a desarrollar. En la primera iteración se realizará una primera identificación de los objetivos. En las iteraciones posteriores puede que sea necesario revisarlos si se han identificado conflictos que les afecten.

Esta información puede que haya manifestado de forma explícita durante las sesiones de elicitación o puede que haya que deducirla de la información recopilada durante dichas sesiones (Duran, 2000).

b.2) Verificación de la especificación generada con respecto a sus requerimientos

La validación de los requerimientos se realizará a partir de descripciones formales de Escenarios en lenguaje natural que describen los requerimientos del sistema en el contexto de las especificaciones funcionales y no funcionales, mostrando cómo se efectúan los procesos de enseñanza y de aprendizaje y qué actores o perfiles de usuario intervienen en éstos a través de las secuencias de tareas descritas para cada uno de los escenarios.

El estándar ISO/IEC 9126-1 (2001) señala que al aplicar un modelo de calidad resulta prácticamente imposible definir (para después evaluar) todos los posibles Escenarios de un sistema, y tendremos, por tanto, que definir aquel grupo de escenarios que mejor represente la globalidad del sistema.

El estándar explica que la evaluación de los Escenarios es la mejor manera de evaluar la calidad de uso del software aunque no especifica ninguna técnica para realizarla.

En el caso de los OA, el término "escenario" se utiliza para indicar una secuencia discreta de pasos dentro de un sistema de gestión del aprendizaje o de otro tipo de aprendizaje relacionada con el sistema (Sicilia & Lytras, 2005). En particular, Luna- Pérez, et. al. (2009) definen "Escenario de Aprendizaje" a la secuencia de pasos en una actividad de aprendizaje con un fin pedagógico específico, el cual comprende un conjunto de reglas a seguir para que los estudiantes puedan adquirir, reforzar o asimilar conocimientos dentro de un marco de estudio.

En esa línea, se desarrollan varios Escenarios de aprendizaje que describan los requerimientos del OA.

Posteriormente se realizan nuevas reuniones Focus Group con los implicados, con el fin de ampliar información sobre aquellos escenarios cuya descripción resultara incompleta y/o confusa a partir de la información obtenida.

b.3) Selección de Escenarios

A partir de la validación de requerimientos, se propone la descripción de Escenarios evaluados como etapa previa al diseño de la navegación e interfaz del usuario.

6.2.1.4. Diseño

Según Pressman (2002), el diseño de software es tanto un proceso como un modelo y este proceso es una secuencia de pasos que hacen posible que el diseñador describa todos los aspectos del software que se va a construir. El proceso de diseño traduce requisitos en una representación de software.

Generalmente la fase de diseño produce un diseño de datos, la arquitectura de la información, un diseño arquitectónico, el diseño de interfaz, y el diseño procedimental (Pressman, 2002).

En el caso de los OA el diseño de datos no juega un papel determinante ya que no requiere de estructuras de datos complejas, ni de un esquema de base de datos por ejemplo.

Hassan y Ortega (2009) definen la arquitectura de información como la actividad y resultado de organizar, clasificar, ordenar, estructurar y describir los contenidos de un sitio Web, con el fin de que sus usuarios puedan satisfacer sus necesidades informativas con el menor esfuerzo posible. Representa su esqueleto, el elemento del diseño que sostiene estructuralmente el contenido.

En el caso de la Web, el diseño de la interacción está íntimamente relacionado con la arquitectura de la información. El diseño de interacción define la estructura y el comportamiento interactivo del sitio, las acciones que se ofrecerán a los usuarios y como responderá la aplicación a las acciones que realice.

Según Bou Bouza (2003), la interactividad supone un esfuerzo de diseño para planificar una navegación entre pantallas en la que el usuario sienta que realmente controla y maneja una aplicación. Esta idea cobra un sentido especial en el caso de los OA pues se orienta al aprendizaje autónomo del estudiante.

Una vez que se ha definido la arquitectura de la información y el diseño de interacción de un sitio Web, se procede a diseñar en detalle su aspecto gráfico o visual.

La interfaz Web posee ciertas particularidades que la hacen diferente de otras aplicaciones multimedia y aunque comparte los mismos principios y varias características respecto de su diseño, es importante conocer sus singularidades.

Powell (2001) señala que un sitio Web se conforma, desde el ámbito del diseño, a partir del contenido, la forma, la función y la finalidad; que se establecen según los requerimientos particulares cada proyecto.

(...) ¿Cómo podríamos definir exactamente el concepto de diseño Web? Es un diseño multidisciplinario muy centrado en el usuario que incluye influencias de las artes visuales, la tecnología, el contenido y la finalidad. (...) Aunque es verdad que la red toma prestado muchas cosas de los otros medios, existen diferencias muy significativas. (p.21).

Para Royo (2004), los códigos lingüísticos gráficos son los sistemas de signos que nos permiten visualizar e interactuar con la información y en la Web podemos identificar dos tipos: los visuales y los secuenciales. En los primeros, se encuentran: la escritura alfabética (tipografía), la escritura no alfabética (iconos, esquemas) y la imagen fija (fotografías). Los códigos secuenciales, son: la imagen en movimiento (video y animación) y la hipertextualidad (estructuras de interacción).

El trabajo principal de un diseño responsable es que las aplicaciones sean “usables”, siendo el usuario el centro de las preocupaciones del diseño (Norman y Draper, 1986). Pero una interfaz usable sin un diseño visual coherente con una identidad y una estética agradable al usuario, jamás será un buen diseño de interfaz gráfica. Se desprende de las definiciones de usabilidad que fueron citadas en el sección 5.2 del capítulo 5, que la misma depende de tres factores:

- El usuario que reconocerá, leerá y manejará la aplicación en función de sus conocimientos y de la capacidad de uso del artefacto.
- La aplicación, que será leída y manejada por el usuario según el diseño de la interfaz.
- El contexto, que influirá y en algunos casos determinará el uso de la aplicación, modificando su significado.

a) Diseño del OA, desde el punto de vista pedagógico

Se deben seleccionar:

i) los recursos didácticos (textos impresos, recursos multimediales, software educativo, etc.).

ii) las estrategias de enseñanza que promuevan un aprendizaje significativo.

Tomamos la clasificación de las estrategias de Diaz Barriga y Hernandez Rojas (2002), de acuerdo al proceso cognitivo asociado:

- Activar (o generar) conocimientos previos y establecer expectativas adecuadas en los estudiantes.
- Orientar la atención de los estudiantes.
- Organizar la información que se ha de aprender.
- Promover el enlace entre los conocimientos previos y al nueva información que se ha de aprender.

iii) Las actividades de aprendizaje que, de acuerdo con Fernandez-Salineró (1999), deben reunir las siguientes características

- Se vinculan siempre a los contenidos.
- Hacen referencia a un sujeto o agente.
- Deben ser propositivas (orientadas hacia un objetivo).
- Se formulan en tiempo y espacio.
- Deben contextualizarse.
- Se precisan en razón de los recursos y de las estrategias metodológicas.
- Incorporan mecanismos de retroinformación y autorregulación.

Desde una visión situada, se aboga por una enseñanza centrada en prácticas educativas auténticas, las cuales requieren ser coherentes, significativas y propositivas que puedan incluir:

- Aprendizaje centrado en la solución de problemas auténticos.
- Análisis de casos (case method).
- Método de proyectos.
- Prácticas situadas o aprendizaje in situ en escenarios reales.
- Trabajo en equipos cooperativos.
- Ejercicios, demostraciones y simulaciones situadas.

iv) La evaluación de los aprendizajes

Existen diferentes maneras de considerar la evaluación: como medición o como comprensión. En el primer modelo tiene funciones de control o acreditación y en el segundo, de discusión y retroalimentación.

Es decir, la evaluación de los aprendizajes cumple varias funciones, entre ellas certificar saberes, pero también ofrece información sobre el proceso de aprendizaje que lleva a cabo el estudiante.

Se deben plantear actividades de evaluación que a cada respuesta del estudiante, se le asocie un texto de refuerzo, pista o explicación orientativa atendiendo educativamente el error a través de una retroalimentación intencionada y centrada en los obstáculos de aprendizaje.

De esta manera, el sistema informa el desempeño del estudiante en cada una de las actividades de evaluación propuesta (feedback automático en los casos de respuestas cerrada y respuesta individual del profesor para el tratamiento del error).

Así, el error es integrado en el aprendizaje, cobra sentido y refuerza la respuesta dada por el estudiante. Se fijan logros o expectativas inmediatas, ya que el estudiante recibe muy pronto la retroalimentación que verifica su logro.

Los recursos multimediales cobran “vida” con el aporte que realice el profesor desde la reflexión propiciada en el contexto de dicho recurso hasta los diálogos que se desarrollen con el estudiante a partir de los bloqueos en el aprendizaje producidos por conflictos de significados.

Es decir, para que el estudiante logre un nivel de generalidad sobre el conocimiento deseado es necesario crear puentes cognitivos en las actividades planteadas mediante recursos multimediales, tanto en el planteo de la actividad como en la retroalimentación posterior ofrecida por el profesor.

Finalmente, en el diseño de materiales didácticos, coincidimos con las recomendaciones que realiza Cabero (2004):

- cuanto menos, más,
- lo técnico debe estar supeditado a lo didáctico,
- la legibilidad contra la irritabilidad,
- evitar el aburrimiento,
- facilitar la interactividad,
- promover la flexibilidad,

- lograr la hipertextualidad,
- generar siempre la participación del estudiante.

Desde el diseño multimedia podemos generar contenidos interactivos que hagan que el estudiante se sienta partícipe en el aprendizaje (diseño constructivo) a través de actividades, mapas conceptuales, gráficos escenificados y simuladores. Es importante aplicar la interactividad cuando ésta tiene un seguimiento por parte de los tutores y el estudiante se siente integrado en el aprendizaje.

Según Herrera (2004), el diseño actúa como apoyo visual que utiliza canales de refuerzo, resortes diferentes al lenguaje escrito, que potencian la intención del mensaje.

En esa línea, Santiso y González (2005), Royo (2004), destacan la función de los siguientes elementos en el diseño de contenidos digitales educativos:

- *Color*: los colores son códigos que ayudan a la identificación de ideas clave, la clasificación visual directa de la información que se presenta reforzando los conceptos. Crear un sistema de colores facilita al estudiante un reconocimiento visual del contenido.
- *Tipografía*: los distintos estilos de tipografías ayudan a distinguir entre conceptos globales y específicos. Determina la correcta legibilidad de la información en la pantalla y tiene una intervención directa en la visión del usuario.
- *Iconos*: encierran significados concretos acerca de acciones que se pueden realizar. Su uso permite minimizar el espacio utilizado en pantalla, manteniendo al mismo tiempo mucha información.
- *Gráficos animados*: los conceptos son pregnantes cuando se los visualiza gráficamente. La utilización de mapas conceptuales, gráficos esquemáticos, teniendo en cuenta los colores y las tipografías permiten al estudiante mayor claridad en los contenidos.
- *Imágenes, ilustraciones*: las imágenes cuando acompañan al contenido ayuda a su comprensión. Son medios de expresión que orientan la actividad cognitiva del estudiante cumpliendo una doble función: representativa y expresiva.
- *Vídeo*: presenta varias aplicaciones como: la posibilidad de mostrar procedimientos complejos realizados por expertos, presentar metáforas visuales con capacidad de simbolismo o analogía respecto de procesos abstractos y la capacidad de simulación acelerada de procesos reales.

- *Simulaciones*: ofrecen un medio de experimentación en donde refuercen la capacidad de observación, análisis y toma de decisiones. Brinda la oportunidad de recrear fenómenos y modelos físicos o químicos o escenarios complejos de la naturaleza pudiendo aplicar los conocimientos teóricos, entender conceptos y procedimientos, resultados y consecuencias. Se constituyen como un medio de entrenamiento ideal (sin los costos reales o riesgos) que implica una verdadera experimentación en laboratorio.

Además, la representación de situaciones y de la información mediante gráficas, la visualización de transformaciones y construcciones geométricas, unida a la simulación, pone en manos de los estudiantes y profesores un mundo de variados matices el cual permite ampliar la comprensión de los conceptos y métodos más allá de la comprensión formal.

Por otra parte, los procesos cognitivos de aprendizaje son activados, dentro de los entornos virtuales, a través de los estímulos sensoriales que la interfaz gráfica provee al usuario (Herrera, 2004). Los estímulos actúan primero sobre la memoria sensorial e inmediatamente después sobre las memorias a corto plazo (MCP) y a largo plazo (MLP). Dado que estos procesos cognitivos son muy complejos, se recomienda administrar los estímulos para evitar una sobrecarga cognitiva al usuario y bloquear o desfavorecer así el aprendizaje efectivo.

b) Diseño de Navegación

Un elemento importante en la estructuración de la información, es el Diseño de Navegación, que consiste de acuerdo a lo planteado por Hassan (2002):

(..) definir la arquitectura de nuestro hipermedia: elementos de interacción entre el usuario y el sistema, enlaces y tipos de enlaces entre los nodos, agrupación de los nodos por categorías o propiedades, y respuestas del sistema ante peticiones del usuario.

Resenfeld & Morville (1998), autores de una de las obras más influyentes sobre representación y organización de la información en formato hipertextual indican que la estructura de la navegación estará basada en el diseño de diagramas, donde la información estará organizada por nodos (bloques, sitios, interfaces), los links, y las interacciones generadas en el sitio web (enlaces, tipos de menús de navegación, elementos de

interacción, botones, texto) con etiquetado de acuerdo al modelo mental de los usuarios.

Coincidimos con Rovira (2002) que en el contexto del hipertexto para la enseñanza y el aprendizaje, la estructura de la información se identifica completamente con la organización de los contenidos didácticos que componen el OA.

Por un lado, mediante la organización y secuenciación de los contenidos se pretende facilitar el aprendizaje, por el otro mediante el diseño de la navegación se intenta facilitar el acceso a la información. La integración de ambas perspectivas supone la creación de materiales didácticos en formato hipertextual eficientes.

La estructura de navegación se suele elegir en función del tipo de contenido. Existen distintos tipos de estructuras (Codina (2000), Rosenfeld & Morville (1998) y Lynch & Horton (1999)):

- Jerárquica: Estructura en árbol donde existe una página índice o principal desde donde se accede al resto de páginas.
- Lineal: Es una estructura muy simple similar a las páginas de un libro.
- Lineal con jerarquía: Es una estructura híbrida que trata de aprovechar las ventajas de las dos estructuras anteriores.
- Red: A partir de la página índice o principal se puede navegar a otra u otras sin ningún orden aparente.

c) Actividades

Debido a la naturaleza dinámica de las interfaces de usuario, autores como Sommerville (2005), Pressman (2002), Granollers (2004), consideran que el prototipo es la forma práctica de diseñar y desarrollar interfaces gráficas implicando al usuario en el proceso de diseño y desarrollo.

En esta fase iterativa las actividades consisten en la construcción de distintos prototipos que se van reformulando y desarrollando luego de las actividades de evaluación. Desde la perspectiva del DCU, los prototipos son evolutivos desde los de baja fidelidad a los de alta fidelidad. Es decir, el prototipo no es eliminado y se utiliza como base para una próxima iteración en el diseño.

Se realizan prototipos en papel y maquetas digitales. El diseño de la navegación se facilita a partir de la estructura aportada por técnicas como Card Sorting. Se construye un Storyboard Navegacional de alguna de las partes del sistema.

A continuación se detallan las técnicas de evaluación de prototipos del Modelo de proceso **MPOBA**:

c-1) Recorrido Cognitivo con implicados:

El objetivo es evaluar el diseño (en base a los prototipos) y la facilidad de aprendizaje (usabilidad pedagógica) del OA.

Procedimiento

i) Definir los usuarios del sistema

La descripción de los usuarios debe incluir la experiencia específica acumulada y el conocimiento adquirido como factores determinantes para la comprobación del factor “cognitivo” durante el recorrido.

ii) Definir las tareas que van a analizarse

El análisis se debe llevar a cabo sobre un número razonable de tareas, que deben seleccionarse de acuerdo con los requisitos propios del sistema y las necesidades de evaluación.

iii) Definir la secuencia correcta de acciones para cada tarea

Para cada tarea, debe haber una descripción de cómo se espera que el usuario la vea antes de aprender a utilizar el sistema, así como de la secuencia de acciones que permiten llevarla a cabo de forma correcta. Esta lista consta de una serie repetitiva de pares de acciones (del usuario) y respuestas (del sistema) que llevan a la consecución de la misma.

iv) Describir detalladamente el prototipo del sistema a utilizar para la evaluación

Es necesario especificar los elementos que deben estar activos en el prototipo, así como la interacción esperada.

v) Implementar para cada tarea la lista completa de acciones necesarias para ser completada con el prototipo descrito

Los evaluadores deben realizar cada una de las tareas determinadas anteriormente siguiendo los pasos especificados y utilizando el prototipo descrito. En este proceso el evaluador utilizará la información del factor cognitivo (experiencia y conocimiento adquirido) de los usuarios para comprobar si la interfaz es adecuada para los mismos. Esta

revisión ha de ser minuciosa para todas las acciones especificadas como necesarias para la consecución de la tarea.

En particular, el evaluador responderá las siguientes preguntas (Dix et.al. 1998):

- ¿Son adecuadas las acciones disponibles de acuerdo a la experiencia y al conocimiento del usuario?
- ¿Percibirán los usuarios que está disponible la acción correcta?. Esto se relaciona con la visibilidad y la comprensibilidad de las acciones en la interfaz. Aquí no se discutirá sobre si la acción se encuentra en el sitio adecuado o no, sino que se incidirá en si esta está presente y es visible.
- Una vez encontrada la acción en la interfaz, ¿asociarán estos usuarios la acción correcta al efecto que se alcanzará?
- Una vez realizada la acción, ¿entenderán los usuarios la realimentación del sistema? (tanto si la acción se ha realizado con éxito como en el caso contrario).

vi) Documentar los resultados:

Mientras se realiza la evaluación, es importante capturar la información por ejemplo en una grabación de video, hoja de reporte, esquemas, etc. (Wharton, 1992).

El evaluador anotará para cada acción las respuestas del sistema y hará los comentarios pertinentes detallando los aspectos negativos de la evaluación relacionándolos con un grado de severidad que permita distinguir aquellos errores más perjudiciales de los que no lo son tanto.

c-2) Pensamiento en voz alta con implicados

El objetivo es capturar aspectos relacionados con las actividades cognitivas de los usuarios potenciales (los estudiantes) del sistema evaluado mediante la evaluación del prototipo.

Procedimiento

- i) Se proporciona a los evaluadores el prototipo a probar y un conjunto de tareas a realizar.
- ii) Se les pide que realicen las tareas y que expliquen en voz alta qué es lo que piensan al respecto mientras están trabajando con la interfaz, describiendo qué es lo que creen que está pasando, por qué toman una u otra acción o qué es lo que están intentando realizar.
- iii) Se registran las reacciones de los evaluadores por medio de grabaciones de video o sonido o notas.

6.2.1.5. Puesta en funcionamiento del OA

Esta fase agrupa toda la programación del software necesario para concretar la creación del OA.

Para construir OAs se requieren los mismos conocimientos que se necesitan para construir cualquier otro material digital (imágenes, vídeos, páginas web, películas flash, etc.) con la diferencia de que además, según Hernández (2005) se necesita:

- Conocer la diferencia entre un asset y un Sharable Content Object (SCO), en el caso de SCORM y el concepto de Unidad Asignable (UA) para el caso de AICC.
- Comprender el concepto de API (Application Programming Interface) y HACP (HTTP AICC Communication Protocol) como métodos de comunicación que tendrá el OA para establecer una comunicación con el LMS.
- Aplicar diferentes soluciones de software que permiten la elaboración y edición de SCOs y/o UAs.

Existen en el mercado un gran número de herramientas para la creación de recursos digitales que además ofrecen la posibilidad de convertirlos en objetos bajo la especificación SCORM o AICC como los mencionados en el capítulo 3, subsección 3.3.2.

a) Construcción del prototipo de Software (Creación del OA)

Se siguen los pasos indicados en el subsección 3.3 del capítulo 3 que se ilustran en la Figura 3.16 del mismo.

Básicamente consiste en:

- Construcción del OA con un herramienta de Software (Figura 6.7)
- Etiquetado : proceso de agregación de metadatos (Figura 6.8)
- Empaquetado: agrupación de los distintos contenidos educativos y la descripción de la estructura de estos contenidos (Figura 6.9)

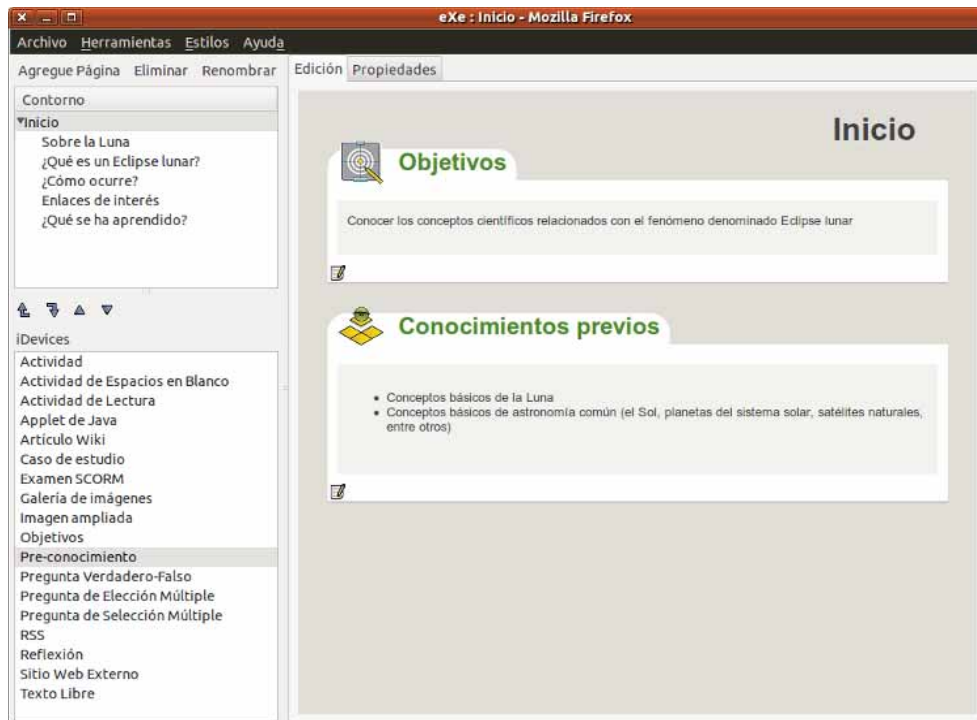


Figura 6.7. Construcción del OA con la herramienta de autor ExeLearning. Tomado de <http://aprendeonline.udea.edu.co>

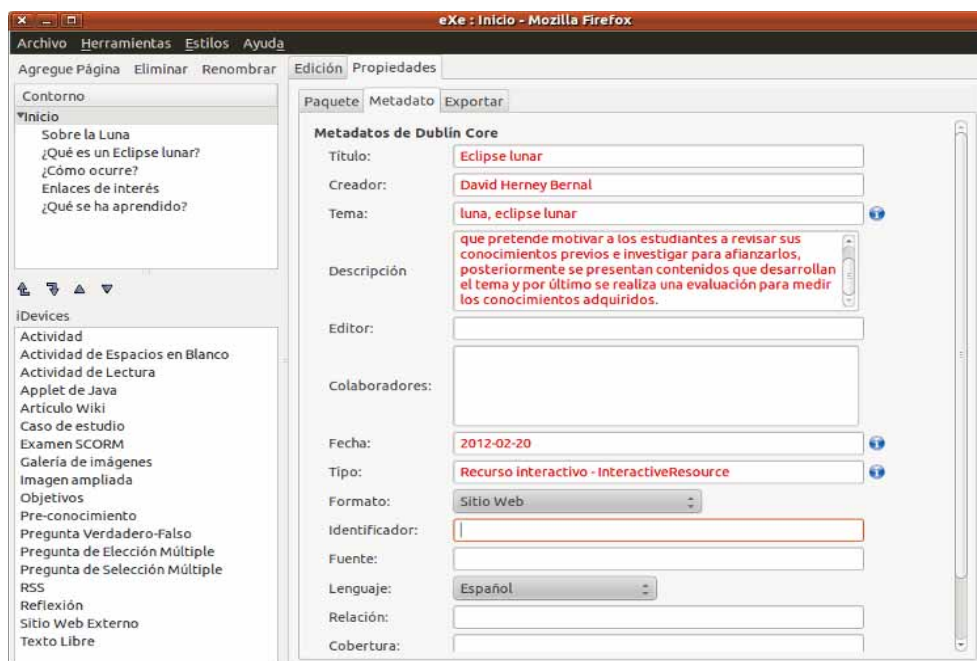


Figura 6.8. Incorporación de Metadatos del OA con la herramienta de autor ExeLearning. Tomado de <http://aprendeonline.udea.edu.co>

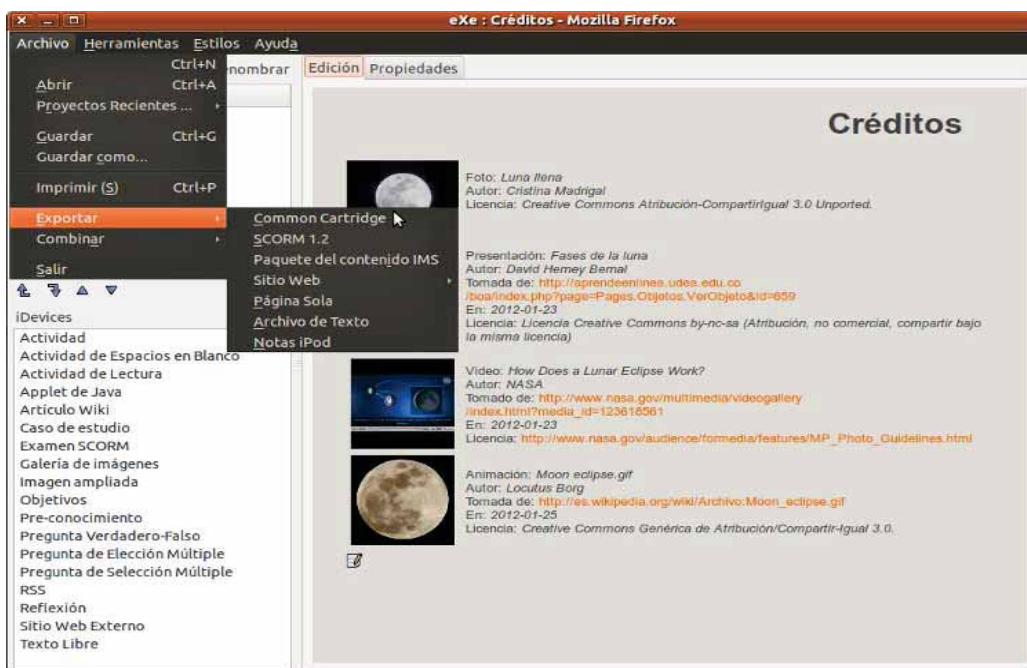


Figura 6.9. Empaquetado del OA con la herramienta de autor ExeLearning. Tomado de <http://aprendeenlinea.udea.edu.co>

En esta etapa se evalúan los atributos del OA mediante la inspección de sus características. El objetivo es determinar si un elemento digital puede ser considerado un OA. Para ello se debe realizar una inspección minuciosa para comprobar compatibilidad con los estándares, granularidad y otros requisitos esenciales de la estructura del OA. Se utilizan los principios de diseño de un OA de Hamel, & Ryan-Jones (2002), como guía y orientación para la inspección:

- Ser una unidad independiente de instrucción de modo que sea capaz de satisfacer un objetivo de aprendizaje individual (se desprende de la definición de OA y de uno de sus atributos: Autocontención conceptual).
- Seguir un formato estándar de instrucción (asociado a los atributos: Interoperabilidad, Escalabilidad y Generatividad).
- Ser relativamente pequeño (relativo a los atributos: Granularidad y la flexibilidad de incorporación en un EVEA por ejemplo).
- Una sucesión de OAs debe tener un contexto (vinculado a la integración en módulos o cursos de un EVEA proporcionando medios para contextualizar la información y al atributo Durabilidad)
- Contener un registro de metadatos (se desprende de la definición y de los atributos : Reusabilidad y Accesibilidad)

b) Implementación del Prototipo de software

La implementación consiste en la importación del paquete (OA etiquetado y empaquetado) a un LMS, en particular nos referimos a un EVEA (Figura 6.10 y Figura 6.11). El objetivo de importar el paquete, es publicarlo en un curso de prueba del EVEA para que este disponible para los evaluadores.

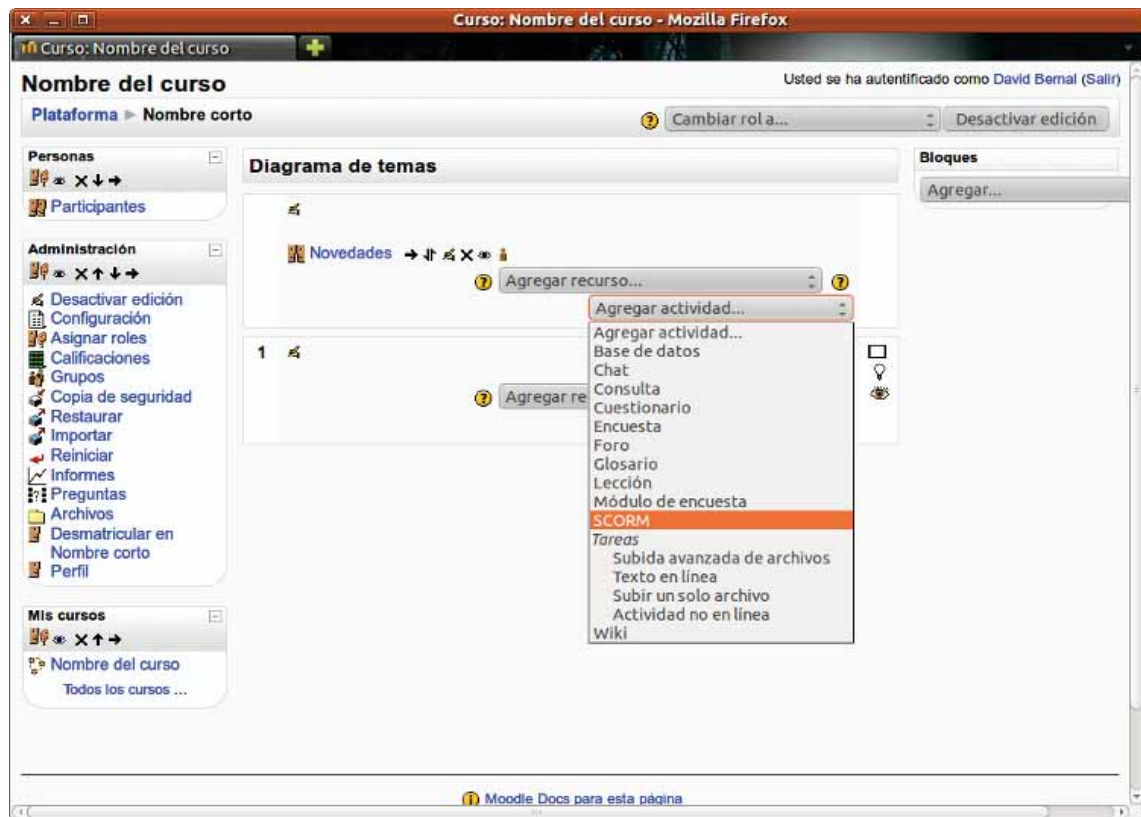


Figura 6.10. Incorporación de un OA (SCORM) en Moodle

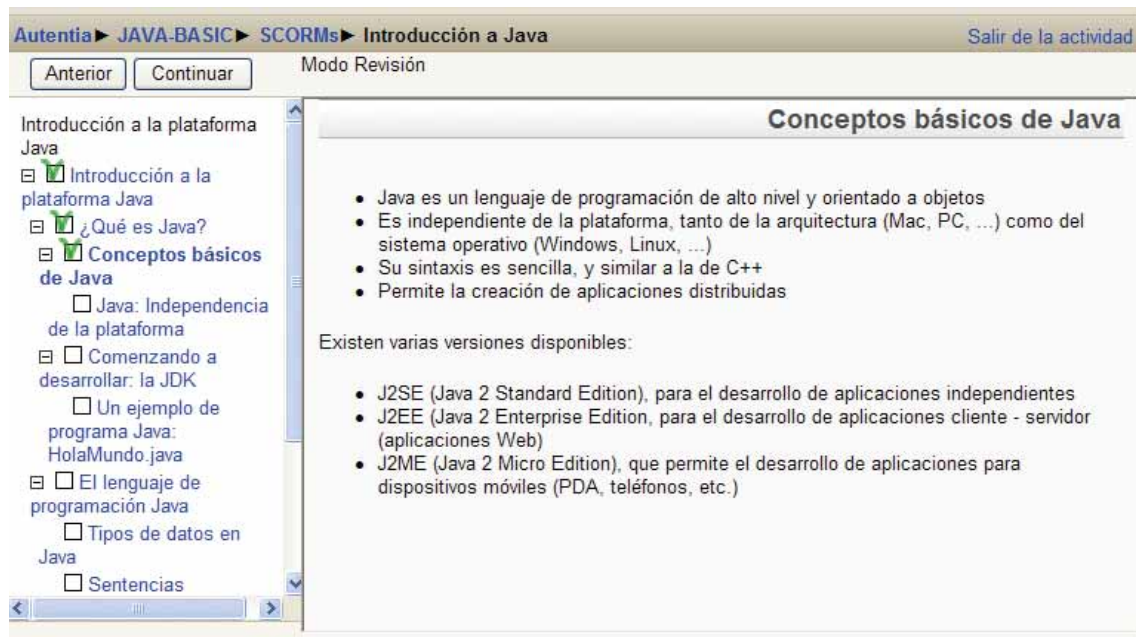


Figura 6.11. Vista de un OA en la Plataforma Moodle

El prototipo de software se implementa después de varias iteraciones de Prototipado-Evaluación y su intención es empezar a ver realmente cómo responde el sistema.

Las aportaciones de evaluar en esta fase son altamente valiosas para no malgastar tiempo en desarrollar software que después deberá seguramente ser cambiado (Granollers, 2004).

En esta etapa del Modelo **MPOBA**, las técnicas de evaluación son: Evaluación Heurística, Observación del experto y Recorrido Cognitivo:

A los evaluadores se les presentan varios escenarios de tareas con el objeto de facilitarle la inspección y que puedan completar el informe de la evaluación. Los expertos son observados durante la ejecución de cada tarea para ver cómo utilizaban la interfaz durante su realización. Se rescatan además los comentarios y observaciones del evaluador, básicamente en cuanto a la forma en la que este "vive" su experiencia con el interfaz, las situaciones que le son incómodas y comportamientos no-naturales y su modelo mental en la interacción con el OA (Hom, 1998).

b-1) Pasos de la evaluación

1. Planificación

i) Adecuación de cada criterio que se utilizará al contexto de uso al cual pertenece el sistema que se desea evaluar.

ii) Selección de los evaluadores:

Respecto al perfil de los revisores, aunque no es imprescindible que sean expertos en usabilidad, diferentes estudios demuestran que conforme más experiencia tengan, mayor será el número de problemas que puedan detectar (González, Lores y Pascual, 2002).

En relación al número de evaluadores, Nielsen y Landauer (1993), Virzi (1992) y Lewis (1994) han publicado artículos sobre el tema del tamaño de la muestra en las pruebas de usabilidad. En estos artículos, los autores presentan un modelo matemático para determinar el tamaño de la muestra para las pruebas de usabilidad. Los autores presentan evidencia empírica de los modelos y realizan varias afirmaciones importantes (Turner, Nielsen & Lewis, 2002):

- La mayoría de los problemas de usabilidad se detectan de 3 a 5 sujetos.
- La ejecución de la misma prueba con sujetos adicionales es poco probable que revele nueva información.
- La mayoría de los problemas de usabilidad graves son detectados por los primeros pocos sujetos.

Es decir, introducir más evaluadores no compensa en cuanto al número de errores detectados y, en cambio, encarece la prueba (Nielsen (2000), Nielsen (2006)).

Se propone la selección de: dos expertos de software relacionado con las tecnologías Web, un consultor de experiencia de usuario, un experto en experiencia de usuario y un especialista del área del contenido.

iii) Elaboración de un paquete de inspección:

Es una colección de heurísticos que ayudarán al experto a realizar el análisis. Hemos definido una Plantilla denominada **GEHOA** (Guía de Evaluación Heurística para Objetos de Aprendizaje) que se detalla en la sección 7.2.2. del capítulo 7.

Durante la prueba, los revisores no sólo deben identificar problemas de usabilidad, sino también ponderar la gravedad de esos problemas, tanto en términos de frecuencia y persistencia del problema, como del impacto o consecuencias que tendrá para el usuario (Manchón, 2003).

iv) Selección de escenarios de tareas para realizar la evaluación.

El análisis se debe limitar a una colección razonable pero representativa de tareas de

prueba. La selección de las tareas se debe basar en el Análisis de Requerimientos. Las tareas de prueba deben ser tan concretas y realistas como sean posibles (Abascal et. al, 2006).

2. Puesta en marcha (evaluación individual con cada experto)

i) Entrenamiento previo a la evaluación:

Se informa al evaluador sobre el tema, contexto de aplicación, perfiles de usuario, descripción del sistema.

ii) Evaluación del OA propiamente dicha por parte del experto:

Se registra la prueba mediante videos y observaciones de campo. Cada experto completa la Plantilla **GEHOA**. Al finalizar cada evaluador expresa en voz alta sus pensamientos, sentimientos y opiniones sobre el OA.

iii) Revisión:

Se analizan cada una de las evaluaciones realizadas para presentar un informe con todos los problemas y sus posibles resoluciones, teniendo en cuenta que el análisis obtenido es cuantitativo y cualitativo.

c) Implantación del Prototipo de Software

En esta etapa se incorpora el paquete (OA etiquetado y empaquetado) a un Curso del EVEA para que este disponible para estudiantes y de esta manera se produzca un proceso de interacción del OA con los usuarios finales.

En esta fase el factor más importante es la reacción o retraining del usuario (User Feedback) (Mayhew, 1999) que proporciona información para:

- El mantenimiento y posibles mejoras del producto.
- La implementación de futuras revisiones del producto.
- El diseño y desarrollo de productos relacionados que serán utilizados por los mismos usuarios o de características similares.
- Ayuda a incrementar el autoaprendizaje en cuanto a la usabilidad mejoras de los ya adquiridos.

Se pretende conocer si el OA ha sido adecuado como unidad de enseñanza, es decir,

valorar su calidad desde el punto de vista de los estudiantes con preguntas relacionadas a su satisfacción con respecto a los contenidos, actividades y evaluación (usabilidad pedagógica), diseño de la interfaz y navegación (usabilidad web).

A continuación se detallan las técnicas de evaluación utilizadas en la implantación del prototipo de software:

c-1) Cuestionarios

Este instrumento es utilizado para la evaluación de la calidad del OA desde el punto de vista del usuario final (el estudiante). También el objetivo es obtener información cuantitativa y cualitativa sobre el grado de satisfacción de los estudiantes al interactuar con el OA.

Procedimiento

i) Se le solicita a los estudiantes que recorran completamente el OA (incluido en el curso del EVEA) y tomen nota de sus observaciones e interacciones.

ii) Los estudiantes cumplimentan el cuestionario que contiene:

- Una conjunto de ítems bajo la forma de afirmaciones o juicios ante los cuales se solicita expresen su acuerdo o desacuerdo (Escala de Likert).
- Una lista de adjetivos que debe de relacionar con el OA (Escala de Diferencial Semántico).
- Preguntas abiertas para recoger información general subjetiva que pueden dar sugerencias interesantes y encontrar errores no previstos.

Los detalles del cuestionario denominado **CUSEOA** (Cuestionario de satisfacción de estudiantes de un OA) se presentan en la sección 7.3. del capítulo 7.

iii) Se analizan las respuestas del cuestionario y se construye un informe con las percepciones de los estudiantes, los problemas detectados y sus posibles resoluciones.

c-2) Grabación de uso:

Típicamente, los EVEAS contienen un registro de la interfaz y estadísticas sobre la frecuencia con la que cada usuario ha utilizado el OA y la frecuencia con que los diversos eventos de interés han ocurrido (Figura 6.12).

El procedimiento consiste en la captura de los registros almacenados en el EVEA de cada

estudiante que interactuó con el OA. Posteriormente se procede a cruzar esta información con los resultados del Cuestionario a fin de rescatar información más detallada de sus percepciones y analizar las respuestas de cada estudiante en el Cuestionario.

aweflauwh					
Name	attempt	Started on	Last accessed on	Score	
<input type="checkbox"/> Remote-Learner Admin	1	Wednesday, December 9, 2009, 11:36 AM	Wednesday, December 9, 2009, 11:40 AM	1	
<input type="checkbox"/> Miss Rolly	1	Tuesday, December 8, 2009, 01:07 PM	Tuesday, December 8, 2009, 01:58 PM	1	
<input type="checkbox"/> Mike's Test User	1	Wednesday, December 9, 2009, 11:28 AM	Wednesday, December 9, 2009, 11:52 AM	1	

Select all / Deselect all: With selected

Figura 6.12. Reportes de Moodle de un OA (SCORM)

c-3) Entrevistas en profundidad a estudiantes seleccionados que participaron en las dos instancias anteriores de esta fase a los efectos de contrastar, comprender y describir información recopilada mediante un análisis retrospectivo de la acción.

d) En producción

El OA sufrirá cambios a lo largo de su vida útil. Estos cambios pueden producirse por errores no detectados en las etapas anteriores, que se produzcan cambios en algunas de sus componentes o se requieran modificaciones funcionales no contempladas anteriormente.

6.2.1.6. La hoja de trabajo del modelo MPOBA

Dada la complejidad del modelo de proceso descripto, surge la necesidad de representar gráficamente una guía que refleje la evolución de los cambios producidos y que a la vez permita a todos los integrantes del equipo de desarrollo del OA, ver qué actividades del Modelo de Proceso se están llevando a cabo, cuándo lo están haciendo y qué relación tienen con las demás.

Este instrumento no reemplaza a la planificación temporal: PERT, CPM o un Diagrama de Gantt de un proyecto, sino que aporta otros elementos al proceso de desarrollo del OA.

La secuencia temporal seguida puede comprobarse en una matriz de actividades, denominada “Hoja de Trabajo” (Figura 6.13) que refleja, en orden cronológico, todas las actividades a realizar.

El esquema fue propuesto por Granollers (2004) en su tesis doctoral y dispone de una serie de columnas para cada una de las fases del Modelo **MPOBA**. Las flechas indican qué tarea precede o sigue a la tarea actual.

El objetivo además, es visualizar que desde todas las fases se promueve la participación activa de los usuarios, tanto en el Análisis de Requerimientos como en el Diseño y en la realización de prototipos y/o su posterior evaluación.

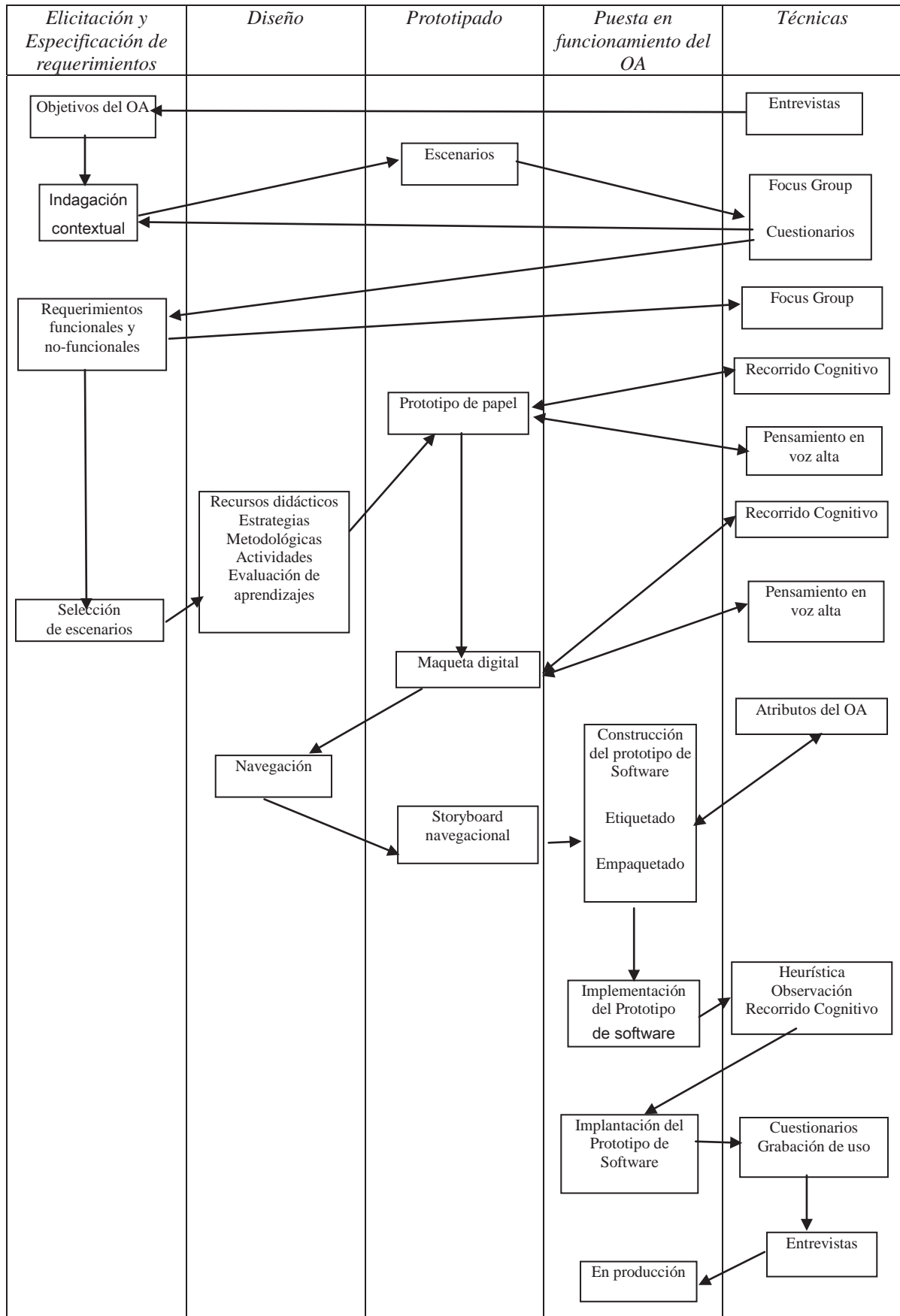


Figura 6.13. Hoja de trabajo del modelo MPOBA

CAPITULO 7. INSTRUMENTOS E INDICADORES

En este capítulo se presentan los instrumentos desarrollados en el marco del Modelo de Proceso MPOBA correspondientes a la evaluación del OA por parte de expertos y estudiantes y sus indicadores de calidad respectivos. En la sección 7.2, se realiza una revisión bibliográfica de los instrumentos utilizados para evaluación por expertos y los trabajos relacionados, se detallan las características de la Plantilla GEHOA (Guía de Evaluación Heurística para OA), el procedimiento para su aplicación, se identifican las dimensiones críticas a evaluar y los indicadores de calidad. Finalmente, en la sección 7.2 se realiza una revisión bibliográfica de los instrumentos utilizados para evaluación por estudiantes y los trabajos relacionados, se detallan las características de Instrumento CUSEOA (Cuestionario de satisfacción de Estudiantes de un OA), el procedimiento para su aplicación, se identifican las dimensiones críticas a evaluar y los indicadores de calidad

7.1. Introducción

En la sección 5.4. del Capítulo 5 se realizó una revisión de los métodos mas utilizados para la evaluación de la usabilidad. Se presentó además una clasificación de acuerdo al lugar, técnica, mecanismo y participantes.

Existe otra clasificación basada en la forma de trabajo del método que los clasifica en dos categorías: analítica y empírica.

Los métodos analíticos son realizados por expertos en usabilidad que se ponen en la posición de usuarios (Blecken, Bruggemann & Marx, 2010). Estos procedimientos son los más adecuados para las evaluaciones preliminares durante la fase de desarrollo del sistema. Los métodos de inspección como “Evaluación Heurística”, “Recorrido Cognitivo”, son ejemplos de esta categoría,

Los métodos empíricos requieren de pruebas con usuarios. Se componen principalmente de test de usabilidad y cuestionarios. Suelen emplearse tanto con un prototipo o con el sistema ya en uso. Su objetivo es determinar la utilidad global del sistema (Blecken, Bruggemann & Marx, 2010).

Los métodos analíticos no deben reemplazar a los métodos empíricos y viceversa, sino que deben complementarse para llega a una evaluación holística.

Esa visión integral de la evaluación, es manifestada por Squires (1999) para recursos educativos en un sitio Web. Este autor propone combinar las nociones clásicas de usabilidad y de los procesos de enseñanza y de aprendizaje, dado que un recurso de esta naturaleza no sólo debe ser “usable” en su diseño pues es determinante su valor pedagógico.

Otros autores coinciden en esta idea, señalando que un recurso educativo en un sitio Web puede ser muy válido desde la perspectiva de la usabilidad técnica, pero puede no tener ningún valor pedagógico, y a la inversa (Quinn (1996), Albion (1999), Squires & Preece (1999)).

Además, Squires (1999) hace hincapié en la necesidad de que los profesionales de la Interacción Persona-Ordenador (IPO) tengan en cuenta los avances de las teorías del aprendizaje y los especialistas en tecnología educativa en la usabilidad.

Los profesionales de IPO y de las áreas de tecnología educativa rara vez hablan el uno al otro o toman nota de los trabajos de cada uno: la literatura sobre tecnología educativa está colmada de interpretaciones ingenuas y simplistas de los problemas de diseño de interfaz, y muchos escritores de la literatura IPO parecen no ser conscientes de los importantes acontecimientos que se han hecho en las teorías del aprendizaje. (Squires, 1999, p.464).

En definitiva, para avanzar en el estudio de los atributos de usabilidad en un OA (como recurso educativo en un sitio Web), se debe investigar a las formas en que las características específicas de diseño de interfaz o navegación tienen un impacto en las tareas u objetivos planteados en el OA.

Al respecto, Feldstein (2002) plantea una serie de preguntas que deberían tenerse en cuenta a la hora de evaluar los atributos de usabilidad de recursos educativos en un sitio Web.

- ¿Hasta qué punto los mapas de sitio, menús en pantalla, y tablas de contenido ayudan a los estudiantes interiorizar y recordar la estructura de los contenidos?
- ¿Hasta qué punto los mapas de sitio, menús en pantalla, y tablas de contenido ayudan a los estudiantes para encontrar los conceptos claves como apoyo a su aprendizaje en los momentos de revisión o repaso?
- ¿La velocidad de narración de un texto (un narrador lee las mismas palabras que están en la pantalla) afectan la capacidad del alumno para recordar hechos y conceptos clave?
- ¿La narración de un texto afectan la capacidad del alumno para procesar

conceptos complejos?

- ¿La interfaz de un debate o discusión (por ejemplo en un foro, el hilo de la discusión) afectan a la frecuencia con que los alumnos "cosechan" determinadas ideas o hechos?
- ¿La interfaz de un debate o discusión afectan la frecuencia con que los estudiantes sintetizan diversos puntos de vista en una conversación?

A continuación se describen los instrumentos desarrollados en el marco del Modelo de Proceso **MPOBA** correspondientes a la evaluación del OA por parte de expertos y estudiantes y sus indicadores de calidad respectivos.

7.2. Instrumento para la evaluación por expertos

7.2.1. Trabajos relacionados

a) Investigaciones

En relación a los recursos educativos en un sitio Web, Feldstein (2002) considera que una solución parcial a la creación de tests de usabilidad simples y baratos, se puede encontrar en la “Evaluación Heurística” utilizando los principios heurísticos con precaución, ya que muchas suposiciones acerca de los usuarios del comercio electrónico no se aplican a los alumnos de cursos de e-learning. El reto es desarrollar “directrices pedagógicas” que se puedan combinar por ejemplo con los principios heurísticos de Nielsen (1994a).

También Squires & Preece (1999) afirman que la simple aplicación de estas heurísticas no podrían ser eficaces porque no tienen en cuenta los problemas específicos de aprendizaje centrado en el diseño de interfaces, así como la cuestión de la integración de la usabilidad y el aprendizaje.

Squires & Preece (1999) proponen una adaptación de los principios de Nielsen, teniendo en cuenta el los principios de la teoría Socio – constructivista (Vygotsky, 1978). El conjunto de heurísticas relacionadas con "aprender con el software" contiene lo siguiente:

- Coincidencia entre los modelos de diseño y modelo mental del alumno.
- La fidelidad de navegación.
- Los niveles apropiados de control del aprendiz.
- Prevención de errores cognitivos periféricos.
- Representaciones simbólicas comprensibles y significativas.

- Apoyar los enfoques personalmente significativos para el aprendizaje.
- Estrategias para el reconocimiento cognitivo de error, el diagnóstico y la recuperación.

b) LORI

Uno de los instrumentos más conocidos para evaluar OAs es LORI (Learning Object Review Instrument). Esta herramienta fue creada por Nesbit, Belfer & Leacock (2009) y utilizada para evaluar la calidad de recursos para un sistema e-learning.

LORI ha sido diseñado como un instrumento de Evaluación Heurística. Como tal, no contiene exhaustivas listas de comprobación detalladas y no se ocupa de todas las eventualidades posibles en el diseño de OA. Por el contrario, LORI identifica nueve dimensiones críticas de la calidad (Tabla 7.1), que abarca las cuestiones pedagógicas, tecnológicas, y los factores de la experiencia del usuario (Leacock & Nesbit, 2007)

Item	Descripción
Calidad de los contenidos	Veracidad, exactitud, presentación equilibrada de ideas y nivel adecuado de detalle.
Adecuación de los objetivos de aprendizaje	Coherencia entre los objetivos, actividades, evaluaciones, y el perfil del alumnado.
Feedback (retroalimentación) y adaptabilidad	Contenido adaptativo o feedback dirigido en función de la respuesta de cada alumno/a y su estilo de aprendizaje.
Motivación	Capacidad de motivar y generar interés en un grupo concreto de alumno/as.
Diseño y presentación	El diseño de la información audiovisual favorece el adecuado procesamiento de la información.
Usabilidad	Facilidad de navegación, interfaz predictiva para el usuario y calidad de los recursos de ayuda de la interfaz.
Accesibilidad	El diseño de los controles y la presentación de la información están adaptados para discapacitados y dispositivos móviles.
Reusabilidad	Capacidad para usarse en distintos escenarios de aprendizaje y con alumno/as de distintos bagajes.
Cumplimiento de estándares	Adecuación a los estándares y especificaciones internacionales.

Tabla 7.1. Descripción de las 9 dimensiones de LORI. Fuente: Nesbit, Belfer & Leacock (2009).

Los evaluadores proporcionan puntuaciones para cada ítem en una escala de 0 a 4. En caso de que un evaluador no tenga la capacidad de evaluar un ítem puede obviarlo. También pueden incluir comentarios adicionales sobre cada elemento de texto.

El Manual de LORI (Nesbit, Belfer & Leacock, 2009) proporciona información sobre la forma de interpretar cada uno de los nueve puntos, incluyendo una descripción más detallada y ejemplos de factores que podrían llevar a asignar una puntuación en un ítem determinado (Figura 7.1). Los evaluadores además tienen acceso a esta información cuando llevan a cabo esta evaluación on line en el portal Elera (<http://www.elera.net/eLera/Home>)

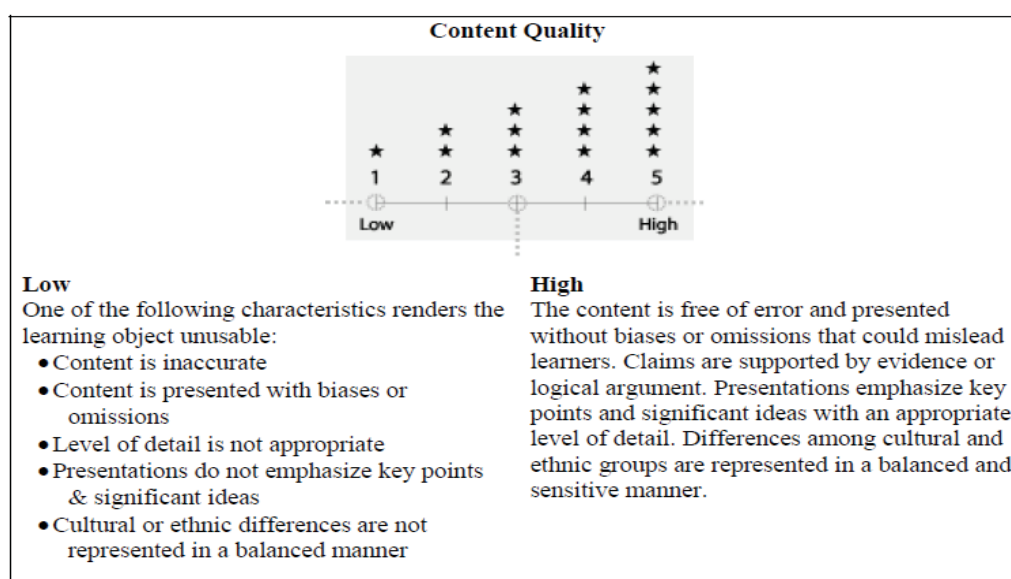


Figura 7.1. Extracto de la rúbrica “Calidad de los contenidos” de LORI. Fuente: Leacock & Nesbit, (2007).

Este modelo puede ser utilizado en la revisión individual, autoevaluación, o en grupo, evaluación por pares. En Nesbit & Leacock (2009) se presenta una posible propuesta de revisión por pares colaborativa en la que participan tanto expertos como técnicos.

c) HEODAR

HEODAR (Herramienta de Evaluación de Objetos Didácticos de Aprendizaje Reutilizables) es un instrumento desarrollado en la Universidad de Salamanca (Morales, Gómez y García Peñalvo, 2008). Esta herramienta fue diseñada tomando en cuenta criterios específicos desde un punto de vista pedagógico y técnico, lo cual es el resultado

de la revisión de diversas propuestas de evaluación de recursos educativos, como también un análisis comparativo con la herramienta de evaluación de OAs: LORI.

Los autores parten de la necesidad de valorar la significatividad lógica y psicológica en los OA. La primera referida al alumno y la segunda al currículo y al material de estudio. La estructura de la escala está basada en dos aspectos:

- **Aspectos pedagógicos**, subdivido, a su vez, en dos categorías: **psicopedagógica** y **didáctico-curricular**. La primera incluye criterios referidos a: motivación y atención, desempeño profesional, nivel de dificultad, interactividad, y creatividad; mientras que en la segunda los criterios se refieren a: contexto, objetivos, tiempo de aprendizaje, contenidos, actividades y retroalimentación.
- Aspectos de **usabilidad**, el cual contempla el **diseño de interfaz** y el **diseño de navegación**. El primero toma en cuenta criterios para valorar: texto, imagen, animaciones, multimedia, sonido y video. Finalmente, el segundo, se refiere a criterios para la página de inicio y navegabilidad.

Para valorar cada uno de los criterios se utiliza una escala del 1 al 5. La escala de evaluación es igual en LORI, y además presenta un ítem de comentarios con observaciones cualitativas.

HEODAR se ha implementado de modo experimental a través de una herramienta software en la plataforma Moodle (Morales et. al. 2010) de la Universidad de Salamanca (“Studium” <http://studium.usal.es>) con el fin de ser utilizada para mejorar la calidad de los materiales didácticos que ahí se gestionan y aumentar las posibilidades de la propia herramienta.

Comparando LORI con HEODAR arribamos a las siguientes conclusiones:

- LORI es fácil de aplicar y se adapta para OAs pequeños y sencillos.
- HEODAR contiene una evaluación más exhaustiva , más adecuada para OA más complejos.
- Las categorías (dimensiones) de HEODAR son más finas que las del LORI. En este sentido, si se requiere evaluar la calidad de aspectos más precisos o específicos, HEODAR sería de más ayuda.
- HEODAR contempla a nivel pedagógico más elementos que LORI.
- HEODAR debería pasar por un proceso de observación y validación empírica

por más tiempo como lo hizo LORI.

7.2.2. Plantilla GEHOA (Guía de evaluación heurística para Objetos de Aprendizaje)

En la subsección 6.1.1.5. : Puesta en funcionamiento del OA – Implementación del Prototipo de software del Capítulo 6, se mencionó que uno de los métodos de evaluación por expertos que se utilizará es la “Evaluación Heurística” con la Plantilla **GEHOA**, cuyas características serán explicadas en esta subsección.

El modelo de Plantilla considera dos dimensiones: pedagógicas y técnicas.

a) Dimensión pedagógica

Para la selección de las dimensiones pedagógicas se parte de la definición de “usabilidad pedagógica” presentada en el capítulo 5: “**usabilidad pedagógica**”: facilidad de aprendizaje, eficiencia de uso pedagógico y la satisfacción con las que las personas son capaces de realizar sus tareas gracias al uso del producto con el que está interactuando. Entendemos eficiencia de uso pedagógico como la capacidad de propiciar aprendizajes significativos mediante interacciones generadas en la Zona de Desarrollo Próximo (ZDP).

El aprendizaje significativo, por recepción o por descubrimiento, comprende la adquisición de nuevos significados.

Ausubel destaca dos dimensiones del material potencialmente significativo:

- Significatividad lógica: coherencia en la estructura interna del material, secuencia lógica en los procesos y consecuencia en las relaciones entre sus elementos componentes.
- Significatividad psicológica: conexión del conocimiento nuevo con el previo. Se refiere a que los contenidos sean comprensibles desde la estructura cognitiva que posee el sujeto que aprende.

Por otro lado, Ausubel (1983) agrega que el alumno “quiera aprender” no basta para que se dé el aprendizaje significativo, pues también es necesario que “pueda aprender” (significación lógica y psicológica del material). Sin embargo, el aprendizaje no puede darse si el alumno no quiere aprender. Este es un componente que se refiere a la disposición emocional y actitudinal, que puede abordarse desde la motivación.

En cuanto a las interacciones generadas en la zona de desarrollo próximo, Vygotsky (1978) no se refiere a actividades genéricas, sino a situaciones de resolución de problemas (problem solving) y distingue dos formas de esta actividad:

- problem solving autónomo, es decir afrontado por el estudiante autónomamente,

- problem solving colaborativo, es decir afrontado por el estudiante bajo la guía de un adulto o en colaboración con un compañero que ha alcanzado un mayor éxito en la actividad.

Por su parte, Coll et. al. (1999), enuncia y recomienda seguir una serie de criterios para a la creación de ZDP en la interacción profesor/estudiantes:

- Insertar actividades en el ámbito de marcos u objetivos más amplios de manera que tome significado de manera más adecuada. Supone, además, informar del conjunto de contenidos y actividades que se realizarán y de la relación entre ellas.
- Proveer una Ayuda ajustada (combinación de los conocimientos previos del estudiante y del apoyo e instrumentos): se relaciona con la variación y diversidad de actividades presentadas y la variación y diversidad en las formas de ayuda, es decir ofrecer un feedback personalizado.
- Posibilitar la participación de todos los estudiantes en las distintas actividades y tareas. Plantear actividades con opciones o alternativas o niveles. Recurrir a materiales en diversos formatos. Estimular el planteo de cuestiones, puntos de vista alternativos, etc.
- Promover la utilización y profundización autónoma de los conocimientos. Generar actividades con distintos niveles de ayuda. Fomentar el aprendizaje de estrategias y habilidades que les permitan seguir aprendiendo de manera autónoma, controlando y regulando de manera más eficaz sus propios procesos de aprendizaje.
- Utilizar el lenguaje de la manera más clara y explícita posible, tratando de evitar y controlar posibles malentendidos o incomprensiones. Definir, ejemplificar o caracterizar los términos nuevos; establecer relaciones explícitas entre conceptos, principios, procedimientos, normas.

b) Dimensión técnica

Diferentes criterios han sido desarrollados para la evaluación de software y diseño de la interfaz. En este caso se parte de los principios heurísticos.

Los principios heurísticos son criterios que pretenden aplicar normas conversacionales a la interacción entre una persona y la interfaz de un sistema o producto, de modo que éstos se entiendan y trabajen juntos de forma efectiva.

El origen de los principios heurísticos entendidos como tales fue presentado por Molich y Nielsen (1990). Aunque, Shneiderman (1986), presentó antes “ocho reglas de oro” para lograr una correcta interacción en una interfaz.

Luego Nielsen (1994b) extrajo del análisis factorial de 249 problemas de usabilidad un conjunto de principios heurísticos.

Los diez principios heurísticos propuestos por Nielsen son los siguientes:

1. Visibilidad del estado del sistema: el sitio web o aplicación debe siempre mantener informado al usuario de lo que está ocurriendo y proporcionarle respuesta en un tiempo razonable.
2. Adecuación entre el sistema y el mundo real: el sitio web o la aplicación debe utilizar el lenguaje del usuario, con expresiones y palabras que le resulten familiares. La información debe aparecer en un orden lógico y natural.
3. Libertad y control por parte del usuario: en caso de elegir alguna opción del sitio web o aplicación por error, el usuario debe disponer de una salida de emergencia claramente delimitada para abandonar el estado no deseado en el que se halla sin tener que mantener un diálogo largo con el sitio o aplicación. Debe disponer también de la capacidad de deshacer o repetir una acción ejecutada.
4. Consistencia y estándares: los usuarios no tienen por qué saber que diferentes palabras, situaciones o acciones significan lo mismo. Es conveniente seguir convenciones.
5. Prevención de errores: es importante prevenir la existencia de errores mediante un diseño adecuado. Aun así, los mensajes de error deben incluir una confirmación antes de ejecutar las acciones de corrección.
6. Reconocimiento antes que recuerdo: hacer visibles objetos, acciones y opciones para que el usuario no tenga por qué recordar información entre distintas secciones o partes del sitio Web o aplicación. Las instrucciones de uso deben estar visibles o ser fácilmente localizables.
7. Flexibilidad y eficiencia en el uso: los aceleradores o atajos de teclado pueden hacer más rápida la interacción para usuarios expertos, de tal forma que el sitio web o aplicación sea útil tanto para usuarios noveles como avanzados. Debe permitirse a los usuarios configurar acciones frecuentes con atajos de teclado.
8. Diseño estético y minimalista: las páginas no deben contener información irrelevante o innecesaria. Cada información extra compite con la información relevante y disminuye su visibilidad.

9. Ayuda a los usuarios a reconocer, diagnosticar y recuperarse de los errores: los mensajes de error deben expresarse en un lenguaje común y sencillo, que indique con precisión el problema y sugiera las posibles alternativas o soluciones.
10. Ayuda y documentación: aunque es mejor que el sitio Web o aplicación pueda ser usado sin documentación, puede ser necesario suministrar cierto tipo de ayuda. En ese caso, la ayuda debe ser fácil de localizar, tiene que especificar los pasos necesarios y no debe ser muy extensa.

Pocos autores, después de Nielsen, consiguieron aportar principios radicalmente distintos; en general las adaptaron en función de la aplicación o sistema o introdujeron matices sobre los ejes que él ya había planteado (Constantine (1994), Instone (1996) Mayhew (1999), Tognazzini (2007)).

La popularidad de estos heurísticos también ha causado el malentendido y la creencia de que tenerlos en cuenta es suficiente para que un sitio web sea usable (Zapata, 2011).

Los criterios heurísticos elaborados para la Plantilla GEHOA se focalizaron en las dos dimensiones: técnica y pedagógica.

Los criterios para la dimensión técnica se basaron en los diez principios heurísticos de Nielsen (1994a), las ocho reglas de oro para el diseño de interfaces descritas por Schneidermann (1986), la adaptación a la Web de los principios de Instone (1996), los principios para el diseño de sistemas centrados en el usuario de Mayhew (1999) y las recomendaciones de Granollers (2004).

Los criterios para la dimensión pedagógica se crearon de acuerdo a las características referidas al Aprendizaje significativo y la Creación de Zona de desarrollo Próximo.

Cada criterio heurístico contiene una serie de sub-reglas (o sub-heurísticas) Las subheurísticas son las preguntas que detallan las características de los criterios, que suelen ser bastante generales.

La selección de las sub-heurísticas se realizará teniendo en cuenta la aplicación o sistema a evaluar. Una vez seleccionados se procede a adecuarlos al contexto de uso (en este caso los OA). Esta adaptación y sus problemáticas son estudiadas por la Usabilidad Transcultural (Harrison & McKinnon, 1999).

La personalización de los criterios le dará mayor valor a la evaluación y orientará al experto en la misma.

En el caso de la dimensión pedagógica, las sub-heurísticas de la plantilla **GEHOA** son el resultado de una revisión de varias propuestas de modelos de evaluación de la calidad

recursos educativos digitales desarrollados en Universidades , entre las que destacan la UNED (García Aretio, Ruíz y Domínguez, 2007; MADI); Universidad Complutense de Madrid (Fernández-Pampillón, Domínguez Romero y de Armas Ranero, 2012); Universidad de Salamanca (Morales, Gómez y García Peñalvo, 2008); Universitat Oberta de Catalunya (Mor, Garreta, Galofré, 2007); Universidad Autónoma de Aguas Calientes (Ruiz González, Muñoz Arteaga y Álvarez Rodríguez, 2007); Instituto Tecnológico de Estudios Superiores de Monterrey (Cruz Meléndez, Alfaro Rivera y Ramírez Montoya, 2012). Asimismo, conjuga varios de los criterios comúnmente compartidos por la mayoría de los modelos de evaluación de OA publicados (Becta (2007) , Leacock & Nesbit (2007), DESIRE (2000), Kurilovas & Dagiene (2011), MELT (2007) y Q4R(2007)).

Cada una de las dimensiones se dividió en categorías transversales: técnica y pedagógica.

En el caso de la técnica se tomaron dos aspectos básicos a evaluar sobre el diseño de los OAs: diseño de interfaz y Estructura y Navegación. En cuanto a la pedagógica las categorías asociadas a la “usabilidad pedagógica”: Significatividad psicológica y Significatividad lógica. Cada categoría contiene los criterios heurísticos asociados. Finalmente se definen las sub-heurísticas involucradas a cada criterio (Ver Tabla 7.2).

	CATEGORÍA	CRITERIOS HEURÍSTICOS	SUB-HEURISTICAS
USABILIDAD PEDAGÓGICA	SIGNIFICATIVIDAD PSICOLÓGICA	Motivación	El contenido es relevante para los objetivos profesionales o personales e intereses del estudiante. Hay elementos que suponen captar el interés del alumno y no resultan ser una distracción que interfiere con el aprendizaje.
		Competencias	Está orientado al desarrollo de competencias genéricas (transversales) y/o específicas. Se proponen diferentes contenidos/actividades para cada tipo/nivel de competencia de alumno.
		Interacción e Interactividad	Se propicia el intercambio con el tutor y/o en forma grupal. Se induce a la participación directa por parte de los estudiantes en diversas actividades.
		Conocimientos previos	La profundidad es adecuada según los conocimientos previos y el nivel de complejidad que el estudiante es capaz de comprender. Se fomenta la capacidad de relacionar conceptos ya aprendidos con los nuevos conceptos.
		Innovación y autonomía	Se promueve la creación de nuevas ideas y la búsqueda de nuevos procedimientos/técnicas/métodos para la resolución de problemas o tareas. Contiene pautas, guías o actividades que fomentan el desarrollo de habilidades metacognitivas y estrategias de aprendizaje que le permita al estudiante planificar, regular y evaluar su propia actividad intelectual.

Tabla 7.2a. Dimensiones, categorías y subheurísticas

	CATEGORÍA	CRITERIOS HEURÍSTICOS	SUB-HEURISTICAS
USABILIDAD PEDAGÓGICA	SIGNIFICATIVIDAD LÓGICA	Objetivos	Se indica lo que se espera sea aprendido. Los objetivos están correctamente formulados. Los objetivos son factibles de ser alcanzados.
		Contenidos	El tiempo de duración estimado en el desarrollo del OA es excesivo. Presenta información suficiente y adecuada al nivel educativo. Los contenidos son coherentes con los objetivos propuestos. Presenta el material de apoyo en un único formato (texto, audio, etc.). Permite interactuar con el contenido a través de enlaces. Proporciona información complementaria para ayudar a los alumnos que deseen profundizar sus conocimientos. Cuida que la información que presenta sea confiable, (datos exactos, referencias bibliográficas, etc.). El contenido no presenta errores, sesgos u omisiones que pudiera confundir o llevar a errores. Se respetan los derechos de propiedad intelectual. Los contenidos se introducen en forma secuenciada para ayudar a una mejor comprensión del mismo. El lenguaje empleado en los contenidos es pertinente a los objetivos y al nivel educativo. Se utilizan múltiples estrategias de enseñanza: resúmenes, ilustraciones, señalizaciones, gráficos, mapas conceptuales, organizadores textuales.
		Actividades	Las actividades complementan las del curso presencial. Se presentan variadas actividades: resolución de problemas, estudio de caso, glosario, mapa conceptual, etc. Presenta actividades de evaluación y práctica. La mayor parte de las actividades se corrigen adecuadamente de manera automática. En las actividades existe posibilidad de que el estudiante reciba feedback personalizado relativo a la calidad o corrección de sus respuestas.

Tabla 7.2b. Dimensiones, categorías y subheurísticas

	CATEGORÍA	CRITERIOS HEURÍSTICOS	SUB-HEURISTICAS
USABILIDAD DE SITIOS WEB	DISEÑO DE INTERFAZ	Lenguaje de los usuarios	<p>No se rompen los párrafos ni la continuidad de las ideas que se exponen en ellos.</p> <p>Marca bloques de contenido a través de títulos o epígrafes</p> <p>La presencia de las imágenes es superflua.</p> <p>Las animaciones atraen la atención del usuario para destacar puntos claves e ideas significativas.</p> <p>La presencia de audio no es superflua, proporciona algún tipo de valor añadido.</p> <p>Los videos proporcionan algún tipo de valor añadido.</p> <p>Las imágenes aclaran la información textual.</p>
		Control y libertad del usuario	<p>Existe un vínculo que permite volver a la página de inicio o retroceder en una acción.</p> <p>La interfaz no requiere tecnologías que necesitan versiones actualizadas de los navegadores o plug-ins externos. Si esto no es así, se describen los requisitos informáticos necesarios.</p> <p>En los recursos multimediales (imágenes, sonido, video) se indica el tiempo de descarga.</p> <p>Las animaciones pueden ser controladas por el usuario (avanzar, detener, etc.)</p>
		Diseño estético y minimalista	<p>Las fuentes son legibles y tienen un tamaño adecuado</p> <p>Los colores y tipos de letras aportan información por sí mismos.</p> <p>Las fuentes utilizan colores con suficiente contraste con el fondo.</p> <p>Las imágenes son claras, se ha cuidado su resolución.</p> <p>Las animaciones proporcionan algún tipo de valor añadido.</p> <p>En los videos, la imagen y el audio se presentan de forma clara.</p>

Tabla 7.2c. Dimensiones, categorías y subheurísticas

	CATEGORÍA	CRITERIOS HEURÍSTICOS	SUB-HEURISTICAS
USABILIDAD DE SITIOS WEB	ESTRUCTURA Y NAVEGACIÓN	Visibilidad del estado del sistema	La página de inicio aclara al usuario dónde se encuentra y el objetivo del sitio. La página de Inicio presenta las principales áreas de contenido del sitio con enlaces para acceder a ella. El usuario sabe dónde se encuentra en todo momento.
		Consistencia y Estándares	El diseño es consistente en todas las pantallas (tamaños, colores, iconos, tipos de letra, etc.). Las áreas de navegación superior, laterales, herramientas de búsqueda y controles (botones, radio buttons,...) siguen los estándares comunes del mercado. Las mismas acciones llevan a los mismos resultados.
		Reconocimiento más que memoria	Utiliza el hipertexto para dividir información extensa en múltiples páginas. El sitio posee una estructura flexible y familiar que permite al usuario controlar su navegación. Los títulos son breves y descriptivos. Los enlaces son fácilmente reconocibles, están claramente diferenciados.
		Flexibilidad y eficiencia de uso	Se utiliza un diseño líquido para que la página se adapte a las diferentes resoluciones posibles que puede tener un usuario.
		Navegación visible	Las pantallas dedican en gran parte espacio al contenido. La información visible es la única esencial para realizar la acción. Las páginas no contienen información que es irrelevante o raramente necesaria.
		Errores	Cuando se produce un error, se informa de forma clara al usuario de lo ocurrido y de cómo solucionar el problema.

Tabla 7.2c. Dimensiones, categorías y subheurísticas

Cada criterio se valora según la escala de Nielsen (1994d)

0 = No estoy de acuerdo que este es un problema de usabilidad en absoluto.

1 = cosmética único problema: no tiene por qué solucionarse a menos que el tiempo adicional está disponible en el proyecto.

2 = problema de usabilidad menor: solucionar esto debe ser de baja prioridad.

3 = problema de usabilidad mayor: importante para solucionar, por lo que se debe dar alta prioridad.

4 = catástrofe de Usabilidad: imprescindible para solucionar este problema antes de que el producto pueda ser implantado.

Se incluye también la opción N/S (No Sabe) para el caso en que el experto no conozca el criterio. Esta opción no tiene puntaje asignado para que no interfiera en la valoración final. Cada categoría contiene además un ítem para comentarios del evaluador. Esta información cualitativa complementa la información cuantitativa.

En particular, en la categoría “Significatividad psicológica” se pregunta al evaluador acerca de uno de los atributos del OA: su reusabilidad tanto en diferentes cursos o contextos de aprendizaje.

El Apéndice C presenta la Plantilla **GEHOA**.

En cuanto a los indicadores de calidad, se tuvieron en cuenta dos: la valoración individual de cada categoría y la valoración global que se calculó como promedio ponderado, en donde cada peso es la proporción de criterios heurísticos asociados a cada categoría tal se muestra en la Tabla 7.3.

CATEGORÍA	NRO. de SUBHEURISTICAS	PESO
Significatividad Psicológica	10	16%
Significatividad lógica	20	33%
Diseño de interfaz	17	28%
Diseño de navegación	14	23%
Total	61	100,00%

Tabla 7.3. Pesos de cada categoría

En el capítulo siguiente se presentan un caso de estudio de aplicación del Modelo de Proceso **MPBOA**, incluyendo el uso de la Plantilla **GEHOA**.

Aún así, coincidimos con Squires & Preece (1999) en que las heurísticas propuestas se ofrecen como un "primer corte" para el desarrollo de una herramienta de evaluación predictiva que tenga en cuenta tanto la usabilidad y las cuestiones de aprendizaje y que las integre. Existe claramente una necesidad de un estudio empírico para probar la eficacia de estas heurísticas, un estudio que refleje en un contexto educativo el análisis factorial de

Nielsen (1994b), que será la próxima etapa de nuestro trabajo.

7.3. Instrumento para la evaluación por estudiantes

7.3.1. Trabajos relacionados

a) Protocolo de evaluación heurística de Reeves

En este trabajo, Reeves et. al (2002) modificaron y adaptaron el protocolo de Nielsen (1994a) para la evaluación de cursos en línea (e-learning) .

Las modificaciones principales introducidas consistieron en la expansión de las diez heurísticas de Nielsen (1994a) desarrolladas para las aplicaciones o sistemas de software en general a quince heurísticas enfocadas en los cursos en línea.

Proponen un "Protocolo para la Evaluación Heurística de E-Learning" que explica cómo todo el proceso debe llevarse a cabo. Este proceso consta de ocho pasos y los problemas de usabilidad se supone que deben ser evaluadas a lo largo de dos escalas (la severidad y persistencia).

Con el fin de realizar y poner en práctica el paradigma de diseño centrado en el alumno, reflexionar sobre las necesidades de los alumnos y comprender sus actitudes, Reeves et. al. (2002), consideran que las técnicas de evaluación del e-learning deben incluir las percepciones de los alumnos también.

Sugiere que tal método incorpore las características básicas siguientes:

1. Basarse en la integración de la usabilidad y el diseño instruccional.
2. Tener en cuenta las percepciones de los alumnos.
3. Debe ser corto, fácil de implementar.

Las quince heurísticas propuestas surgen de la combinación de "Diseño Instruccional y usabilidad" (Reeves et. al., 2002):

1. Visibilidad del estado del sistema: el sistema de e-learning mantiene informado al estudiante sobre lo que está sucediendo, a través de respuestas en un plazo de tiempo razonable.
2. Adecuación entre el sistema y el mundo real: la interfaz emplea palabras, frases y conceptos familiares para el estudiante o apropiado para el contenido, en lugar de términos orientados al sistema. Siempre que sea posible, utiliza convenciones del mundo real que hacen que la información aparezca en un orden natural y lógico.

3. Ayuda a los usuarios a reconocer, diagnosticar y recuperarse de los errores: el sistema de e-learning permite al estudiante a recuperarse de los errores de entrada y proporciona una clara "salida" sin tener que pasar por un diálogo ampliado.
4. Consistencia y estándares: cuando sea apropiado para el contenido y el público objetivo, el sistema de e-learning en general se adhiere a convenciones de software y es coherente en el uso de diferentes palabras, situaciones o acciones.
5. Prevención de errores: el sistema de e-learning está diseñado para evitar los problemas comunes que se produzcan en primer lugar.
6. Apoyo a la navegación: el sistema de e-learning hace que los objetos, acciones y opciones visibles no tenga que recordar la información cuando se navega en una parte del sistema a otro. Las instrucciones de uso del sistema son siempre visibles o fácilmente recuperables.
7. Estética: la pantalla no contienen información que es irrelevante. Los “sonidos o llamadas” no se añadieron a título gratuito.
8. Ayuda y documentación: el sistema de e-learning ofrece ayuda y documentación que sea fácilmente accesible cuando sea necesario. La ayuda que da una serie de medidas concretas para que el usuario continúe. Toda la documentación está escrita de forma clara y concisa.
9. Interactividad: el sistema de e-learning ofrece contenidos relacionados con las interacciones y tareas que apoyan el aprendizaje significativo.
10. Diseño de mensajes: el sistema de e-learning presenta la información de acuerdo con los principios adecuados de procesamiento de la información.
11. Diseño de aprendizaje: las interacciones en el sistema de e-learning han sido diseñados de acuerdo con sólidos principios de teoría del aprendizaje.
12. Integración de Medios de Comunicación: la inclusión de los medios de comunicación en el sistema de e-learning sirve con fines pedagógicos y/o motivación.
13. Evaluación: el sistema de e-learning ofrece oportunidades de evaluación que están alineados con los objetivos del programa y el contenido.
14. Recursos: El sistema de e-learning proporciona acceso a todos los recursos necesarios para apoyar el aprendizaje efectivo.
15. Comentarios: El sistema de e-learning proporciona información que es contextual y relevante para el problema o la tarea en la que el estudiante se dedica.

b) Cuestionario PMLQ

Nokelainen(2006) ha realizado una revisión de los principales trabajos que han tratado la usabilidad de recursos educativos, destacando los trabajos de Reeves (1994), Squires & Preece (1996, 1999), Quinn (1996), Albion (1999) y Horila, Nokelainen, Syvänen y Överlund (2002). En estos trabajos, Nokelainen (2006) identifica cinco criterios comunes, que atribuye a aspectos técnicos de los recursos: “control del alumno”, “posibilidad de aprendizaje cooperativo/ colaborativo”, “explicitación de los objetivos”, “autenticidad del material de aprendizaje”, “ayuda o soporte”.

El modelo desarrollado por Nokelainen, que posteriormente se traslada a un cuestionario, el PMLQ o “Pedagogically Meaningful Learning Questionnaire”, aborda el problema desde el punto de vista de la experiencia subjetiva de los estudiantes en su interacción directa con los contenidos educativos, y está compuesto por diez criterios, con sus respectivas sub-dimensiones, que los definen y delimitan (Tabla 7.4):

Criterios	Subdimensiones
1. Learner control	Minimum memory load; meaningful encoding; Responsibility for learning; User control; Elaboration.
2. Learner activity	Reflective thinking; Problem-based learning; Use of primary data sources; Immersion; Ownership; Primary data source (for PBL) (for teacher); Facilitative teacher (for teacher); Didactic teacher (for teacher); Individual/distance learning (for teacher).
3. Cooperative/Collaborative learning	Support for conversation and dialogue; Group work; Asynchronous social navigation; Synchronous social navigation; Asynchronous social navigation monitoring (for teacher); Synchronous social navigation monitoring (for teacher); Tertiary courseware.
4. Goal orientation	Explicit goals; Usefulness of goals; Focus on results; Focused goals; Monitor one's own studies (pedagogic feedback); Set one's own goals (for teacher)
5. Applicability	Authentic material; Perceived usefulness; Learning by doing; Adequate material for the learners needs (human development); Pretesting and diagnostics; Prompting; Fading; Scaffolding; Meaningful encoding.
6. Added value	Overall added value for learning; Effectiveness for learning; Added value of pictures; Added value of sounds; Added value of animations.
7. Motivation	Intrinsic goal orientation; Extrinsic goal orientation; Meaningfulness of studies; Immersion.
8. Valuation of previous knowledge	Prerequisites; Elaboration; Examples.
9. Flexibility	Pretesting and diagnostics; Task decomposition; Repetitive tasks.
10. Feedback	Encouraging feedback; Accurate feedback; Errorless learning.

Tabla 7.4: Criterios de evaluación de usabilidad en recursos educativos de Nokelainen (2006)

c) System usability scale (SUS)

El cuestionario SUS fue desarrollado en 1986 como parte de la introducción de la ingeniería de usabilidad a los sistemas de oficina de Digital Equipment Co. Ltd. Se basa en la norma ISO 9241-11 (1998).

Su propósito era proporcionar un test fácil de completar (número mínimo de cuestiones), fácil de puntuar y que permitiera establecer comparaciones cruzadas entre productos, es decir, que fuera posible comparar unas soluciones con otras y optar por la mejor (Brooke, 1996). SUS es de dominio público y de uso libre, siempre que se cite adecuadamente la fuente: © Digital Equipment Corporation.

Contiene una escala Likert y su resultado final es un valor único que representa una medida compuesta de la usabilidad del sistema global sometido a estudio (Figura 7.2).

	Completamente de acuerdo				Completamente en desacuerdo
1. Creo que me gustará usar con frecuencia este sistema	1	2	3	4	5
2. Encontré el sistema Innecesariamente complejo	1	2	3	4	5
3. Pensé que era fácil utilizar el sistema	1	2	3	4	5
4. Creo que necesitaría del apoyo de un experto para recorrer el sistema	1	2	3	4	5
5. Encontré las diversas posibilidades del sistema bastante bien Integradas	1	2	3	4	5
6. Pensé que había demasiada Inconsistencia en el sistema	1	2	3	4	5
7. Imagino que la mayoría de las personas aprenderían muy rápidamente a utilizar el sistema.	1	2	3	4	5
8. Encontré el sistema muy Incómodo de usar	1	2	3	4	5
9. Me sentí muy confiado en el manejo del sistema	1	2	3	4	5
10. Necesito aprender muchas cosas antes de manejarme en el sistema	1	2	3	4	5

Figura 7.2. Cuestionario SUS

SUS ha adquirido mucha popularidad, en un estudio realizado por Bangor, Kortum & Miller (2008), se informa que hasta la fecha de la publicación, este cuestionario fue implementado por mas de 2300 usuarios en 200 estudios de usabilidad , lo que proporciona un punto de referencia para ser utilizado por otros profesionales de usabilidad.

En contraste con otros cuestionarios de usabilidad, SUS es muy corto y por lo tanto recomendable para llevar a cabo en el combinación con otras medidas de evaluación de la usabilidad (Blecken, Bruggemann & Marx,2010).

d) QUIS (Questionnaire for User Interface Satisfaction):

El cuestionario fue desarrollado a finales de los años 80 por el laboratorio de IPO de la Universidad de Maryland, EE.UU (Chin et al. 1988, Harper y Norman 1993) y fue basado

en la escala de Shneiderman (1986).

Se han desarrollado varias versiones con diversas subescalas, preguntas, y niveles de confiabilidad.

QUIS está centrado en el usuario para evaluar su percepción de la usabilidad de la interfaz.

En la versión 7 (QUIS, 2001). Los aspectos que evalúa son los siguientes:

1. Reacción global al sistema, para ello emplea preguntas como 'terrible' vs. 'frustrado', 'lánguido' vs. 'estimulante', etc. No trata características específicas de la interfaz utilizada ni la interacción.
2. Factores de pantalla: fuentes y negrita, la lógica de la interfaz, la secuencia de pantallas, el control del usuario, la recuperación después del error.
3. Terminología y sistema de retroalimentación, para medir la claridad de los mensajes con las preguntas relacionadas.
4. Factores de aprendizaje, referido a la experiencia de aprendizaje y también a las características específicas del sistema tales como realimentación, lógica de secuencias y posibilidad de intuir.
5. Capacidades del sistema, referido a las experiencias del usuario con respecto a la velocidad del funcionamiento, confiabilidad, capacidad de gestión de error y la flexibilidad del sistema.

Además contiene cuatro secciones opcionales para evaluar componentes específicos del sistema, tales como, manuales técnicos, ayuda y tutoriales on-line o software de instalación.

Cada ítem es valorado en una escala de 1 a 9 con adjetivos fijados a la derecha (positivos) y a la izquierda (negativos) y la opción “no aplicable”. Adicionalmente incluye espacios para comentarios por cada uno de los factores específicos de la interfaz (Figura 7.3).

El propósito del cuestionario QUIS es servir como guía en el diseño o rediseño de sistemas, ofrecer una herramienta para evaluar áreas potenciales de mejoramiento de sistemas, proveer a los investigadores con un instrumento válido para conducir evaluaciones comparativas.

PARTE 3: Impresión General del Usuario

Por favor circule los números que mejor reflejen sus impresiones sobre el uso de este sistema.
NA significa "No aplica".

3.1	Impresión general del sistema:	muy buena 1 2 3 4 5 6 7 8 9	muy mala 1 2 3 4 5 6 7 8 9	NA
3.2		frustrante 1 2 3 4 5 6 7 8 9	agradable 1 2 3 4 5 6 7 8 9	NA
3.3		aburrido 1 2 3 4 5 6 7 8 9	estimulante 1 2 3 4 5 6 7 8 9	NA
3.4		difícil 1 2 3 4 5 6 7 8 9	fácil 1 2 3 4 5 6 7 8 9	NA
3.5		capacidad inadecuada 1 2 3 4 5 6 7 8 9	capacidad adecuada 1 2 3 4 5 6 7 8 9	NA
3.6		rígido 1 2 3 4 5 6 7 8 9	flexible 1 2 3 4 5 6 7 8 9	NA

Figura 7.3. Parte 3: Impresión General del Usuario. Tomado de QUIS (2001)

e) SUMI

SUMI (Software Usability Measuring Inventory) fue desarrollado por la Universidad College Cork (Kirakowski, Porteous y Corbett, 1992) y es utilizado para medir la satisfacción y valorar la percepción del usuario de la calidad del software.

Es un cuestionario de 50 ítems, desarrollado, validado y estandarizado. Este cuestionario está referenciado en estándares de calidad ISO como el ISO 9241-10 (2010) (principios de dialogo) y en el ISO 9126-2 (2003) (métricas externas y características de la calidad del software). Se encuentra disponible, previo pago, en varios lenguajes.

Los resultados que arroja SUMI están basados en una gran base de datos que consta de perfiles de usabilidad de más de 2000 diferentes clases de aplicaciones tales como: procesadores de textos, hojas de cálculo, paquetes CAD, programas de comunicaciones, etc. A partir de las respuestas obtenidas mediante esta herramienta y conceptos estadísticos puede calcularse el valor de la usabilidad del producto (Veenendaal, 1998).

SUMI proporciona una valoración global de la usabilidad y la de cinco sub-escalas:

- Eficiencia: grado al cual el usuario puede lograr las metas de su interacción con el producto en una manera directa y oportuna.
- Afectividad: en que medida el producto captura las respuestas emocionales

del usuario.

- Utilidad: en que grado el producto ofrece asistencia al usuario.
- Control: grado al cual el usuario siente que él, y no el producto, es el que marca el paso.
- Aprendizaje: facilidad con la cual el usuario consigue comenzar a utilizar el producto así como aprender nuevas características del mismo.

Estas sub-escalas representan las dimensiones sobre las que los usuarios finales estructuran sus juicios cuando valoran la usabilidad del software.

f) Comparación de cuestionarios

Alva (2005) comparó 9 herramientas de evaluación de usabilidad : WAMMI (Kirakowski, Claridge & Whitehand, 1998), ISOMETRICS (Gedica, Hamborg & Duentsch,1999) , SUMI (Bevan, 1995) , MUMMS (2003) , PROKUS (Zülch, G. & Stowasser, 2000) , QUIS (2001) , DRUM (Macleod & Rengger, 1993) , SMEQ (Zijlstra,1993) y TLX(Hart, & Staveland, 1988) con un total de 25 dimensiones de medición de usabilidad, aunque ninguna herramienta considera criterios de evaluación de aspectos educativos.

El objetivo de los cuestionarios DRUM, SMEQ y TLX es informar sobre el desempeño del usuario, no específicamente la satisfacción.

Según un estudio realizado por Restrepo Jaramillo (2010), los cuestionarios WAMMI, SUS, SUMI, ISOMETRICS y QUIS miden parcialmente algunas de las dimensiones halladas por estudios de satisfacción para el contexto Web, como la experiencia de interacción, los resultados cognitivos, el placer, la privacidad y la seguridad, la apariencia visual, los aspectos técnicos y la navegación.

Por otra parte, Alva (2005) concluyó que las herramientas SUMI y QUIS son las que consideran un mayor número de criterios de evaluación de usabilidad dentro de las 9 analizadas.

A pesar de su sencillez, SUS, alcanza una fiabilidad que varía entre 0.85 y 0.91, según distintos estudios, lo que lo sitúa entre los más fiables en comparación con otros instrumentos similares para distintos tamaños de muestras (Tullis & Stetson, 2004).

7.3.2. Instrumento CUSEOA (Cuestionario de satisfacción de estudiantes de un OA)

Según Williams (2000), los estudiantes son algunos de los usuarios más importantes de OAs y, por supuesto, varían según sus necesidades y valoraciones que pueden o no coincidir con la de sus profesores. Los estudiantes tienen que evaluar cada oportunidad de aprendizaje y escoger aprender de ella o no porque ellos son la clave para la evaluación de los OAs.

Por otra parte, Nielsen (1993), definió los cinco atributos básicos de la usabilidad:

- Facilidad de aprendizaje: rapidez con que un usuario aprende a utilizar un sistema con el que no ha tenido contacto anteriormente (que lo haga de forma sencilla, rápida e intuitiva).
- Eficiencia: que el usuario pueda alcanzar un alto nivel de productividad al saber usar un sistema.
- Retención en el tiempo: que el usuario recuerde fácilmente cómo se utilizaba el sistema si deja de emplearlo por un tiempo.
- Tasas de error de los usuarios: se refiere a la cantidad y gravedad de errores que comete el usuario. Al cometer un fallo, el sistema debe informar al usuario y ayudarlo a solventarlo.
- Satisfacción subjetiva: hace referencia a si los usuarios se sienten cómodos y satisfechos utilizando el sistema, es decir, si les gusta o no (impresión subjetiva).

Blecken, Bruggemann & Marx (2010) afirman que si los primeros cuatro requisitos no se cumplen, los usuarios no podrán disfrutar de trabajar con el sistema. La satisfacción está estrechamente relacionada con los cuatro primeros atributos de la usabilidad.

Se pretende así, conocer si el OA ha sido adecuado como unidad de enseñanza, es decir, valorar su calidad desde el punto de vista de los usuarios finales: los estudiantes, con un cuestionario de satisfacción que integre las dimensiones pedagógica y técnica.

Por otra parte, esta evaluación puede definir muchas cuestiones acerca de la valoración de un contenido, por ejemplo si los estudiantes manifiestan dificultades para comprenderlo o no ha facilitado su aprendizaje a pesar de ser calificado como de alta calidad por expertos en una evaluación heurística.

Para la construcción de **CUSEOA**, además de lo expresado en los párrafos anteriores se tuvieron en cuenta:

- Las recomendaciones de Reeves et. al. (2002) sobre las características básicas de un cuestionario de evaluación de usabilidad de recursos educativos en línea.
- El Cuestionario PMLQ de Nokelainen (2006) que aborda el problema desde el punto de vista de la experiencia subjetiva de los estudiantes en su interacción directa con los contenidos educativos y se basa en la revisión de numerosos trabajos de investigación en el campo.
- El cuestionario SUS por su fiabilidad. Además, a pesar de ser corto puede combinarse con otros métodos como propone el Modelo de proceso **MPOBA** presentado en esta tesis.
- El cuestionario QUIS por los aspectos que evalúa, la adaptación de los criterios heurísticos al contexto de un recurso educativo y por seguir los principios de DCU.

La estructura de **CUSEOA** es la siguiente:

1. Reacción global al OA
2. Dimensión pedagógica
3. Dimensión técnica

También contiene un ítem de comentarios para que el estudiante exprese libremente su opinión sobre el OA. Esta información complementará a la obtenida en las otras tres partes de **CUSEOA**

1. Reacción global al OA:

Tal lo expresado en los párrafos anteriores, el estudiante valorará el OA, no sólo por la funcionalidad, usabilidad, los contenidos, sino también por las emociones o los sentimientos que le provoca.

El diferencial semántico (DS) es uno de los métodos más utilizados en el llamado “diseño emocional”. La misma aporta información sobre las emociones que el objeto genera, obteniendo el valor connotativo y captando el significado afectivo que el usuario tiene de él (Mondragón, Vergara y Company, 2005).

El procedimiento de aplicación es el siguiente: ante un objeto o imagen se solicita al sujeto emitir un juicio subjetivo. El juicio debe darse de acuerdo a una escala con dos descriptores o adjetivos opuestos situados en los extremos.

En los últimos años se han desarrollado metodologías basadas en DCU que utilizan el diferencial semántico para medir ese valor emotivo del producto, permitiendo producir nuevos productos basados en los deseos y demandas del consumidor. La Ingeniería Kansei (IK) (Nagamachi, 1995) es una de las metodologías precursoras. Se trata de una herramienta de ingeniería que permite captar las necesidades emocionales de los usuarios y establecer modelos de predicción matemáticos para relacionar las características de los productos con esas necesidades emocionales.

Es decir, desde el punto de vista técnico un buen producto debería satisfacer todas las expectativas del consumidor, pero especialmente la de provocar una respuesta emocional positiva.

Desde el punto de vista pedagógico, como ya se indicó antes, el aprendizaje no puede darse si el alumno no quiere aprender. Este es un componente que se refiere a la disposición emocional y actitudinal, que puede abordarse desde la motivación (Ausubel, 1983).

Esta componente del instrumento **CUSEOA**, aborda estos aspectos en dos partes.

- a) Se le presenta al estudiante una serie de adjetivos bipolares y se solicita que lo clasifique en una escala 7 puntos (Ver apéndice D).
- b) Valoración de recomendación del OA a otros estudiantes a través una proposición o afirmación que debe calificar utilizando una escala desde 1 correspondiente a “totalmente en desacuerdo” hasta 5 para “totalmente de acuerdo”).

2. Dimensión pedagógica:

Comprende aspectos relativos a los objetivos, contenidos teóricos, actividades y realimentación. Su propósito es valorar la significatividad lógica y psicológica en los OA desde la perspectiva del estudiante (Tabla 7.5).

Consta de 6 proposiciones o afirmaciones que debe calificar utilizando una escala desde 1 correspondiente a “totalmente en desacuerdo” hasta 5 para “totalmente de acuerdo”.

3. Dimensión técnica (Usabilidad Web):

Comprende aspectos relativos al diseño de la interfaz y la estructura y navegación (Tabla 7.6). El estudiante debe valorar la Funcionalidad y Usabilidad luego de la interacción con el OA. Consta de 7 proposiciones o afirmaciones que debe calificar utilizando una escala desde 1 correspondiente a “totalmente en desacuerdo” hasta 5 para “totalmente de acuerdo”.

	CATEGORÍA	CRITERIOS HEURÍSTICOS	SUB-HEURISTICAS
USABILIDAD PEDAGÓGICA	SIGNIFICATIVIDAD PSICOLÓGICA	Motivación	Recomendaría este recurso a otra persona.
		Conocimientos previos	El nivel de dificultad de los contenidos fue elevado para mis conocimientos previos.
		Innovación y autonomía	El sistema informa sobre mi progreso.
	SIGNIFICATIVIDAD LÓGICA	Objetivos	Los objetivos indican lo que se espera que sea aprendido.
		Contenidos	El material teórico me ayudó a comprender los conceptos.
		Actividades	Las actividades han sido claras y significativas para mi aprendizaje
USABILIDAD DE SITIOS WEB	DISEÑO DE INTERFAZ	Lenguaje de los usuarios	Las imágenes empleadas me ayudaron a aclarar los contenidos. Los videos y las animaciones me ayudaron a aclarar los contenidos.
		Diseño estético y minimalista	En general, los colores y el diseño de todo el recurso son adecuados.
	ESTRUCTURA Y NAVEGACIÓN	Visibilidad del estado del sistema	Me encontré perdido cuando recorría el recurso, no sabía dónde me encontraba
		Reconocimiento más que memoria	El texto es conciso y preciso. Los títulos son inadecuados, no se sabe cual es la acción que se debe realizar.
		Navegación visible	La información está mal organizada.

Tabla 7.5. Categorías y criterios heurísticos

En cuanto a los indicadores se tuvieron en cuenta:

- La valoración individual de las dimensiones pedagógica y técnica.
- La valoración global de las dimensiones pedagógica y técnica que se calcula como promedio ponderado, en donde cada peso es la proporción de criterios heurísticos asociados a cada dimensión tal como se muestra en la Tabla 7.7.

DIMENSION	NRO. DE SUB-HEURISTICAS	PESO
Usabilidad Pedagógica	6	46%
Usabilidad Web	7	54%
Total	13	100%

Tabla 7.6. Pesos de cada categoría

- La reacción global a partir del análisis de los resultados de la escala de diferencial semántico.

Como ya se mencionó en la subsección 6.2.1.5. del Capítulo 6, estos resultados se combinan con el análisis de la grabación de uso resultante de la interacción de cada uno de los estudiantes que respondió el cuestionario.

CAPITULO 8. TRABAJO EXPERIMENTAL: OBJETO DE APRENDIZAJE PARA UN CURSO DE PROGRAMACIÓN INICIAL O ALGORITMIA

En este capítulo se presentan los resultados experimentales correspondientes a la aplicación del Modelo de Proceso MPOBA a un OA para un Curso de Programación inicial o Algoritmia. Se detallan y documentan tanto el proceso de elaboración, como los resultados obtenidos de la aplicación del Modelo de Proceso. Se realiza el análisis de los resultados obtenidos y las conclusiones que se derivan de los mismos.

8.1. Introducción

La enseñanza de la programación es un tema con múltiples facetas, que sigue siendo una línea abierta de investigación en buena parte del mundo. Se han propuesto muchos enfoques y herramientas distintas en los últimos años como los de: Cassola (2004), Kölling (1999), Zhu & Zhou (2003) entre otros.

Linn & Dalbey(1989), señala que aprender a programar implica:

- Aprender la sintaxis y la semántica características del lenguaje de programación.
- Aprender a combinar el conocimiento aprendido con el ya conocido.
- Desarrollar las habilidades generales de resolución de problemas.

En palabras de Wirth (1971): “La programación es el arte y la técnica de construir y formular algoritmos de una forma sistemática”. Moursund (2008) afirma que la enseñanza de la programación capacita al estudiante para solucionar problemas de todo tipo y de manera organizada. Además elabora estructuras mentales para la solución de problemas, sostiene López (2008).

Sin embargo, su aprendizaje no es sencillo. Según Hernández (2006), las principales dificultades en el aprendizaje de la programación, son:

- Exigencia de un elevado nivel de abstracción.

- Necesidad de un buen nivel de conocimiento y práctica de técnicas de resolución de problemas.
- La existencia de alumnos con niveles de conocimiento diferentes, lo cual se traduce en ritmos de aprendizaje muy variados y tal vez descompensados.
- La imposibilidad de atención individualizada a cada alumno, debido a lo numeroso de los grupos.
- La imposibilidad de asumir un único método para la resolución de los problemas.

Otras dificultades de los estudiantes novatos en la programación, también son presentadas por otros autores:

- Unos pocos tipos de errores representan la mayoría de los errores (Spohrer & Soloway, 1986).
- Abstractar un problema de su descripción (McCracken et al., 2001).
- Localización, lectura y comprensión de piezas de código (Lister et. al., 2004).

Las evidencias que en los cursos de programación se presentan altas tasas de reprobación, poca motivación (Jenkins, 1998) y alto porcentaje de deserción, revelan la necesidad de enfrentar el problema de la enseñanza y el aprendizaje de la programación desde aproximaciones alternativas. Las soluciones simples (cambiar de libro, utilizar otro lenguaje de programación, cambiar el orden de los temas, etc.) ya han sido intentadas en infinidad de variantes sin que se logren mejoras efectivas (CUPI, 2009).

Llamosa et. al. (2009) estudiaron las dificultades que se presentan en el proceso de enseñanza y de aprendizaje de la programación orientada a objetos e identificó que el estilo de enseñanza que se establece desconoce el estilo de aprendizaje del estudiante, no se hace un diagnóstico para determinar los conocimientos previos del estudiante, no se utiliza material educativo apropiado para dar soporte a la enseñanza; además, hace falta una cuidadosa planificación, organización, evaluación y mejoramiento del curso.

En este sentido, el aprendizaje de la programación no depende sólo de la disciplina, disposición y perseverancia de cada estudiante, sino también de las competencias pedagógicas del profesor y de las condiciones que el medio le ofrezca, analizando constantemente cómo se enseña y presentando propuestas que integren los conocimientos del campo de la programación con los de la pedagogía y la didáctica, acordes con las tecnologías y las tendencias actuales.

Si nos remontamos históricamente a los primeros pasos en la forma de programar,

observamos que los primeros programas eran el resultado de un proceso creativo más que de un razonamiento riguroso sobre un diseño de solución; dada la precariedad de conocimientos sobre el tema, lo importante era que el programa se ejecutara y desarrollara las operaciones para las que fue construido. Dijkstra & Feijen (1988), criticaron esta forma artesanal de programar e insistieron en que la corrección del programa debería demostrarse mediante un proceso formal de razonamiento a partir de su especificación.

Van Amstel (1984) consideró tres enfoques para la enseñanza de la programación: formal, semiformal y no formal:

- Formal: propende por una aplicación de fundamentos matemáticos para diseñar algoritmos en los que se pueda demostrar formalmente su corrección (apropiado para matemáticos y para estudiantes de Ciencias de la Computación).
- Semiformal: busca la corrección con base en una derivación formal, pero admite cierto margen de informalidad pues considera que no es posible demostrar matemáticamente la corrección en la solución de todos los problemas (apropiado para ingenieros de sistemas e informáticos).
- No formal : no exige el razonamiento matemático, de manera que los programas pueden construirse mediante prueba y error, intentando una y otra vez hasta lograr una solución aceptable (para niveles de formación técnica y en carreras profesionales donde la programación es un área complementaria).

Este caso de estudio que presentamos corresponde a un OA para un curso de programación inicial de la Facultad de Ingeniería de la Universidad Nacional de Mar del Plata. Las carreras que se dictan son “No -informáticas”.

El primer paso para abordar el problema de la enseñanza de la programación en la Ingeniería es analizar qué debe aportar la Informática a la formación de un ingeniero. Lowe (2000) realizó un estudio sobre la contribución de la Informática al perfil de un ingeniero. Dicho perfil es descrito a través de un conjunto de atributos, tales como conocimiento técnico, madurez, capacidad de planteamiento y resolución de problemas entre otras. Estos atributos Lowe (2000) los agrupa en tres campos: formación profesional, desarrollo personal y desarrollo académico. La contribución de la Informática para este autor se realiza a tres niveles:

1. Desarrollo de competencias en áreas específicas tales como: aprender un determinado lenguaje de programación o herramientas de software. También contribuirá al desarrollo de habilidades necesarias para apoyar su aprendizaje en

otras áreas (como el uso de herramientas de modelado en el diseño de sistemas, hojas de cálculo, etc.).

2. Desarrollo de competencias relacionadas con la comprensión de clases generales de herramientas informáticas, sin atarse a productos específicos.
3. Conocer las aplicaciones informáticas de su campo profesional.

Por otra parte, el objetivo de un curso de programación no es únicamente que el estudiante aprenda a escribir un programa. En forma transversal se puede contribuir a la generación de otras competencias: entender un problema (abstraer, modelar, analizar), plantear soluciones efectivas (reflexionar sobre una abstracción, definir estrategias, seguir un proceso, aplicar una metodología, descomponer en subproblemas), a manejar lenguajes para expresar una solución (codificar, entender y respetar una sintaxis), a utilizar herramientas que entiendan esos lenguajes (programar, compilar, ejecutar, depurar), a probar que la solución sea válida (entender el concepto de corrección y de prueba), a justificar las decisiones tomadas (medir, argumentar), etc. (CUPI, 2009) . Estas son competencias genéricas con las que debe contar cualquier profesional en Ingeniería tal lo señala el Consejo Federal de Decanos de Ingeniería (CONFEDI) en su documento “Competencias Genéricas – Desarrollo de Competencias en la Enseñanza de la Ingeniería (CONFEDI, 2006), el Proyecto Tuning (2007) y el Accreditation Board of Engineering and Technology, ABET (2011).

La educación en Ingeniería es un campo con fuertes tradiciones, pero diversos ejemplos de modificación de enfoques tradicionales a otros que utilizan estrategias pedagógicas que involucran a los estudiantes como agentes de generación de su propio conocimiento, han evidenciado la necesidad de cambio y la fuente de posibles soluciones frente a dificultades en la enseñanza de ingeniería (De Graff & Kolmos, 2007).

8.2. Aplicación del Modelo de proceso MPOBA

A lo largo de esta sección detallaremos y documentaremos tanto el proceso de elaboración como los resultados obtenidos de la aplicación del Modelo de Proceso MPOBA para el desarrollo de un OA para un curso de Programación inicial o Algoritmia.

8.2.1. Planificación del OA

a) Caracterización de los usuarios finales (profesores y estudiantes)

Si se pretende caracterizar al docente en los espacios universitarios generalmente se lo asocia con la imagen del profesor como transmisor del conocimiento, preocupado por brindar información, esclarecer y explicar los contenidos presentados y estableciendo estrategias que le permitan evaluar desde el control que las competencias alcanzadas por los estudiantes sean las adecuadas. En este contexto el estudiante no tiene más que un rol pasivo, el proceso de aprendizaje se desarrolla en forma individual y los docentes tienen la certeza que se realiza el proceso de enseñanza tal como estaba estipulada y queda la inquietud sobre lo que sucede con el aprendizaje.

Con la experiencia de muchos años en el dictado de Cursos de Programación inicial, coincidimos con: el informe CUPI (2009), Kurland et.al. (1986), McCracken et.al. (2001), Kay et al. (2000), Fernández & Millán (2003), Robins, Rountree & Rountree (2003) en cuanto a la caracterización de los profesores y estudiantes señaladas por los autores. Las conclusiones arribadas son las siguientes:

- Los cursos típicos de programación están muy guiados por el eje asociado con el lenguaje, y los demás sólo aparecen puntualmente en algunas partes del curso. Esto hace que el curso (al igual que la mayoría de libros para esos cursos) esté basado en un recorrido de las estructuras sintácticas del lenguaje. Todos los demás aspectos que hacen parte de la labor de programar, se ven de manera lateral, o sencillamente se ignoran.
- Estructurar un curso en base a un lenguaje particular hace que en muchos casos el estudiante no adquiere una visión real de lo que es programar, dándole más importancia a los elementos del lenguaje que al proceso de construir un programa. En las clases prácticas, por ejemplo, se responden muchas más preguntas ligadas al uso de una librería en particular, que a la manera de especificar un problema. El estudiante menosprecia algunos de los elementos que el profesor intenta introducir de manera lateral al curso, y eso dificulta el proceso de enseñanza.
- Al final del cursado, el estudiante tiene la impresión de que aprendió un lenguaje, pero no es consciente ni valora realmente las competencias (como las mencionadas en la introducción) que generó. Es por ello que, en algunos casos, si el estudiante sabe que no va a utilizar en su vida profesional el lenguaje de programación que vio en el curso, siente que está perdiendo el tiempo.
- En las clases tradicionales, el profesor explica los conceptos mediante el uso del pizarrón y/o diapositivas para complementar la explicación. Estos recursos son válidos

para una gran cantidad de temas teóricos pero son insuficientes en otros más prácticos como programación. Muchos de los estudiantes no son capaces de asimilar toda la teoría y tienden a concentrarse en los aspectos superficiales, ya que no hay tiempo para profundizar en las bases teóricas. Generalmente se complementa con clases prácticas, aunque por lo general son insuficientes para proporcionar al estudiante un conocimiento profundo de los principios de programación (a veces debido a que el ritmo enseñanza es superior al ritmo de aprendizaje y el estudiante no es capaz de seguir al profesor en las explicaciones).

- Un estudiante sin experiencia en la programación, tiene dificultad en comprender cómo funciona un programa. Un programa estructurado es una amalgama de declaraciones (e.g. variables, indicadores, tipos de compuestos, etc.), instrucciones (e.g. los bucles y sentencias condicionales) y las definiciones de función que no siguen una secuencia concreta de codificación. Estas declaraciones son difíciles de entender, por la naturaleza diferente de los tipos de datos empleados en el programa; las instrucciones no son fáciles de seguir (especialmente bucles) debido a sus diferentes comportamientos, así como a la evaluación de las expresiones condicionales sobre el cual su comportamientos dependen, y las funciones se confunden, porque se rompe el flujo de ejecución secuencial.
- Los novatos pasan poco tiempo en la planificación y las pruebas del código. Cuando es necesario, tratan de corregir sus programas con pequeñas correcciones locales en lugar de ir más fondo, es decir en la reformulación de los programas. Se enfocan en la programación "línea por línea" en lugar de un nivel más amplio como las estructuras que componen el programa.
- Los estudiantes tienen dificultades con frecuencia en la comprensión de todas las cuestiones relativas a la ejecución de un programa. Se necesita tiempo para aprender la relación entre el código de un programa y el mecanismo que describe.
- En un curso típico, los conceptos se presentan siguiendo un orden "de abajo hacia arriba". Primero los elementos básicos (tipos simples, operadores, expresiones, etc.) y luego, se van introduciendo las estructuras de control, la construcción de funciones, el paso de parámetros, los vectores, las matrices, registros, etc. Sólo al final del curso el estudiante es capaz de construir un programa completo. Esta metodología en general provoca que:
 1. El estudiante no tenga una visión clara de la razón por la cual se introduce un concepto, porque no le ve una necesidad real, dificultando su proceso de aprendizaje.

2. Muchos de los temas que se presentan resultan artificiales debido al tamaño de los problemas sobre los cuales se puede trabajar. Si construir la especificación formal de un problema toma el doble de lo que se necesita para implementarlo y probarlo, puede llevar al estudiante a concluir que la especificación es una pérdida de tiempo (quieren directamente codificar). El estudiante lo interpreta como caprichos del profesor, que intentará cumplir una vez haya puesto a funcionar el programa.

3. Se necesita cubrir una gran cantidad de temas (un par de meses de estudio) antes de poder hacer algo interesante con contexto, y que pueda dar al estudiante la sensación de que lo que está aprendiendo puede ser útil para su vida profesional, sumado a que los conocimientos previos sólo alcanzan a asignaturas básicas como Física, Matemática y Química. La mayoría de los ejercicios que se abordan corresponden a problemas puntuales sin contexto (e.g. calcular el número de veces que aparece un elemento en un arreglo, o calcular la suma de una serie). Sólo hacia el final del curso es posible construir un programa de mayor envergadura, el cual no se parece en nada a cualquier producto comercial que haya utilizado antes.

b) Plataforma (posibilidades/restricciones)

Los productos incorporados en el Proyecto Integral de la Facultad de Ingeniería de la Universidad Nacional de Mar del Plata (UNMDP) están basados en la filosofía del software libre. Bajo este paradigma y luego de analizar las ofertas disponibles, fue seleccionada la plataforma educativa Moodle.

Esta plataforma fue adaptada mediante el desarrollo de prestaciones pertinentes a las características y necesidades propias del proyecto entre las que se encuentran el editor de símbolos GEnero, que surge a partir de sucesivas mejoras del editor DragMath de licencia GNU (Figura 8.1).

La interfaz de Moodle es estándar, sólo es posible cambiar la apariencia del fondo. Se incluyen botones, iconos, menús y barras de navegación propios de esta plataforma educativa (Figura 8.2).



Figura 8.1. Pantalla de un Curso de la Plataforma de la Facultad de Ingeniería

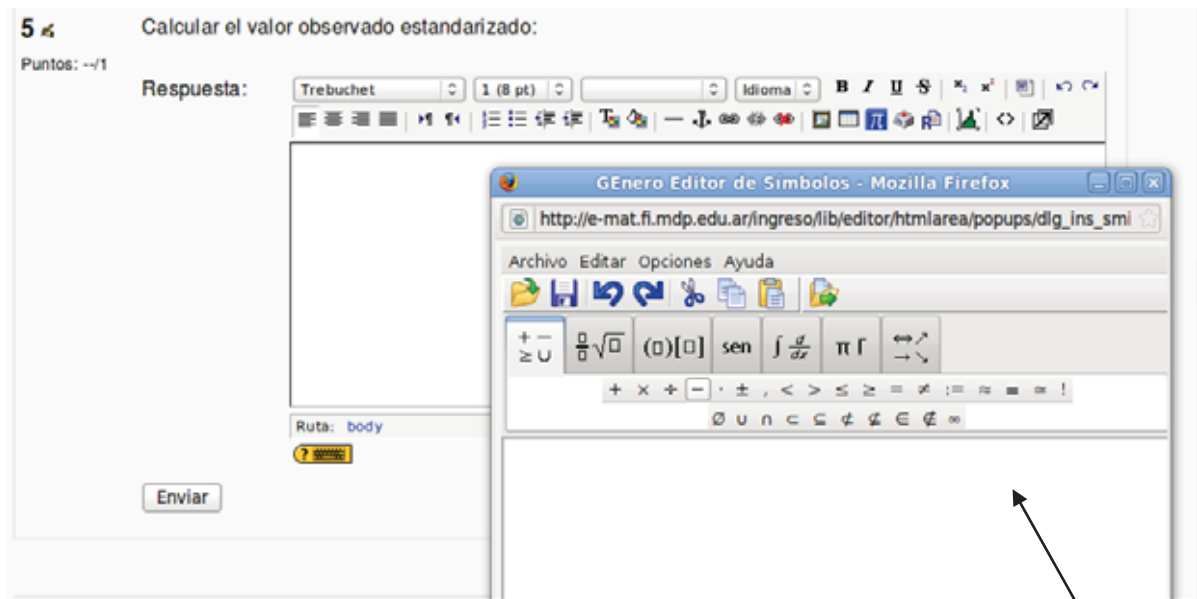


Figura 8.2. Editor de símbolos Genero incorporado a la Plataforma de la Facultad de Ingeniería

c) Determinación de los perfiles de usuario y roles

Para diferenciar entre tipos de usuarios, Moodle proporciona un sistema de roles en función de los permisos y capacidades que deban asignarse a cada usuario, entendiendo por capacidad a la posibilidad de realizar cierta acción en el sistema. Así un estudiante tendrá, entre otras, capacidades para realizar actividades, mientras que un profesor tendrá capacidades para la edición de un curso y un administrador las necesarias para introducir cualquier tipo de modificaciones en el sitio Web (Figura 8.3).

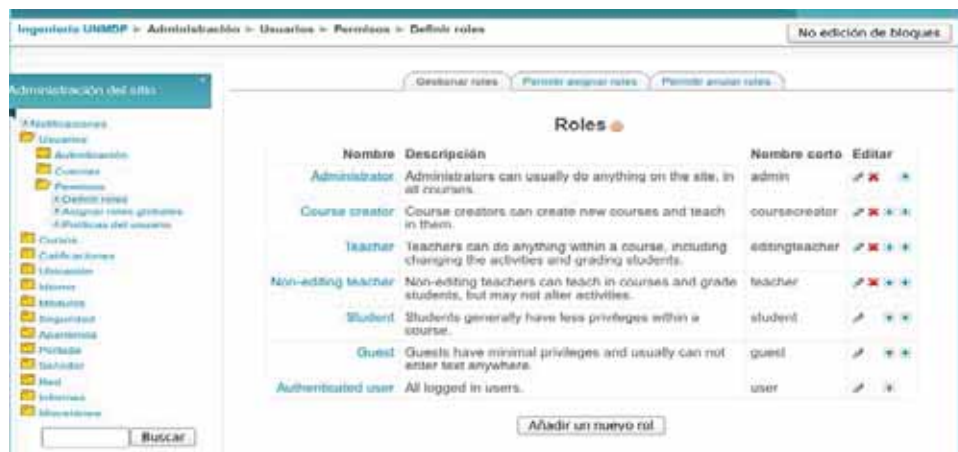


Figura 8.3. Roles de la Plataforma Moodle

8.2.2. Elicitación y Especificación de requerimientos

a) Elicitación de conocimiento para la generación de requerimientos

En la Facultad de Ingeniería el curso de Introducción a la Programación (denominado Computación) es obligatorio para las 8 carreras (Eléctrica, Electromecánica, Electrónica, en Alimentos, en Materiales, Industrial y Mecánica), lo que representa una población semestral de cerca de 300 estudiantes, repartidos en aproximadamente 5 comisiones.

En la Tabla 8.1 se presenta el programa de Computación:

<p>Objetivo General: formar al estudiante de las diversas especialidades de ingeniería en la resolución de problemas usando apropiadamente una computadora.</p> <p>Objetivos específicos: Se desea lograr que el estudiante :</p> <ul style="list-style-type: none"> • Opere las computadoras disponibles usando un Sistema Operativo • Conozca y utilice adecuadamente las características fundamentales del paradigma de programación estructurada • Adquiera las nociones fundamentales del lenguaje Pascal. • Resuelva problemas de mediana complejidad utilizando una computadora.

Contenidos	
Unidad I	Introducción: Componente de un sistema de Computación. Introducción a un sistema operativo. Representación de la Información. Clasificación de los lenguajes de programación. Descripción de un lenguaje de programación: semántica y sintaxis. Algoritmo. El Programa y el programador. Concepto de: datos y programas: Definición, diseño, codificación, compilación o traducción, ejecución y depuración. Paradigmas de programación. Metodologías para la resolución de problemas por medio de una computadora.
Unidad II	Programación estructurada: Análisis Top-down y bottom-up. Refinamientos sucesivos. . Acciones primitivas y no primitivas. Algoritmos estructurados. Estructuras de control. Introducción a Redes e Internet.
Unidad III	Un lenguaje estructurado: Pascal. Estructura de un programa Pascal. Diagramas sintácticos. Tipos primitivos de datos. Declaraciones. Identificadores. Operadores. Expresiones aritméticas, relacionales y lógicas. Sentencias simples y compuestas. Entrada/Salida. Selección y repetición en Pascal. Tipos de bucles. Tipos de datos estructurados estáticos: arreglos, registros. Subprogramas: funciones y procedimientos. Subprogramas estándar y definidas por el programador. Pasaje de parámetros. Introducción a la programación modular. Unidades Pascal. Iteración y recursividad. Archivos Pascal.

Tabla 8.1. Programa de Computación

a-1) Entrevistas

Se entrevistó a los integrantes de la cátedra de Computación con diferentes perfiles (profesores y ayudantes) a fin de recabar información sobre sus experiencias, opiniones y criterios referentes a la enseñanza y aprendizaje de la asignatura. En particular se preguntó cuál sería el tema que seleccionarían para el desarrollo del OA.

Consideraron que se podrían abordar temas claves como: estructuras de control, estructuras de datos, subprogramas, programación modular.

Por otra parte, de estos temas claves, algunos de ellos atraviesan muchos conceptos y métodos fundamentales como es el caso de los subprogramas:

- a) aplicar el método de resolución de problemas sobre todo referido a la etapa de diseño del algoritmo (descomposición, refinamiento)
- b) Dar valor a la reusabilidad en búsqueda de la eficiencia en la escritura el código.
- c) Se avanza sobre la necesidad de mayor abstracción procedural introduciendo conceptos claves como funciones y procedimientos como la forma más adecuada de concebir los problemas en el paradigma estructurado.
- d) Se definen y caracterizan los objetos claves de la comunicación entre los subprogramas: los parámetros, variables locales y globales.

a-2) Indagación Contextual

Se realizó un análisis contextual de las tareas que realizan los usuarios involucrados

(profesores y estudiantes).

Los procesos de intervención pedagógica se realizan en tres modalidades semanalmente:

- 1.- Clase magistral utilizando recursos como pizarrón y medios audiovisuales (2 horas).
- 2.- Trabajo en laboratorio/taller de computadoras para diseñar, editar, compilar, ejecutar y depurar programas (3 horas).
- 3.- Taller-Grupo operativo (resolución de problemas mediante el análisis y diseño de programas) (1 hora).

Los estudiantes concurren a los laboratorios de computación en dónde realizan las siguientes actividades:

- Diseño de algoritmos en papel.
- Codificación del algoritmo en un programa fuente Pascal (en computadora).
- Compilación, depuración y ejecución de programas
- Resolución de problemas de creciente dificultad en computadora.

Estas actividades se llevan a cabo con guías de Trabajos Prácticos con el apoyo de profesores (Figura 8.4).



Figura 8.4 Clases de Laboratorio de la Asignatura Computación

a-3) Cuestionario o Encuesta

Profundizando aún más, se administró una encuesta a 74 estudiantes de 2do año con el objeto de conocer sus hábitos de interacción y acceso a Internet y su percepción en cuanto al propio aprendizaje.

Los resultados arrojaron las siguientes conclusiones:

- Cuentan con los recursos necesarios para emprender un estudio que involucre interacción con la Web.
- Frecuencia de conexión: todos los días.
- Motivo más importante o frecuente: Leer o escribir correos electrónicos y chat con amigos.
- Búsqueda de información: a través de Motores de búsqueda.
- Sitios más visitados: intereses personales.
- Acción más frecuente con la información: Guardar los sitios o leer la información a medida que la encuentra.
- Consideran que la información que le brinda la Red es confiable aunque la validan con otros sitios, con libros o anotaciones.
- Consideran que tienen experiencia para buscar información a través de Internet.

En cuanto al aprendizaje : el 49% considera que son sistemáticos en su estudio , 72% sostiene que tienen capacidad de leer analítica y comprensivamente en varios formatos, el 51% afirman que es necesario autoevaluarse en sus aprendizajes y el 60% consideran beneficioso el trabajo en grupo.

La Tabla 8.2, resume los resultados de la percepción de los estudiantes en cuanto a su propio aprendizaje.

	Sistematicidad	Análisis	Autoevaluación	Trabajo_grupo
Totalmente en desacuerdo	10.8	0.00	2.7	2.7
En desacuerdo	37.8	12.2	8.1	8.1
Indiferente	18.9	16.2	37.8	29.7
De acuerdo	20.3	52.7	43.2	48.6
Totalmente de acuerdo	10.8	18.9	8.1	10.8
No contesta	1.4	0.0	0.0	0.0
TOTAL	10.8	100	100	100

Tabla 8.2. Resultados de la percepción de los estudiantes en relación al estudio

a-4) Focus Group con implicados

Con toda la documentación recopilada de los puntos a-1), a-2) y a-3) se convocó a integrantes de la cátedra con el fin de definir el tema del OA y los lineamientos a seguir para cumplir con los objetivos .

Se arribaron a las siguientes conclusiones:

- El tema seleccionado para el OA fue “Pasaje de parámetros”.
- Presentar los contenidos principales involucrados.
- Mostrar ejemplos que involucren los contenidos con diferentes medios: texto, imagen, video, audio, etc.
- Incorporar actividades de autoevaluación y grupales

b) Especificación

b.1) Definición de los objetivos del OA:

Objetivo General: Resolver problemas utilizando el pasaje de parámetros

Objetivos específicos:

- ✓ *Entender la descomposición como forma de resolución de problemas.*
- ✓ *Dar valor a la reusabilidad en búsqueda de la eficiencia en la escritura el código.*
- ✓ *Establecer comunicación entre módulos.*
- ✓ *Comprender las ventajas de la descomposición*
- ✓ *Diferenciar los distintos tipos de pasaje de parámetros.*
- ✓ *Distinguir entre variables locales y variables globales.*

b.2) Verificación de la especificación generada con respecto a sus requerimientos

Se desarrollaron varios Escenarios de aprendizaje para describir los requerimientos del OA. Se plantearon los siguientes escenarios:

- Escenario 1 : video explicando pasaje de parámetros por valor con variables simples
- Escenario 2 : video explicando pasaje de parámetros por referencia con variables simples
- Escenario 3 : video explicando pasaje de parámetros por valor con variables estructuradas
- Escenario 4: video explicando pasaje de parámetros por referencia con variables

estructuradas.

- Escenario 5: Problemas resueltos
- Escenario 6: Actividad individual: autoevaluación
- Escenario 7: Actividad grupal
- Escenario 8: Actividad de pares

Posteriormente, se realizaron nuevas reuniones Focus Group con docentes de la cátedra de Computación con el fin de ampliar información sobre aquellos Escenarios cuya descripción resultara incompleta y/o confusa a partir de la información obtenida.

b.3) Selección de Escenarios

A partir de la validación de requerimientos, se propone la descripción de Escenarios evaluados como etapa previa al diseño de la navegación e interfaz del usuario. Las Tablas 8.3, 8.4 y 8.5 ilustran tres de los Escenarios desarrollados. En el Apéndice A, se presentan el resto de los escenarios.

Componente	Descripción
Nombre	Video 1: ejemplo variables simples
Objetivo	Proporcionar información por diversos medios: texto, imagen y sonido sobre el pasaje de parámetros por valor utilizando variables simples.
Contexto	El usuario debe tener un cuenta Moodle del curso y una computadora con lo siguiente: <ul style="list-style-type: none"> • Complemento de Adobe Flash Player 10.0.22 o posterior. • Firefox 1.1 o posterior, Safari 1.0 o posterior, Google Chrome u Opera. • Conexión de banda ancha de un mínimo de 500 Kbps.
Recursos	video
Actores	Estudiante
Set de episodios	<ol style="list-style-type: none"> 1. El usuario hace clic en el link que hace referencia al video ejemplo del pasaje de parámetros – variables simples. 2. Se despliega una ventana pop-up que presenta el objetivo del video y debajo el video embebido. 3. El usuario clickea en el video y comienza la reproducción. El video tiene controles para avanzar, retroceder, parar y pasar a pantalla completa. 4. El video presenta el problema sencillo “swap” a resolver que trabaja con variables simples. 5. El video tiene la pantalla dividida en a) el código fuente Pascal que resuelve el problema, b) la recreación de la memoria c) La pantalla de salida del programa. 6. El usuario observa como se ejecutan cada una de las sentencias del programa, las cuales están señaladas con un color. Simultáneamente en la otra parte de la pantalla se van mostrando los cambios en la memoria y en la pantalla de salida.

	<ol style="list-style-type: none"> 7. El usuario escucha a un locutor que explica los pasos más relevantes. 8. El usuario observa que los momentos claves son señalados además con globos explicativos. 9. Se resalta con el relato del locutor y visualmente el tipo de pasaje de parámetros por valor, en el sector de la memoria se destaca que no cambian los valores de los parámetros actuales cuando cambian los valores de los formales y que en realidad no se resuelve el problema con este tipo de pasaje de parámetros. 10. Se repiten los pasos 6 a 9 en donde el usuario tiene la libertad de “avanzar”, “retroceder” o “parar” el video. 11. El usuario presiona el botón “parar” que finaliza la reproducción del video.
Casos alternativos	Si el usuario no tiene los complementos necesarios, el sistema se lo indicará, podrá descargarlos y luego ver el video.
Dudas:	<ul style="list-style-type: none"> • Enunciado del problema a resolver. • Momentos claves a destacar visual y auditivamente en el programa Pascal. • Metáforas para recrear la memoria

Tabla 8.3. Escenario 1: video explicando pasaje de parámetros por valor con variables simples

Componente	Descripción
Nombre	Problemas resueltos
Objetivo	Facilitar el aprendizaje del “Pasaje de parámetros” por medio de la comprensión de las condiciones de un problema y la asimilación de la solución mediante la visualización paso a paso de la misma.
Contexto	<p>El usuario debe tener un cuenta Moodle del curso y una computadora con lo siguiente:</p> <ul style="list-style-type: none"> • Firefox 1.1 o posterior, Safari 1.0 o posterior, Google Chrome u Opera . • Lector de archivos pdf (Adobe Reader o un Visor de documentos) • Conexión de banda ancha de un mínimo de 500 Kbps .
Recursos	Archivo pdf
Actores	Estudiante
Set de episodios	<ol style="list-style-type: none"> 1. El usuario hace clic en el link que hace referencia a problemas resueltos. 2. Se despliega una ventana pop-up que contiene un mensaje indicando que se presentarán varios problemas resueltos y debajo un botón para la descarga indicando el tamaño del archivo. El archivo tiene formato pdf. 3. El usuario descarga el archivo. 4. El documento presenta un índice con enlaces a cada uno de los ejemplos. 5. El ejemplo presenta a) análisis top-down b) estructura jerárquica del programa con un diagrama c) solución con la explicación detallada de cada elemento y su función (tipos de datos, variables, ctes, parámetros, procedimientos y funciones) y las sentencias ejecutables. 6. Al final del ejemplo hay un botón que permite volver al principio del documento.

	<ol style="list-style-type: none"> 7. Se repiten los pasos 4 a 6 en donde el usuario tiene la libertad de recorrer el documento. 8. El usuario cierra el documento.
Casos alternativos	Si el usuario no tiene los complementos necesarios, el sistema se lo indicará, podrá descargarlos y luego ver el documento pdf.
Dudas:	<ul style="list-style-type: none"> • Enunciado de los ejemplos resueltos. • Momentos claves a destacar visual y auditivamente en el programa Pascal.

Tabla 8.4. Escenario 5: Problemas resueltos

Componente	Descripción
Nombre	Actividad grupal
Objetivo	Fomentar el trabajo colaborativo para la resolución de un problema que involucra “Pasaje de Parámetros” y fortalecer el aprendizaje de una mayor cantidad de estudiantes.
Contexto	<p>El usuario debe tener un cuenta Moodle del curso y una computadora con lo siguiente:</p> <ul style="list-style-type: none"> • Complemento de Adobe Flash Player 10.0.22 o posterior. • Firefox 1.1 o posterior, Safari 1.0 o posterior, Google Chrome u Opera. • Conexión de banda ancha de un mínimo de 500 Kbps .
Recursos	<ul style="list-style-type: none"> • Herramienta de comunicación “CHAT” del Curso Moodle. • Herramienta de colaboración WIKI por grupos en el Curso Moodle.
Actores	Grupo de Estudiantes y Profesor
Set de episodios	<ol style="list-style-type: none"> 1. Se forman equipos de tres a cinco estudiantes para resolver el problema. 2. Antes de iniciar la resolución del problema los estudiantes leen cuidadosamente las orientaciones específicas para trabajar con la WIKI que se encuentra en el curso Moodle. 3. Para la resolución de la consigna los estudiantes conversan y/o discuten la mejor alternativa de solución, pero sólo entre los integrantes del equipo. 4. Los integrantes del grupo utilizan la Wiki para la creación colectiva de la solución. 5. Los integrantes van editando el documento ,guardan los cambios , hasta llevar a una solución consensuada por todos los integrantes del problema.. 6. Al finalizar la actividad se envía un correo al profesor para su evaluación y devolución.
Casos alternativos	
Dudas:	<ul style="list-style-type: none"> • Enunciado del problema a resolver.

Tabla 8.5. Escenario 7: Actividad grupal

8.2.3. Diseño

a) Diseño del OA, desde el punto de vista pedagógico

En el diseño del OA, desde el punto de vista pedagógico se trabajó con integrantes de la cátedra de Computación (Focus Group):

- a) La definición de los enunciados de los problemas de: los videos, las actividades grupales y de pares (ver Apéndice A) y los ejemplos resueltos (ver Apéndice B).
- b) Las metáforas para describir las acciones en los videos (asignación de memoria de las variables, pasaje de parámetros por valor y referencia, código del programa Pascal, salida del programa) a fin de definir el estilo de la interfaz desde el punto pedagógico y técnico.

Dichas metáforas fueron pensadas teniendo en cuenta en cada problema:

- Debe ser claro lo que hace, de forma que si el estudiante ejecuta los pasos sepa qué, cómo y cuándo hacerlo (comprensible).
- El orden de ejecución de las instrucciones del programa debe estar perfectamente indicado (preciso).
- Si se presenta un obstáculo insuperable que no arroja el resultado esperado o si se resuelve el problema (realizable)

En el Apéndice A, se presenta una introducción al tema “Pasaje de Parámetros”.

b) Diseño de la Navegación

Es necesario desarrollar un esquema general del OA, el cual indicará cómo están interrelacionados sus componentes (ver subsección 2.3.3 del capítulo 2): los contenidos, objetivos, actividades de aprendizaje y la evaluación.

Para analizar las diferentes alternativas que se tienen para organizar cada una de las componentes, se utilizó la técnica de ordenamiento de tarjetas o Card sorting con cuatro integrantes de la cátedra de Computación.

Se partió del siguiente conjunto de tarjetas sin sugerir títulos para las categorías principales (Figura 8.5).

1 OBJETIVOS Página 1	2 EJEMPLO 1 (PROBLEMA RESUELTO) Página 2	3 ACTIVIDAD 1 (MULTIPLE CHOICE) Página 3	4 INICIO Página 4	5 PASAJE DE PARÁMETROS POR VALOR (INTRODUCCION) Página 5
6 INTRODUCCION GENERAL Página 6	7 PROBLEMA 3 CON VARIABLES SIMPLES Página 7	8 PROBLEMA 2 CON VARIABLES ESTRUCTURADAS Página 8	9 PASAJE DE PARÁMETROS POR REFERENCIA (INTRODUCCION) Página 9	10 PROBLEMA 4 CON VARIABLES ESTRUCTURADAS Página 10
11 ACTIVIDADES (INTRODUCCION) Página 11	12 ACTIVIDAD 3 (DE A PARES) Página 12	13 ACTIVIDAD 2 (GRUPAL) Página 13	14 PROBLEMA 1 CON VARIABLES SIMPLES Página 14	15 PROBLEMAS RESUELTOS (INTRODUCCION) Página 15
16 EJEMPLO 3 (PROBLEMA RESUELTO) Página 16	17 EJEMPLO 2 (PROBLEMA RESUELTO) Página 17	18 ACTIVIDAD 3 (DE A PARES) Página 18		

Figura 8.5. Tarjetas para la técnica Card Sorting

La prueba se realizó de forma manual (con tarjetas en papel) para que los participantes las agruparan en una pizarra. El proceso se ilustra en las Figuras 8.6, 8.7 y 8.8.

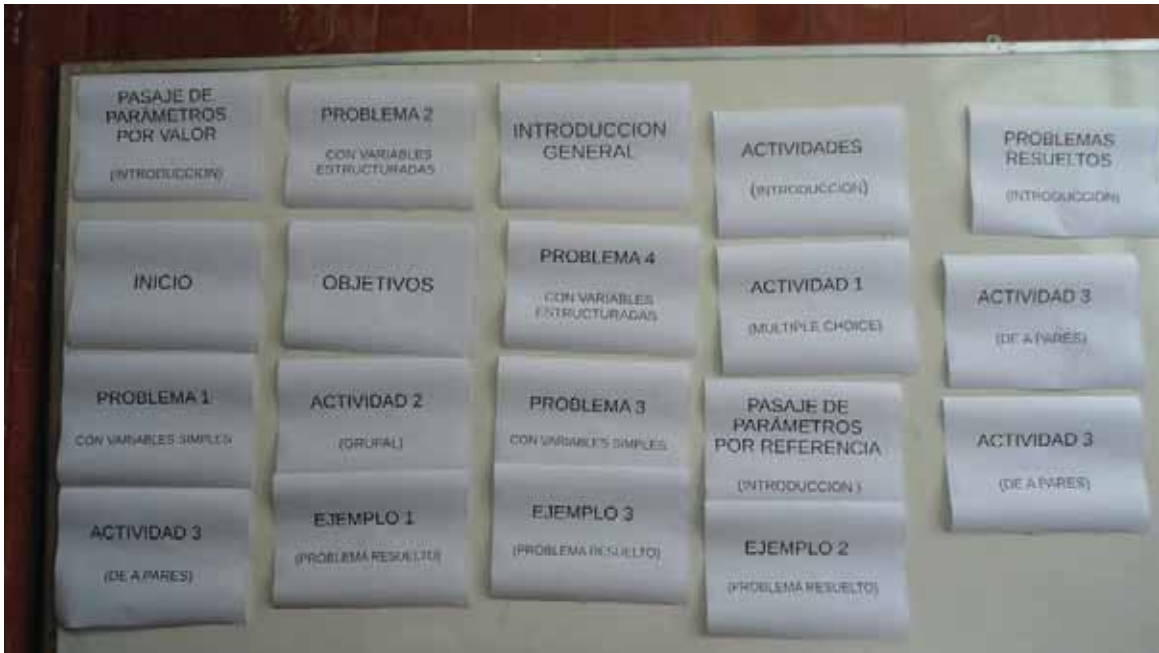


Figura 8.6. Tarjetas disponibles



Figura 8.7. Ordenamiento de las tarjetas por los expertos

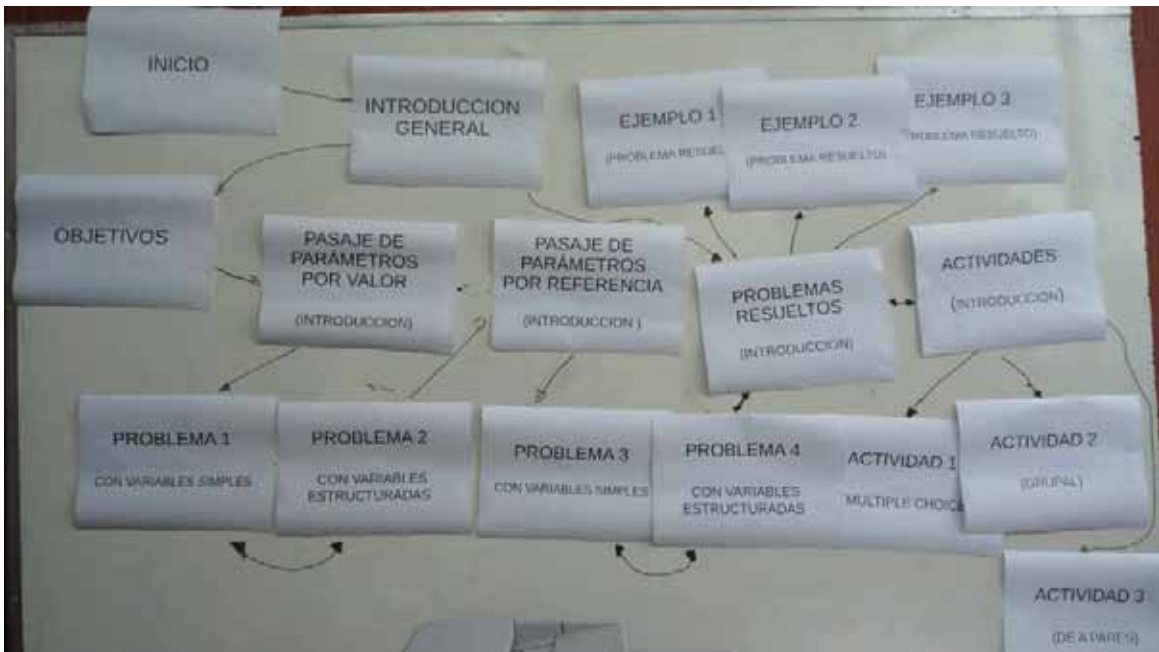


Figura 8.8. Recorrido ideal del OA: de acuerdo a la jerarquía de los expertos

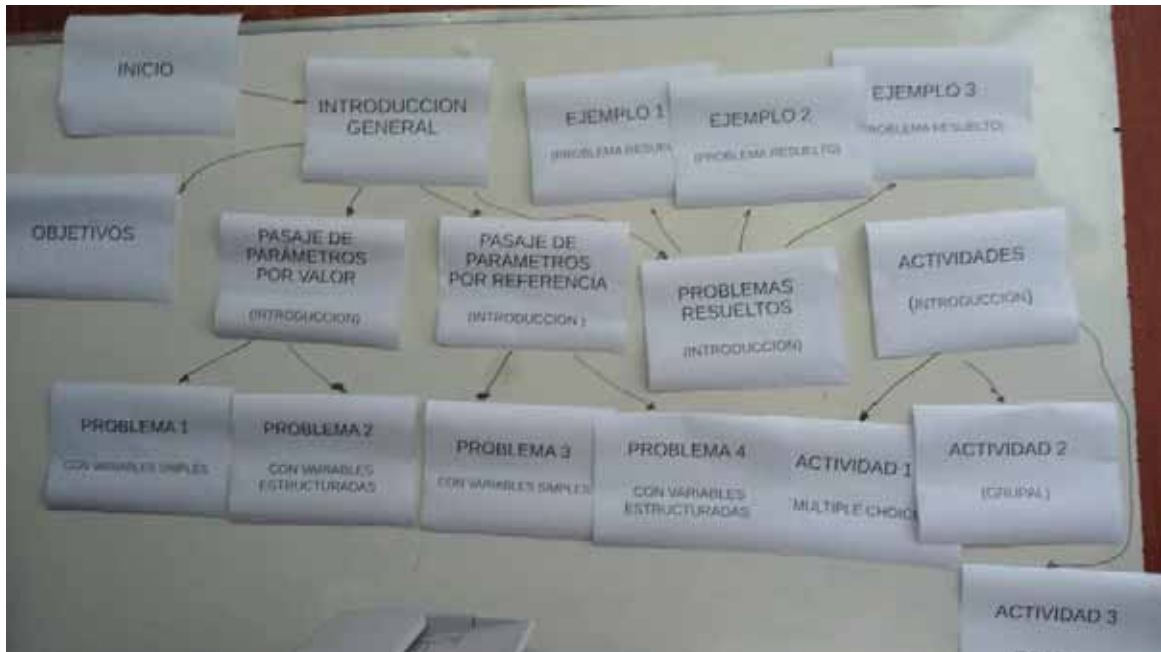


Figura 8.9. Recorrido del OA

En la Figura 8.10 se presenta el Diagrama de Navegación, luego de la aplicación de la técnica Card Sorting.

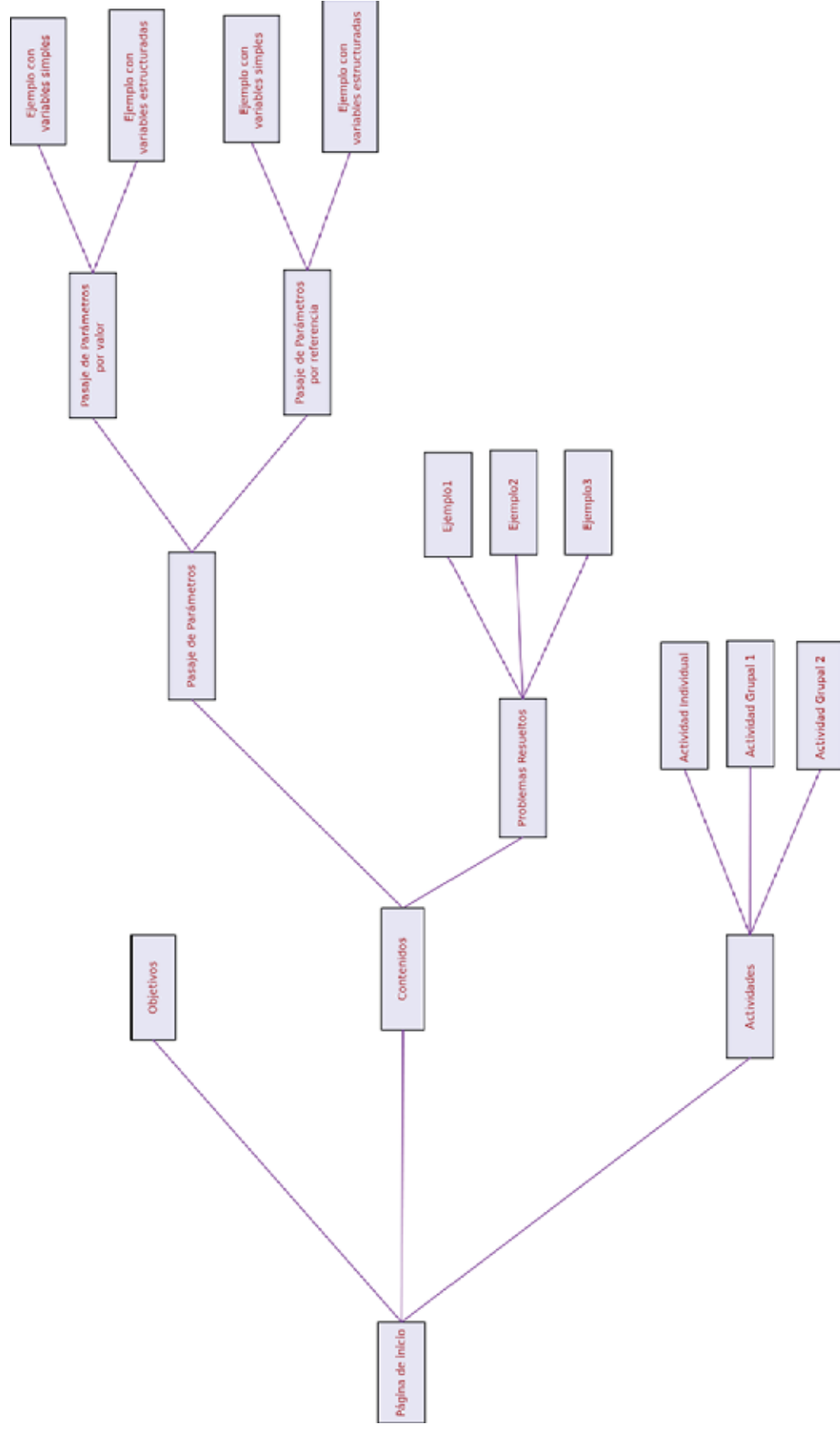


Figura 8.10. Diagrama de Navegación

c) Actividades

c-1) Desarrollo de prototipos

Se realizaron una serie de prototipos en papel que tras sucesivas evaluaciones (con expertos evaluadores y con los integrantes de la cátedra de Computación: profesores y auxiliares alumnos) los cuales evolucionaron hasta concretarse las maquetas digitales finales del OA correspondientes a los distintos componentes del OA. Las evaluaciones se realizaron utilizando las técnicas Recorrido Cognitivo y Pensamiento en voz alta.

i) Prototipo papel para los videos

Ya pueden observarse algunas metáforas encontradas plasmadas en el papel (Figura 8.11 y Figura 8.12)

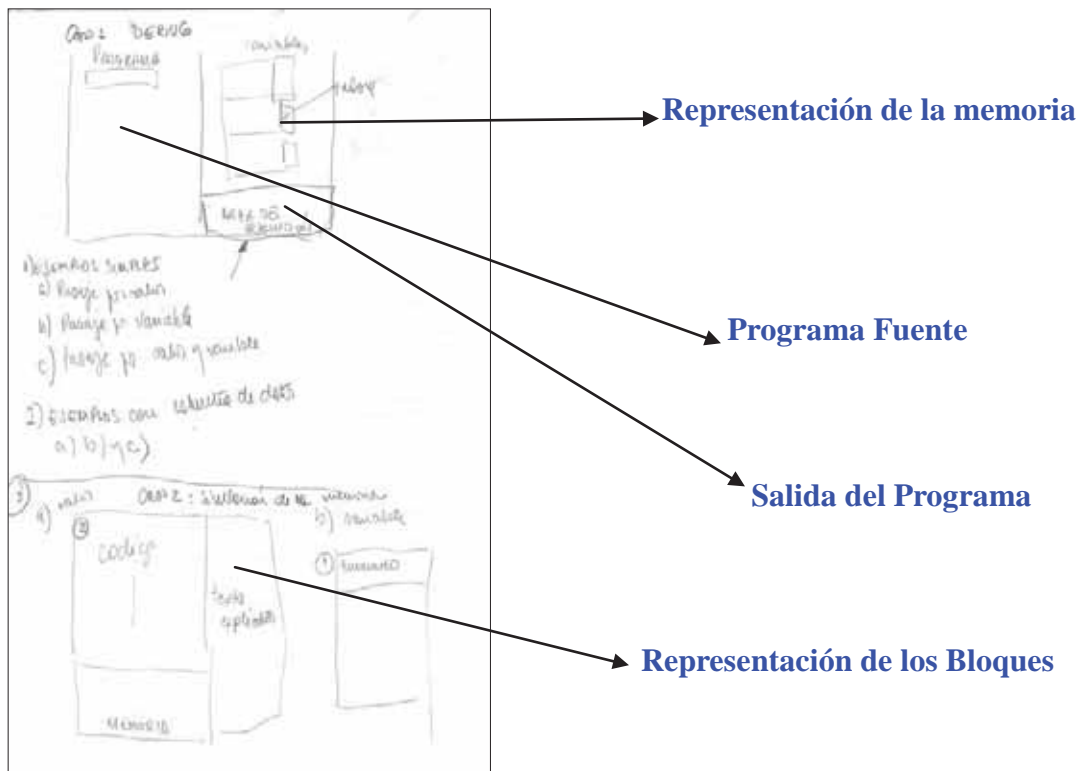


Figura 8.11. Prototipo 1 en papel para videos

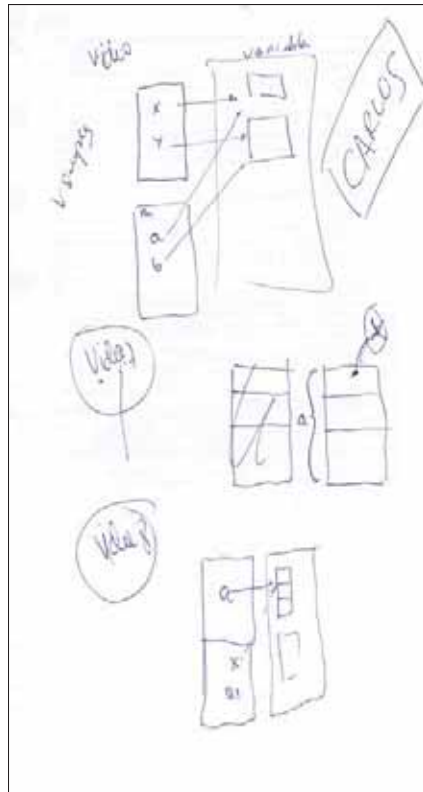


Figura 8.12. Prototipo 2 en papel para videos

ii) Prototipo en papel para Problemas Resueltos

En la Figura 8.13 puede observarse el esquema general en papel propuesto para los problemas resueltos

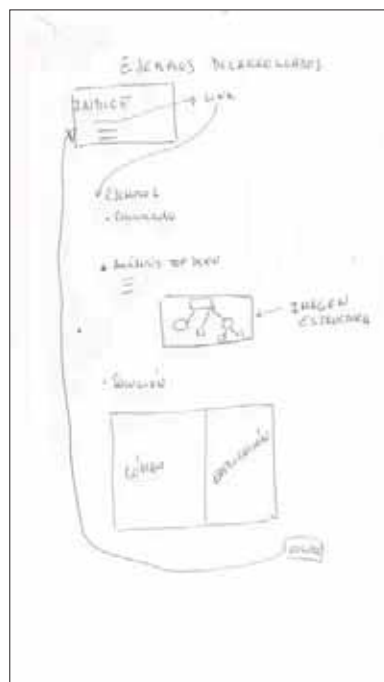


Figura 8.13. Prototipo en papel para Problemas Resueltos

El esquema planteado es el siguiente: se presenta un índice con el título de cada uno de los problemas a resolver y luego para cada problema: su enunciado, el análisis top-down y la solución.

iii) Maquetas digitales para videos

A continuación se ilustra la evolución de las distintas maquetas digitales. Para el video 1 (Figura 8.14 y Figura 8.15) y para el video 3 (Figura 8.16 y Figura 8.17)

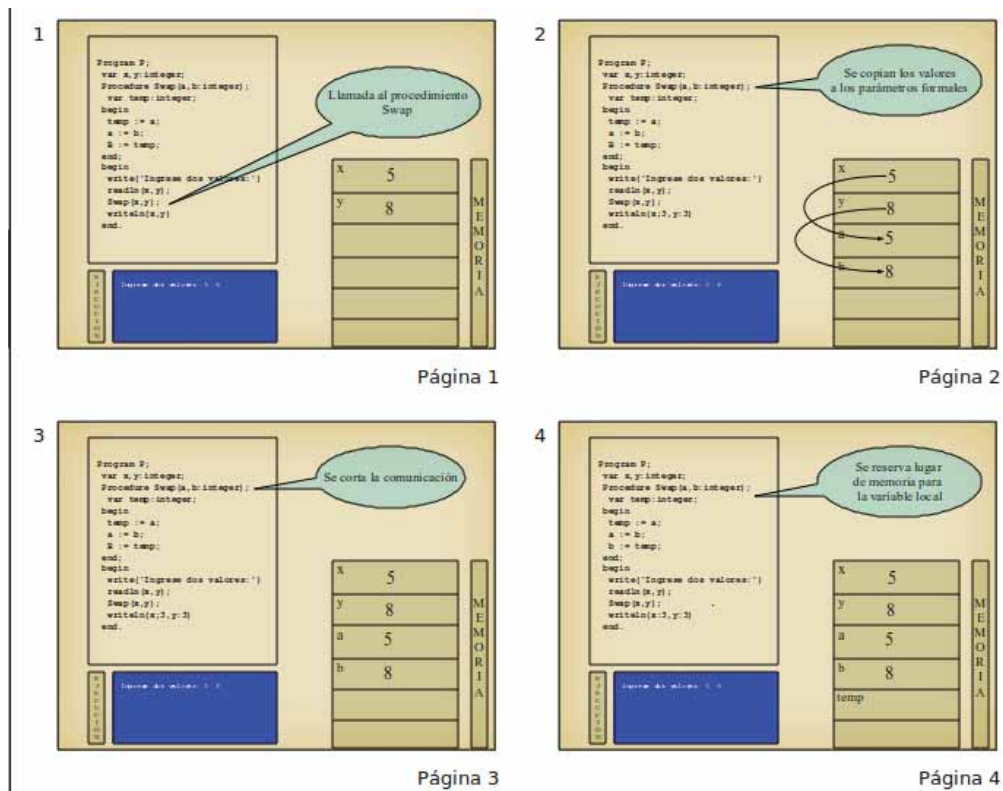


Figura 8.14. Maqueta digital para el video 1 – Versión 1

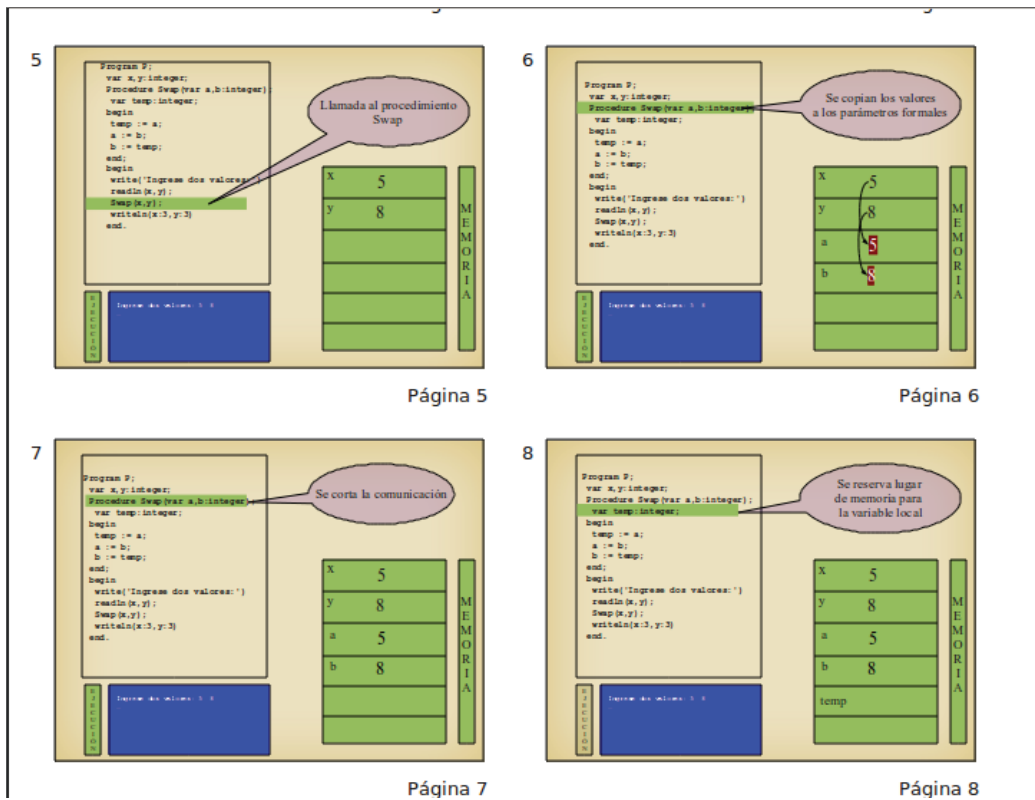


Figura 8.15. Maqueta digital para el video 1 – Versión 2

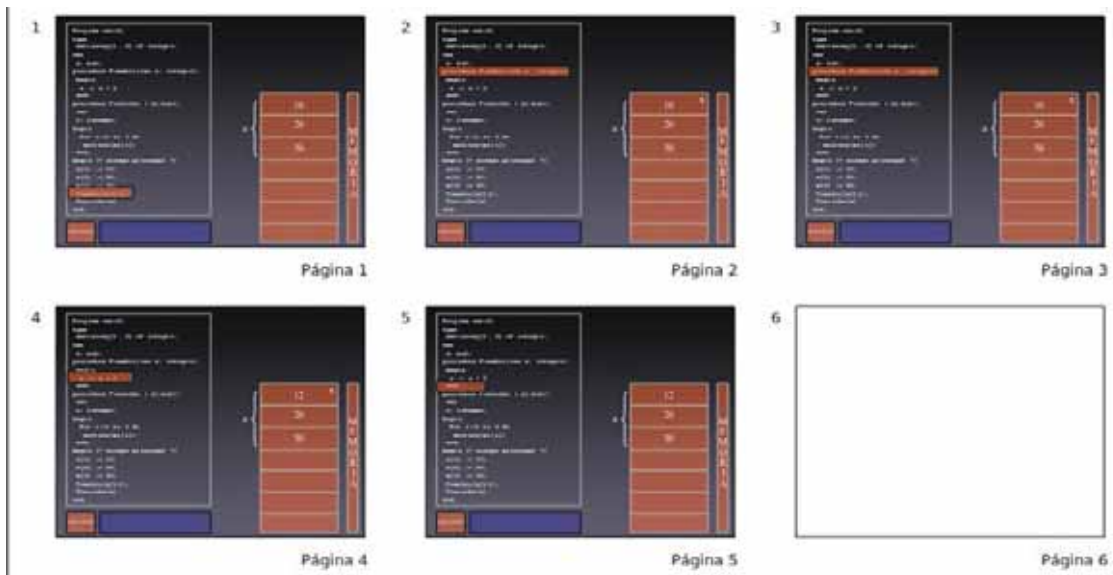


Figura 8.16. Maqueta digital para el video 4 – Versión 1



Figura 8.17. Maqueta digital para el video 4 – Versión 2

iv) Story board para videos

Las Figuras 8.18 y 8.19 muestran los Storyboard de los videos 1 y 4 respectivamente



Figura 8.18. Storyboard para el video 1



Figura 8.19. Storyboard para el video 4

v) Maqueta digital para Problemas Resueltos

En el Apéndice B, se presenta el documento correspondiente a los Problemas Resueltos.

8.2.4. Puesta en funcionamiento del OA

a) Construcción del prototipo de Software (Creación del OA)

Se construyó un prototipo de software utilizando la herramienta ExeLearning (ver subsección 3.3.2 del capítulo 3).

Las Figuras 8.20, 8.21, 8.22, 8.23 y 8.24, muestran algunas pantallas del prototipo de software.



Figura 8.20. Pantalla correspondiente al Objetivo del OA



Figura 8.21. Pantalla correspondiente a la presentación del Video1



Figura 8.22. Pantalla correspondiente a la Introducción Pasaje de parámetros por referencia



Figura 8.23. Pantalla correspondiente a la Presentación del video 4



Figura 8.24. Pantalla correspondiente a la Presentación de los Problemas resueltos

En esta etapa se evalúan los atributos del OA mediante la inspección de sus características. El objetivo es determinar si un elemento digital puede ser considerado un OA.

- Ser una unidad independiente de instrucción: el OA desarrollado corresponde al tema “Pasaje de Parámetros” de la Unidad III del Curso de Computación.
- Seguir un formato estándar de instrucción: la herramienta ExeLearning permitió empaquetar el OA al formato SCORM 1.2.

- Ser relativamente pequeño :

Consideramos lo propuesto por LOM (2002) que establece 4 niveles de granularidad los cuales indican el tamaño lógico del OA. Dichos niveles oscilan entre el nivel de agregación más pequeño (nivel 1) hasta OAs que contienen a su vez OAs de nivel 3 u otros OAs de nivel 4:

- OA nivel 1: Se refiere al nivel más atómico o granular de agregación, eg: imágenes, segmentos de texto o vídeos se corresponden con los assets de SCORM (2009).
- OA nivel 2: Una lección con un objetivo de aprendizaje, con un tipo de contenido (datos y conceptos, procedimientos y procesos, reflexiones) el cual puede estar conformado por varios OAs de nivel 1, y finalmente y actividades de evaluación y práctica (opcional). Se corresponden con los SCOs (Sharables Content Object) de SCORM e incluye archivos ZIP y otros paquetes que pueden ser accedidos como archivos individuales pero que a su vez forman un único recurso agregado.
- OA nivel 3: Un módulo de aprendizaje compuesto por un conjunto de lecciones (OA nivel 2) con un mínimo de tres tipos de contenidos (datos y conceptos, procedimientos y procesos o principios) y actividades de evaluación y práctica (opcional). Comprenden los recursos que pueden ser descompuestos en dos o más OAs y se corresponden con el concepto de curso y con los Content Aggregation de SCORM.
- OA nivel 4: Un curso compuesto por un conjunto de módulos (OA nivel 3) con un mínimo de tres tipos de contenidos (datos y conceptos, procedimiento y procesos) y actividades de evaluación y práctica (opcional).

Siguiendo esta clasificación el OA “Pasaje de Parámetros” tiene nivel de granularidad 2 y por su estructura puede incorporarse a un OA de mayor granularidad: un unidad de aprendizaje: Funciones y Procedimientos.

- Una sucesión de OAs debe tener un contexto: El OA se incorpora al Curso de Computación de la Plataforma Educativa Moodle. Este Curso contiene diversos materiales que apoyan al curso presencial.
- Contener un registro de metadatos: El OA contiene metadatos de acuerdo al estándar Dublin Core (Figura 8.25).

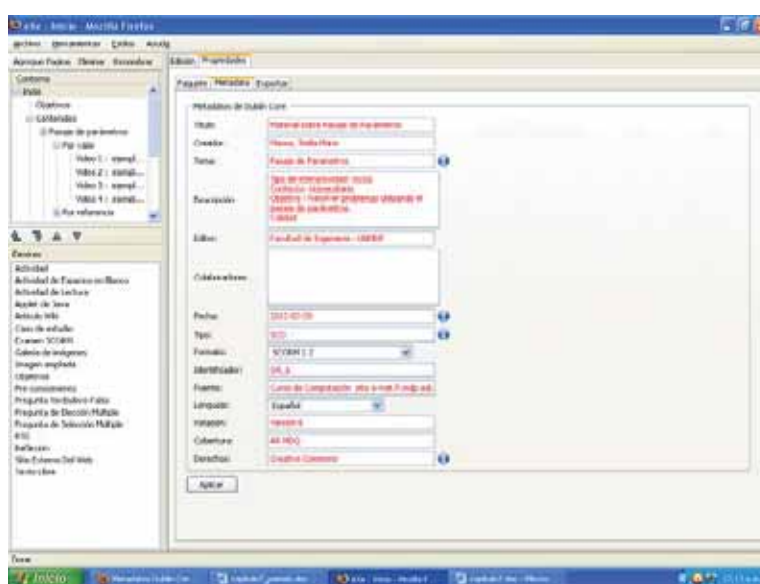
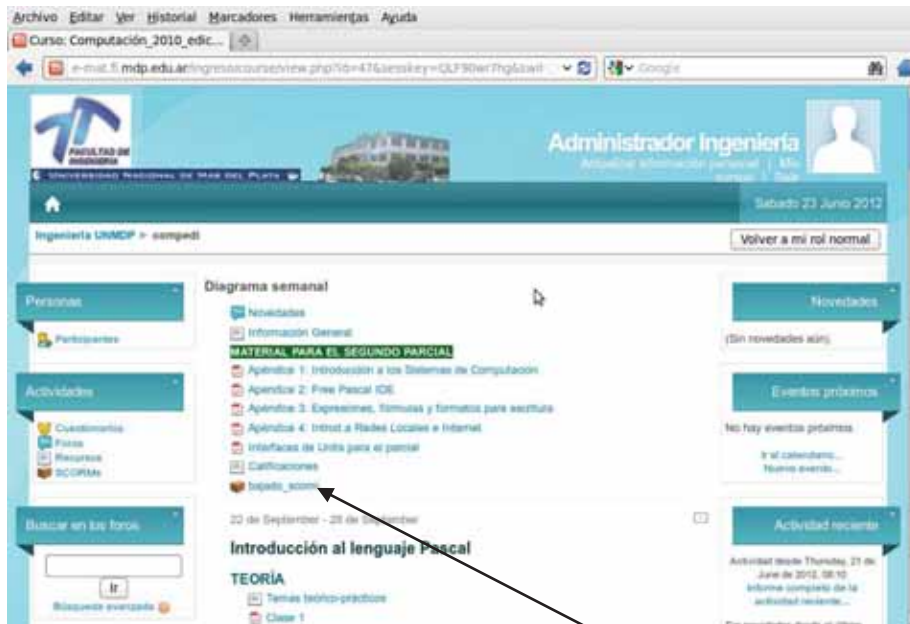


Figura 8.25. Incorporación de Metadatos en el OA. Estándar Dublin Core

b) Implementación del Prototipo de software

El OA fue incorporado al Curso de prueba de la Plataforma Educativa Moodle (Figura 8.26)



Recurso Scorm

Figura 8.26. Incorporación del prototipo de software (SCORM) en un Curso de Prueba de la Plataforma Educativa

Luego de su incorporación en el curso, la página de inicio es la que ilustra la Figura 8.27.



Figura 8.27. Vista del OA en la Plataforma Moodle

En esta etapa del Modelo MPOBA, las técnicas de evaluación utilizadas con expertos fueron: Evaluación Heurística, Observación del experto y Recorrido Cognitivo. A

continuación se presentan los resultados obtenidos y la posterior discusión de los mismos.

b-1) Pasos de la Evaluación Heurística

1. Planificación

i) Selección de los evaluadores: de acuerdo a la subsección 6.2.1.5. , inciso b-1 del capítulo 6, fueron seleccionados 5 evaluadores con los siguientes perfiles:

- Consultor de experiencia de usuario (Evaluador 1)
- Consultor de experiencia de usuario y experto en contenido (Evaluador 2)
- Experto de software relacionado con las tecnologías Web (Evaluador 3)
- Experto en experiencia de usuario (Evaluador 4)
- Especialista del área del contenido educativo (Evaluador 5)

ii) Elaboración de un paquete de inspección: se utilizó la Plantilla **GEHOA** explicada en la subsección 7.2.2. del capítulo 7. En el Apéndice C se presenta dicha plantilla.

iii) Selección de escenarios de tareas para realizar la evaluación: los expertos debían recorrer todo el OA pues su granularidad es pequeña.

2. Puesta en marcha (evaluación individual con cada experto)

i) Entrenamiento previo a la evaluación:

Se informa a los evaluadores sobre el tema, contexto de aplicación, perfiles de usuario, descripción del sistema. (Figuras 8.28).



Figura 8.28. Entrenamiento previo a la evaluación

ii) Evaluación del OA propiamente dicha por parte del experto:

La experimentación se realizó en cinco sesiones de 2 horas (cada una con cada experto) las cuales fueron filmadas con el consentimiento de los evaluadores (Figura 8.29). También dos observadores tomaron notas de la evaluación. Cada puntaje fue registrado en la Plantilla **GEHOA** para su posterior análisis. Los evaluadores recorrieron el OA para poder responder a cada uno de los criterios heurísticos.



Figura 8.29. Evaluación heurística

Cada experto evaluador puntuó los sub-criterios heurísticos explicados en la subsección 7.2.2 del Capítulo 7, con una escala de 0 (no es un problema de usabilidad) a 4 (catástrofe de Usabilidad). La tabla con los resultados se puede ver en el Apéndice C.

Como se explicó en la subsección 7.2.2 del Capítulo 7, se tuvieron en cuenta dos indicadores de calidad: la valoración individual de cada categoría y la valoración global que se calculó como promedio ponderado, en donde cada peso es la proporción de criterios heurísticos asociados a cada categoría tal se muestra en la Tabla 8.6.

CATEGORÍA	NRO DE SUB-HEURISTICAS	PESO
Significatividad Psicológica	10	16%
Significatividad lógica	20	33%
Diseño de interfaz	17	28%
Diseño de navegación	14	23%
Total	61	100%

Tabla 8.6. Pesos de cada categoría

Se calculó la puntuación de cada categoría individualmente y se comparó con el puntaje catástrofe máximo que resulta de valorar todos los sub-criterios con el valor 4. En el Gráfico 8.1, se ilustran los resultados obtenidos. Se puede observar que en todas las categorías, los puntajes obtenidos están muy alejados del puntaje catástrofe. Lo que indica que los problemas de usabilidad no son graves.

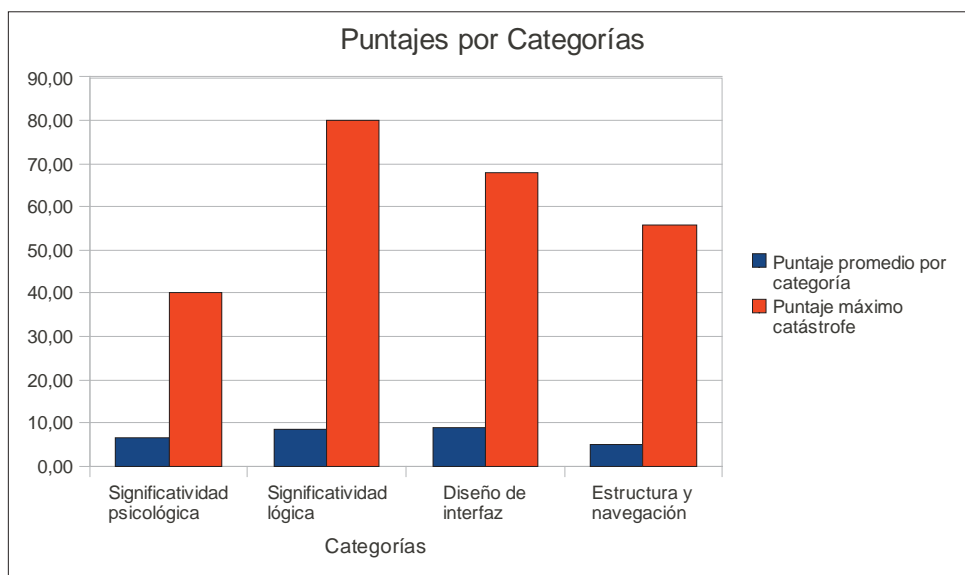


Gráfico 8.1. Puntuación promedio individual de cada categoría

En este sentido, si comparamos las categorías (Gráfico 8.2), se observa que la categoría “Significatividad psicológica” es la peor puntuada. Analizando los sub-criterios evaluados de esa categoría en particular,

- ✓ *El OA propone diferentes contenidos/actividades para cada tipo/nivel de competencia del alumno: en este caso el OA contiene diferentes contenidos y actividades pero sin considerar el nivel de competencia.*
- ✓ *Se induce a la participación directa por parte de los estudiantes en diversas*

actividades: las actividades planteadas se realizan por fuera del OA, es una limitación de la herramienta Exelearning. Las actividades se realizan utilizando en un caso desde la WIKI de Moodle y en el otro desde un espacio en el curso donde suben el documento con el programa fuente.

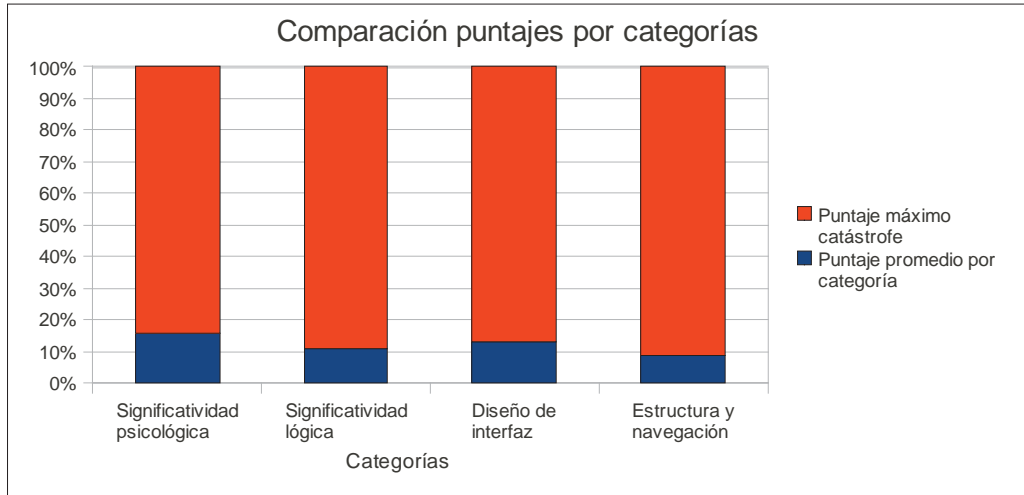


Gráfico 8.2. Comparación de Puntuación promedio individual de cada categoría con puntaje máximo

En cuanto a la valoración global de la calidad del OA, en el Gráfico 8.3 se muestran los puntajes ponderados para cada evaluador. También se indica el puntaje ponderado catástrofe que resulta de considerar el valor 4 (máximo valor individual) para cada criterio.

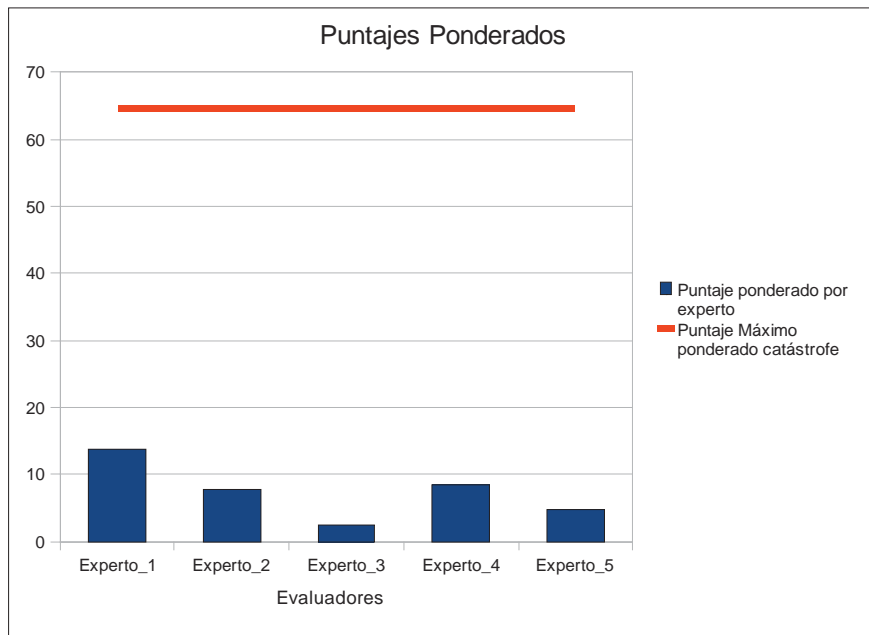


Gráfico 8.3. Puntajes ponderados y puntaje máximo catástrofe

Se acompañan a estos resultados las medidas estadísticas de la Tabla 8.7:

Promedio ponderado	7.47
Mediana	3.48
Desvío Estándar	4.2
Coef.variación	56.24
Promedio Ponderado Catástrofe	64.19

Tabla 8.7. Medidas estadísticas de la Evaluación Heurística

Se observa que el puntaje promedio ponderado no es representativo, la dispersión es alta debido a la diferencia en los puntajes con los evaluadores. Podemos tomar la mediana como representante, que no se ve afectada por valores extremos. Además el coeficiente de variación indica que la dispersión de los valores respecto del valor medio es del 56.24 %. Analizando estos resultados, la valoración global (VG) podría considerarse alta:

$$\text{VGOA/VIDEAL} = 0,05 \text{ (Valoración global del OA dividido la valoración ideal).}$$

b-2) Recorrido cognitivo y Observación del usuario

Este análisis pretende rescatar otras “aristas” del objeto que no se ven reflejadas en la evaluación numérica. Las Figuras 8.30 y 8.31 muestran imágenes de momentos de la evaluación realizada.





Figura 8.30. Observación del experto



Figura 8.31. Recorrido Cognitivo

De la transcripción de los videos y las observaciones se extrajeron las siguientes conclusiones agrupadas en categorías:

- Aspectos positivos
 - ✓ Los videos captan el interés del usuario y refuerzan los contenidos.
 - ✓ El audio es importante porque utiliza el lenguaje específico y es importante que lo aprendan los alumnos. El audio refuerza.
 - ✓ El audio es importante porque muestra el proceso oculto, sino sería lo mismo que utilizar el “paso a paso” en el IDE de Pascal. Tiene un valor agregado.
 - ✓ Es adecuado que se presente un problema con arreglos y otro con variables simples para que el estudiante puede identificar el proceso de pasaje de parámetros.
 - ✓ No hay elementos que distraigan la atención.
 - ✓ Se promueve la creación de nuevas ideas y la búsqueda de nuevos procedimientos/técnicas/métodos para la resolución de problemas o tareas a través de las actividades grupales.
 - ✓ Por tratarse de un recurso complementario a las clases presenciales, el nivel de complejidad es adecuado.
 - ✓ Factibilidad de reutilización en otros cursos de programación inicial o como material previo de cursos de programación avanzada.
 - ✓ Del análisis del lenguaje gestual, considerando las expresiones universales, se interpretan los estados de ánimo del experto al interactuar con el OA: Concentración al recorrer el OA, Aceptación de los conceptos presentados, Interés en el material audiovisual.

- Propuestas :
 - ✓ Incorporar una máquina virtual con el IDE al curso virtual.
 - ✓ Agregar un Cuestionario de autoevaluación antes de pasar a un nuevo tema (de Pasaje de Parámetros por valor a Pasaje de Parámetros por referencia).
 - ✓ En la consignas de los videos, cuando se hace click en un concepto clave, se despliegue la definición o al estilo de un audiolibro.
 - ✓ Incorporar FAQ (preguntas frecuentes).
 - ✓ Incorporar un Glosario de términos.

- Aspectos a mejorar:
 - a) Criterios pedagógicos para evaluar objetos de aprendizaje-Categoría Significatividad lógica

Contenidos:

- ✓ Luego de los enunciados de los problemas de los videos, agregar una figura que ilustre lo que se tiene que resolver (i.e. el proceso swap para el video 1, uno ejemplo numérico para video 3).
- ✓ Errores tipográficos en algunos textos.
- ✓ Falta la Bibliografía o Referencias Bibliográficas
- ✓ Incorporar el lote de prueba al enunciado de los problemas de los videos.
- ✓ En los casos de pasaje de parámetros por valor no se resuelve el problema (e.g. video 1: no intercambia los valores). La consigna tendría que decir lo que “debería hacer pero no hace”.

Actividades:

- ✓ La Actividad 1 individual (multiple choice) no es significativa para el aprendizaje.
- ✓ Existe posibilidad de que el estudiante reciba feedback personalizado relativo a la calidad o corrección de sus respuestas pero fuera del OA.

- b) Criterios técnicos para evaluar objetos de aprendizaje - Categoría Diseño de interfaz

Lenguaje de los usuarios:

- ✓ En el relato de los videos, sólo puntualizar o reforzar lo más importante.
- ✓ Utilizar solo el modelo de representación de la memoria de los videos impares.
- ✓ En el video 7 colocar más leyendas para destacar las acciones o cambios con colores.
- ✓ En los Problemas Resueltos, Ejemplo 1: el texto explicativo no es amigable, no está ubicado en la posición donde ocurren las acciones o se encuentran los objetos del programa (es un texto plano confuso).

Control y libertad del usuario:

- ✓ En la Actividad 1 individual - ejercicio 1 no deja elegir correctamente las opciones ni volver para atrás.
- ✓ Puede haber problemas con las ventanas emergentes si el navegador del usuario lo tiene bloqueado.

Diseño estético y minimalista:

- ✓ En los objetivos aumentar el tamaño de la fuente.
- ✓ Modificar la ubicación de cada video (centrado).
- ✓ El video 2 y el video 3, terminan y se queda con la última imagen congelada.
- ✓ En la introducción a los contenidos, “despejar” el texto, está muy arriba de la ventana.
- ✓ Separar con varias líneas el texto que acompaña a los videos del lugar donde se encuentra el video.

Consistencia y Estándares:

- ✓ Modificar los colores de fondo oscuro de algunos videos, es confuso.
- ✓ Homogeneizar: todos los fondos iguales al que se utiliza en el video 1.

Reconocimiento más que memoria:

- ✓ Resaltar en los textos los conceptos importantes.
- ✓ Modificar el fondo de los globos que indican los procesos en el análisis top-down de los ejemplos resueltos.
- ✓ En los videos, enfatizar aún más las acciones que se están ejecutando, por ejemplo que aparezca el puntero del mouse.
- ✓ Cuando se refiere a variables estructuradas, aclarar que en este caso se trata de un vector.

Errores:

- ✓ En la Actividad 1 individual, cuando se produce un error, no se informa de forma clara.

Luego de la evaluación de los expertos se realizaron modificaciones en el OA. A continuación, se ilustran con imágenes algunos de los cambios realizados.

- Estructura del OA (Figura 8.32)
 - a) En el prototipo de software se habían incorporado varias versiones de un mismo problema a resolver con distintos fondos de pantalla y metáforas diferentes. A partir de los resultados de la evaluación realizada por los expertos sólo se dejaron aquellos que recibieron una evaluación positiva.
 - b) La Actividad 1 – individual tuvo muchas críticas desfavorables, por lo que se decidió eliminar.

Nota: ver Figura 8.27 para comparar las modificaciones efectuadas.

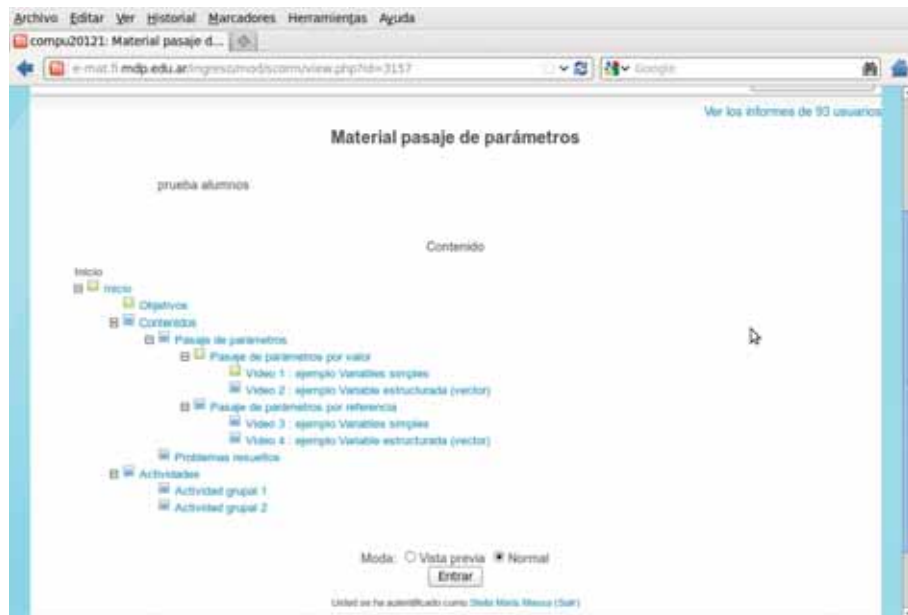


Figura 8.32. Estructura del OA modificada

- Apariencia
 - a) Se modificaron algunas fuentes de los textos (Figura 8.33). Para comparar las modificaciones efectuadas ver la Figura 8.20.
 - b) Se agregaron espacios a los textos (Figura 8.34). Para comparar las modificaciones efectuadas ver la Figura 8.24.



Figura 8.33. Modificación de fuente y tamaño



Figura 8.34. Modificación de espacios

- Aspectos pedagógicos
 - a) Se incorporaron ejemplos a los problemas de los videos (Figura 8.35 y Figura 8.36). Para comparar las modificaciones efectuadas ver las Figuras 8.21 y 8.23.
 - b) Se utilizó un único tipo de Modelo de Representación de la Memoria y un único fondo de pantalla.
 - c) Se destacaron los conceptos claves (Figura 8.37). Para comparar las modificaciones efectuadas ver la Figura 8.22.



Figura 8.35. Modificación del video 1



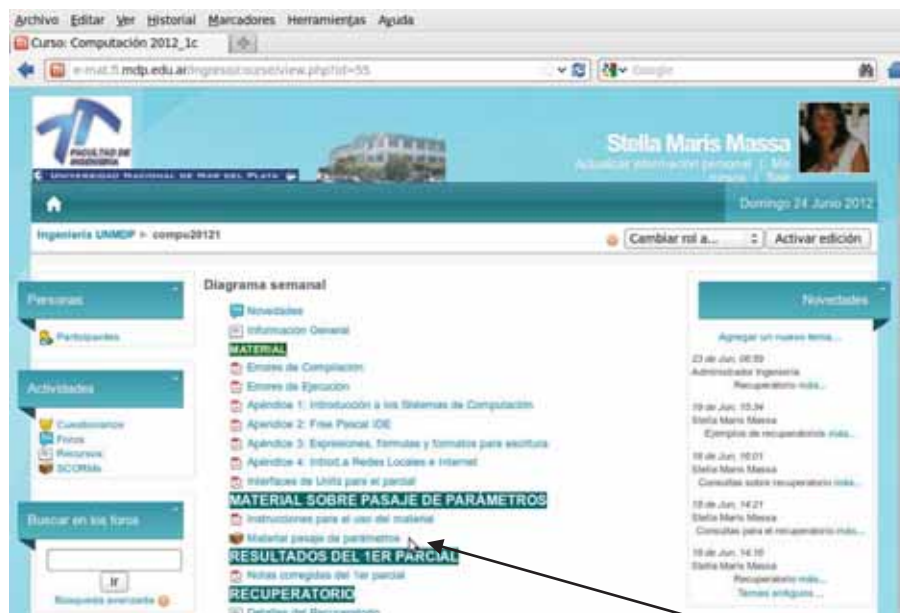
Figura 8.36. Modificación del video 4



Figura 8.37. Modificación de textos

c) Implantación del Prototipo de Software

En esta etapa se incorpora el OA al Curso Computación de la Plataforma de la Plataforma Educativa de la Facultad de Ingeniería de la UNMDP (Figura 8.38) para que este disponible para estudiantes y de esta manera se produzca un proceso de interacción del OA con los usuarios finales.



Implantación del OA

Figura 8.38. OA en la Plataforma Educativa Moodle

El OA fue recorrido por 33 estudiantes del curso de Computación, algunos de ellos desde sus computadoras (13) y otros en una clase práctica (20).

c-1) Cuestionarios

Al grupo que recorrió el OA en la clase práctica se le administró el cuestionario **CUSEOA** que fue explicado en la subsección 7.3.2. del Capítulo 7.

Este instrumento es utilizado para la evaluación de la calidad del OA desde el punto de vista del usuario final (el estudiante) y para obtener información cuantitativa y cualitativa sobre el grado de satisfacción al interactuar con el OA. Consta de dos partes:

- i. Reacción global al OA (mediante el método del Diferencial Semántico y la valoración de recomendación del OA)
- ii. Dimensión pedagógica y técnica (mediante una escala de Likert)

c-1.1) Resultados y análisis

i) Reacción global al OA

Se le presenta al estudiante una serie de adjetivos bipolares y se solicita que lo clasifique en una escala 7 puntos (Ver apéndice D).

Para evitar prejuicios, sólo se emplearon escalas positivas. El rango real es -3 a +3. Donde +3 significa máxima valoración para el descriptor de la izquierda y -3 significa máxima valoración para el descriptor de la derecha. La Tabla 8.8, muestra los puntajes presentados a los estudiantes, y los puntajes reales.

PUNTAJE	1	2	3	4	5	6	7	
difícil								fácil
frustrante								satisfactorio
aburrido								ameno
rígido								flexible
PUNTAJE REAL	-3	-2	-1	0	1	2	3	

Tabla 8.8. Escala de Diferencial Semántico

Se observa que se agrupan en el primer paso las palabras Flexible con Fácil y Satisfactorio con Ameno. A partir del dendograma se puede concluir que para los estudiantes:

- La facilidad del OA se jerarquiza con la flexibilidad (integración entre la calidad pedagógica y técnica).
- Les resulta satisfactorio si es ameno (el recurso responde a la satisfacción del estudiante si le resulta ameno).
- No obstante estos adjetivos, no deben considerarse equivalentes, sino sólo jerárquicamente relacionados
- Con respecto a la Valoración de recomendación del OA a otros estudiantes, los resultados obtenidos se muestran en la Figura 8.39.

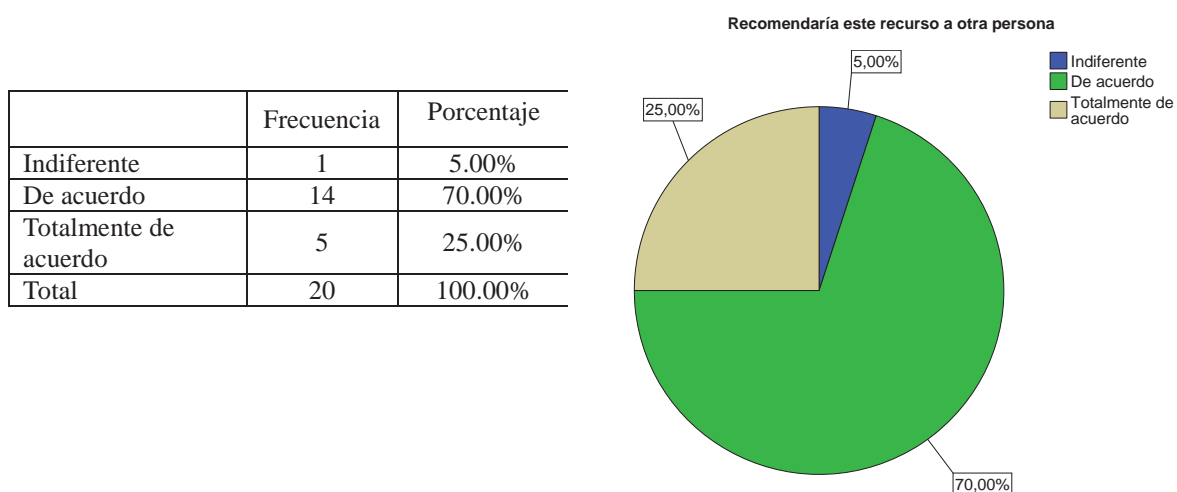


Figura 8.39. Recomendación de los estudiantes

El 95% de los estudiantes recomendaría el recurso, con lo cual indica que es valorado positivamente.

ii) Dimensión pedagógica y técnica

Las tablas de resultados correspondientes a cada una de las proposiciones se encuentran en el Apéndice D.

También en el Apéndice D se ilustran los resultados gráficamente.

A continuación se presentan las conclusiones a partir del análisis de las tablas:

- El 90% de los estudiantes está totalmente de acuerdo en que el objetivo indica lo que se

espera que sea aprendido.

- El 65% de los estudiantes coincide en que el nivel de dificultad es adecuado a sus conocimientos previos. El 30% es indiferente: no puede analizar el grado de dificultad o no le interesa analizarlo.
- El 80% de los estudiantes consideró que el material teórico del OA le ayudó a comprender los conceptos.
- El 90% de los estudiantes afirma que las actividades planteadas han sido claras y significativas para su propio aprendizaje.
- El 75% de los estudiantes manifestó que el recurso no informa sobre su progreso. En este caso no hay una feedback (retroalimentación) dentro del OA, sino por fuera de él a través de los intercambios que pueden producirse entre el profesor y sus propios compañeros en las actividades grupales y de pares. Es necesario incorporar mayor retroalimentación dentro del OA.
- El 65% de los estudiantes coincide en que las pistas sobre los errores cometidos son útiles. El 35% es indiferente. Analizando el sentido de esta proposición desde el punto de vista del estudiante, podemos pensar que ellos lo asociaron con las correcciones particulares y no con las recomendaciones sobre los errores posibles presentados en el OA. Se debería replantear esta proposición y reescribirla para aclarar su intención, o desdoblarla en dos proposiciones (recomendaciones sobre el error y corrección del error).
- El 80% de los estudiantes afirmó que el texto es conciso y preciso.
- El 85% de los estudiantes considera que los títulos indican claramente la acción a realizar.
- El 95% de los estudiantes manifestó que las imágenes empleadas aclaran los contenidos presentados. Es decir, la imagen tiene un valor añadido.

- Todos los estudiantes coincidieron en que claramente sabían en que parte del OA se encontraban en cada momento. Lo que indica que el diseño de la interfaz y las metáforas empleadas son adecuadas.
- El 95% de los estudiantes consideró muy positiva la incorporación de los videos. Afirman que los videos aclaran los contenidos presentados.
- Todos están totalmente de acuerdo en que la información está bien organizada.
- El 65% de los estudiantes está de acuerdo con los colores y el diseño. Para el 35%, el diseño le resulta indiferente.

c-2) Grabación de uso:

El procedimiento consistió en capturar los registros de la interacción del estudiante con el OA que se encuentran almacenados en el curso de Computación de la Plataforma Educativa Moodle.

c-2.1) Resultados y análisis

Como se describió anteriormente los 33 estudiantes se dividieron en dos grupos:

- Grupo A: sólo interacción con el OA (N= 13)
- Grupo B: interacción y valoración del OA a través del cuestionario (N = 20)

Los registros capturados fueron agrupados en las siguientes categorías (divididas en variables) para su posterior análisis:

- Estructura del OA : Árbol de contenidos (AC)
- Objetivos: Objetivos del OA (OBJ), Objetivos de contenidos teóricos (CON), Objetivos de las actividades (OAC)
- Contenidos conceptuales: Pasaje de Parámetros (PP), Pasaje de Parámetros por valor (PPV), Pasaje de Parámetros por referencia (PPR)
- Videos : Pasaje de Parámetros por valor con variables simples (V1), Pasaje de Parámetros por valor con variables estructuradas (V2), Pasaje de Parámetros por referencia con variables simples (V3), Pasaje de Parámetros por referencia con variables estructuradas (V4)
- Ejemplos: Problemas Resueltos (PR)

- Actividades : Actividad Grupal (AG1), Actividad de Pares (AG2)

La categoría Ejemplos: “Problemas Resueltos (PR)” tiene como función la descarga del archivo con problemas resueltos. Dado que no hay interacción, los valores obtenidos indican el tiempo que llevó descargarlo solamente que depende proveedor de Internet del usuario, no teniendo pues sentido analizar estos valores.

Con respecto a la categoría Actividades, como se explicó anteriormente, comprende los enunciados de los problemas que se resuelven fuera del OA, con lo cual no tiene sentido analizar estos valores. Esta categoría podría analizarse en otro estudio a posteriori a partir de los resultados de la interacción fuera del OA.

La tabla con los registros capturados y agrupados en las categorías mencionadas se encuentra en el Apéndice D. Los histogramas correspondientes a la distribución de cada una de las componentes de cada categoría se muestran en el Apéndice D.

Se calcularon los estadísticos para cada uno de las componentes (medido en segundos) de cada categoría. Los resultados correspondientes al Grupo A se muestran en las Tablas 8.10 y 8.11.

N=13		ESTRUCTURA DEL OA	OBJETIVOS			CONTENIDOS CONCEPTUALES		
		AC	OBJ	CON	OAC	PP	PPV	PPR
Estadísticos		5.80	10.78	6.54	3.50	6.49	3.16	94.78
Media		5.29	4.09	2.04	1.11	3.15	2.22	1.15
Mediana		4.01	15.58	13.92	6.00	8.18	2.09	331.99
Desv. tít.		69.26%	144.54%	212.77%	171.59%	126.04%	65.94%	350.26%
Mínimo		0.98	0.09	0.89	0.00	0.91	0.92	0.73
Máximo		14.06	56.90	52.41	22.10	22.42	7.97	1199.62
Percentiles	25	2.78	1.27	1.36	0.49	1.07	1.74	0.92
	50	5.29	4.09	2.04	1.11	3.15	2.22	1.15
	75	7.52	16.63	4.43	4.28	11.56	4.88	3.88

Tabla 8.10. Estadísticos del Grupo A - Parte 1.

N=13		VIDEOS			
		V1	V2	V3	V4
Estadísticos		22.66	18.88	6.19	1.39
Media		1.26	0.83	1.26	0.54
Mediana		68.51	39.87	16.89	1.88
Desv. tít.		302.33%	211.12%	273.00%	135.62%
Mínimo		0.65	0.00	0.00	0.00
Máximo		249.69	110.34	62.15	7.02
Percentiles	25	0.79	0.35	0.25	0.32
	50	1.26	0.83	1.26	0.54
	75	3.72	11.85	3.34	2.02

Tabla 8.11. Estadísticos del Grupo A - Parte 2.

Los resultados correspondientes al Grupo B se muestran en las Tablas 8.12 y 8.13.

N=20		ESTRUCTURA DEL OA	OBJETIVOS			CONTENIDOS CONCEPTUALES		
Estadísticos		INICIO	OBJ	CON	OAC	PP	PPV	PPR
Media		89.01	14.93	4.24	3.04	162.33	40.65	42.96
Mediana		5.14	1.88	2.83	1.84	14.40	18.64	3.35
Desv. tıp.		304.23	33.42	4.36	3.08	457.37	68.13	94.94
CV		341.79%	223.90%	103.02%	101.17%	281.76%	167.63%	221.00%
Mínimo		0.00	0.00	0.00	0.00	0.00	0.00	0.00
Máximo		1350.22	134.71	14.24	10.09	1970.33	292.21	367.00
Percentiles	25	0.08	0.02	0.79	1.03	0.94	0.95	1.00
	50	5.14	1.88	2.83	1.84	14.40	18.64	3.35
	75	11.38	10.35	7.04	4.36	34.89	57.23	36.31

Tabla 8.12. Estadísticos del Grupo B - Parte 1.

N=20		VIDEOS			
Estadísticos		V1	V2	V3	V4
Media		236.19	166.91	72.55	113.38
Mediana		211.05	129.72	59.37	3.58
Desv. tıp.		284.28	202.90	74.67	153.29
CV		120.36%	121.56%	102.93%	135.20%
Mínimo		0.18	0.06	0.00	0.00
Máximo		1146.51	687.18	206.84	559.81
Percentiles	25	7.32	0.75	0.66	1.15
	50	211.05	129.72	59.37	3.58
	75	317.85	265.43	139.42	237.25

Tabla 8.13. Estadísticos del Grupo B - Parte 2.

i) Análisis de resultados del Grupo A

Los tiempos utilizados registraron variabilidad, indicando que cada estudiante siguió su propio ritmo. La media no puede considerarse un valor representativo, se analizaron entonces otras Medidas de Posición no centrales como los cuartiles.

- Estructura del OA : Árbol de contenidos (AC)

El tiempo utilizado registra variabilidad (CV= 69.26%). El 75% de los estudiantes utilizó 7.52 segundos o menos para reconocer el árbol, lo que indica su simplicidad.

- Objetivos: A pesar de la variabilidad registrada, de los tres objetivos, en el que

más se detuvieron fue en el “Objetivo General del OA”, que aunque consta de una sola línea (Ver Figura 8.33), despertó más interés. El 50% de los estudiantes utilizó 4.09 segundos o menos y el 75%, 16.63 o menos.

- **Contenidos Conceptuales:** Aunque la variabilidad registrada es muy grande, de los tres contenidos, en el que más tiempo se detuvieron fue en “Pasaje de Parámetros” (PP). El 50% de los estudiantes utilizó 3.15 segundos o más y el 75% 11.56 o menos.

- **Videos :**
 - Video 1: El 50% utilizó 1.26 segundos o más, lo que indica que no lo vieron completo ya que el video dura 75 segundos. El 75% utilizó 3.72 segundos o menos, es decir no lo vieron completo.
 - Video 2: El 50% empleó 0.83 segundos o más, lo que indica que el 50% no lo vio completo, ya que este video dura 93 segundos. El 75% utilizó 11.85 segundos o menos, indicando que este porcentaje de estudiantes no vió el video completo.
 - e) Video 3: El 50% utilizó 1.26 segundos o más, lo que indica que no lo terminaron de ver completo, ya que el video dura 85 segundos. El 75% utilizó 3.34 segundos o menos, indicando que este porcentaje de estudiantes no vio el video completo. Más aún, ninguno de los estudiantes de este grupo lo vio completo pues el valor máximo es 62.15 segundos. Este video es similar al video 1, pero la diferencia es que resuelve el problema Swap al utilizar pasaje de parámetros por referencia.
 - Video 4: El 50% empleó 1.26 segundos o más, lo que indica que la mitad de los estudiantes no lo vio completo, ya que el video dura 120 segundos. El 75% utilizó 2.02 segundos o menos, indicando que este porcentaje de estudiantes no vio el video completo. Más aún, ninguno de los estudiantes de este grupo lo vio completo pues el valor máximo es 2.02 segundos.

i) Análisis de resultados del Grupo B

Los tiempos utilizados registraron mucha variabilidad, indicando que cada estudiante

siguió su propio ritmo. La media no puede considerarse un valor representativo, se analizaron entonces otras Medidas de Posición no centrales como los cuartiles.

- Estructura del OA : Árbol de contenidos (AC)
El tiempo utilizado registra mucha variabilidad (CV= 341.79%). El 75% de los estudiantes utilizó 11.38 segundos o menos para reconocer el árbol, lo que indica su simplicidad.
- Objetivos: A pesar de la variabilidad registrada, de los tres objetivos, en el que más se detuvieron fue en el “Objetivo General del OA”, que aunque consta de una sola línea (Ver Figura buscar la de los objetivos), despertó más interés. El 50% de los estudiantes utilizó 1,88 segundos o menos y el 75%, 10.35 o menos.
- Contenidos Conceptuales: Aunque la variabilidad registrada es muy grande, de los tres contenidos, en el que más tiempo se detuvieron fue en “Pasaje de Parámetros por valor” (PPV). El 50% de los estudiantes utilizó 18.64 segundos o más y el 75% 57.23 o menos.
- Videos :
 - Video 1: El 50% utilizó 211.05 segundos o más, lo que indica que se detuvieron más del triple del tiempo que dura que es de 75 segundos. El 75% utilizó 317.85 segundos o menos, indicando que por lo menos cuadruplicaron el tiempo de duración del video.
 - Video 2: El 50% empleó 129.72 segundos o más, lo que indica que no se detuvieron tanto tiempo como en el Video 1, ya que este video dura 93 segundos. El 75% utilizó 265.43 segundos o menos, indicando que por lo menos casi triplicaron el tiempo de duración del video.
 - f) Video 3: El 50% utilizó 59.37 segundos o más, lo que indica que no lo terminaron de ver completo, ya que el video dura 85 segundos. El 75% utilizó 139.42 segundos o menos, indicando que por lo menos casi duplicaron el tiempo de duración del video. Este video es similar al video 1, pero la diferencia es que resuelve el problema Swap al utilizar pasaje de parámetros por referencia.
 - Video 4: El 50% empleó 3.58 segundos o más, lo que indica que la mitad

de los estudiantes no lo vio completo, ya que el video dura 120 segundos. Sin embargo del resto un 25% se detuvo 237.25 o más, que casi duplica la duración del video.

iii) Comparación de grupos

Para determinar cuáles son las componentes del OA que permiten establecer las diferencias entre los estudiantes que no respondieron el cuestionario (Grupo A) y los que lo respondieron (Grupo B) se realizaron pruebas de diferencias de medias.

Previo a ello, se aplicó la prueba de normalidad de Kolmogorov-Smirnov para definir si se optaba por una prueba paramétrica o no-paramétrica. Los resultados se presentan en el Apéndice D. Las variables AC, OBJ, CON, OAC, PP, PPR ($p > 0.05$), mientras que las variables PPV, V1, V2, V3, V4 no ($p < 0.05$).

Para las variables con distribución normal se le realizó una prueba T-Student para muestras independientes y para el resto una Prueba de Mann-Whitney para muestras independientes. No se encontraron diferencias significativas ($p > 0.05$) entre el Grupo A y el Grupo B para las variables AC, OBJ, CON, OAC, PP, PPR y PPV. Es decir no hay diferencia entre los estudiantes que respondieron el cuestionario y los que no lo respondieron con respecto al recorrido realizado en: Árbol de contenidos (AC), Objetivos del OA (OBJ), Objetivos de contenidos teóricos (CON), Objetivos de las actividades (OAC), Contenidos conceptuales: referidos al Pasaje de Parámetros (PP), Pasaje de Parámetros por valor (PPV) y Pasaje de Parámetros por referencia (PPR).

Se encontraron diferencias significativas ($p < 0.05$) entre el Grupo A y el Grupo B para las variables V1, V2, V3, V4. Hay diferencia entre los estudiantes que respondieron el cuestionario y los que no lo respondieron en el recorrido de los videos.

Las tablas correspondientes se encuentran en el Apéndice D.

Concluimos que las diferencias entre ambos grupos se encuentran en el recorrido de los videos.

c-3) Entrevistas en profundidad

Resumimos a continuación algunos de los comentarios realizados por los estudiantes al terminar de responder el cuestionario:

- “Me ayudó a aclararlas ideas y me gustaría que agreguen más material de este tipo para los temas siguientes”
- “La información brindada me parece de gran ayuda para aclarar dudas ya que al ver

gráficamente los pasos de los programas nos hace sentirnos más cómodos cuando escribimos y ejecutamos programas en ella computadora”

- “Es necesario que exista material interactivo y que se tenga en cuenta nuestra opinión al respecto. El material sirve para comprender las dudas de la teoría”
- “Me parece ameno el material, fue dinámico y permite la incorporación de conceptos. La utilización de videos facilita ver la ejecución del programa en forma clara y ordenada”
- “En mi opinión, la existencia del cuestionario ayuda a mejorar el material, y más cuando se trata de un recurso interactivo. La presencia del material beneficia al aprendizaje ya que ayuda a reforzar y visualizar los conceptos vistos en clase”
- “Debería haber en el curso más videos de este tipo sobre como hacer programas”
- “Los videos ayudan mucho a comprender el funcionamiento y orden de los procedimientos”
- “El material está bueno para personas que tienen un conocimiento ya adquirido. Sería importante que cuando se hace referencia a temas anteriores haya un hipervínculo que te lleve a un resumen de ese tema para recordarlo”
- “Me hubiese gustado que haya algunos ejemplos más difíciles, similares a los que toman en los parciales”
- “Me pareció muy útil la información extra al módulo teórico de la cátedra, muy interactivo y de fácil interpretación, reforzando así los conocimientos previos. Por lo tanto sugiero que para los siguientes temas se incorporen material de este tipo”

Luego del análisis de los resultados obtenidos a partir del cuestionario y los registros de logs, se entrevistó a cinco estudiantes del Grupo B, y se envió un correo electrónico a los estudiantes del Grupo A a los efectos de contrastar, comprender y describir la información recopilada mediante un análisis retrospectivo de la acción.

Los estudiantes que no tuvieron la obligación de contestar el cuestionario (Grupo A) se detuvieron más en la parte conceptual y los videos los desestimaron. Resolvieron los problemas planteados en las Actividades y realizaron consultas por correo electrónico acerca de la resolución de los mismos. Concibieron el OA como un recurso de aprendizaje y no se detuvieron en su evaluación.

En cambio los estudiantes que debían responder el cuestionario (Grupo B) al verse en la responsabilidad de evaluar el OA, lo siguieron con más detenimiento, sobre todo

analizaron con profundidad los videos.

d) En producción

Luego de realizar las modificaciones necesarias a partir de los resultados de la evaluación por estudiantes se procederá a la puesta en práctica del OA en el curso de Computación de la Plataforma Educativa Moodle. Estará disponible para todos los estudiantes de manera que puedan interactuar con el OA. Esto implicará además que el mismo sufra cambios a lo largo de su vida útil. Estos cambios podrán producirse por errores no detectados en las etapas anteriores, por que se produzcan cambios en algunas de sus componentes o se requieran modificaciones funcionales no contempladas anteriormente.

9. CONCLUSIONES Y LÍNEAS FUTURAS DE INVESTIGACIÓN Y DESARROLLO

En este capítulo se presentan las conclusiones de esta Tesis y las líneas futuras de investigación y desarrollo.

9.1. Conclusiones

Sobre la base del objetivo general de esta tesis “Formular una metodología transferible de desarrollo y evaluación de la calidad de Objetos de Aprendizaje (OA)” se indican a continuación los resultados más significativos derivados del trabajo de investigación realizado:

- Se ha presentado un Modelo de proceso para el desarrollo de Objetos de Aprendizaje (OA), el MPOBA, que aporta una serie de particularidades para este tipo de software educativo.

En el Modelo MPOBA se integran las actividades del Diseño Centrado en el Usuario (DCU) con la metodología de desarrollo de software de sistemas interactivos.

Consideramos que la incorporación de los principios del DCU en la construcción de los OA con las adaptaciones pertinentes permitirá relevar elementos significativos en el diseño de estos objetos proporcionando a los estudiantes mejores experiencias a partir de un mayor grado de implicación de todos los actores de los procesos de enseñanza y de aprendizaje.

- La propuesta de integración de Ingeniería del Software (IS) con la Ingeniería de la Usabilidad (IU) del Modelo MPOBA se encuentra en línea con las investigaciones internacionales actuales. En particular se han caracterizado y seleccionado las técnicas pertenecientes al campo de la Interacción Persona-Ordenador (IPO) según criterios relevantes para la integración desde una perspectiva de IS.
- El Modelo MPOBA, proporciona una diversidad de actividades en cada una de sus fases iterativas adaptables a diferentes OA y contextos educativos. Su generalidad y

flexibilidad se combina con la sistematicidad de los pasos a seguir claramente identificados.

- Se propone una metodología sistemática para evaluar un OA en todos los momentos del proceso de desarrollo a través de diferentes actividades, criterios, instrumentos y técnicas, indicando además quiénes deben participar en cada tarea. Se garantiza de esta manera una de los principios del DCU: realizar evaluaciones de prototipos desde el principio del desarrollo, en forma continua.
- La perspectiva de calidad de un OA abordada rescata tanto, criterios de usabilidad de sitios web y criterios pedagógicos, con un tratamiento específico. Esta posición no dista de otras existentes, pero se diferencia en su tratamiento.
A través de estos criterios es posible conocer la valoración de expertos, docentes y estudiantes mediante los distintos indicadores propuestos en varios momentos específicos del desarrollo del OA. Los resultados obtenidos permiten realizar análisis tanto cuantitativos y cualitativos acerca de la calidad del OA que retroalimentan el proceso de construcción del mismo.
- Los instrumentos de evaluación de las propuestas existentes considerar ciertos criterios generales desde el punto de vista técnico y pedagógico. En esta tesis se identifican y justifican las dimensiones críticas a evaluar. En este sentido, se plantean los aspectos concretos a tener en cuenta a partir de los criterios definidos.
- Los resultados obtenidos a partir de las diversas evaluaciones, rescatan además valoraciones que se constituyen en un aporte valioso para la documentación incluida en los OA a través de sus metadatos. Esta información aporta datos importantes en cuanto a la calidad e idoneidad del OA y contribuyen a la decisión de su incorporación en los cursos de un Entorno virtual de enseñanza y aprendizaje (EVEA) favoreciendo de esta manera su reutilización.
- La definición adoptada sobre el concepto de OA y la descripción de los elementos que lo componen aportan las primeras orientaciones para el diseño instruccional del OA. Se caracterizan los estándares y especificaciones que facilitan su interoperabilidad y reutilización. Posteriormente en el propio desarrollo del Modelo

MPOBA, se presentan diversas estrategias didácticas y recomendaciones específicas para el diseño instruccional.

- El trabajo de campo se concretó en un contexto real de enseñanza. La aplicación del Modelo MPOBA ilustra y permite analizar la validez de cada una de las fases del mismo, las técnicas y los instrumentos. Se proporcionan de esta manera más elementos que permiten transferir esta metodología en la creación de OA en otros contextos educativos. Como valor agregado, el trabajo con los docentes promovió una reflexión acerca de sus prácticas, revelando nuevos recorridos del currículum. Se destaca además la participación de los estudiantes cuyos aportes fueron muy significativos para este trabajo de investigación.

9.2. Futuras líneas de investigación y desarrollo

- Investigar la aplicación del modelo MPOBA en otras áreas del conocimiento, para determinar los parámetros que se deben ajustar según el contexto
- Incorporar en forma dinámica información propia del estudiante, de modo de adaptar (con técnicas de minería de datos e Inteligencia Artificial) aspectos de los OA generados con la metodología MPOBA.

Además de las líneas de investigación, se presentan otras acciones futuras derivadas de la investigación realizada en esta tesis:

- Proveer de un sistema de ayuda al docente para cada OA respecto a los inconvenientes detectados en los estudiantes en esa etapa de su aprendizaje.
- Proporcionar capacitación a los equipos de desarrollo en la aplicación del Modelo MPOBA.
- Generar un material instruccional que oriente a los equipos de desarrollo en la aplicación del modelo MPOBA.

CAPITULO 10. REFERENCIAS

- Abascal, J., Aedo, I., Cañas, J., Gea, M., Gil, A. B., Lorés, J., Martínez, A. B., Ortega, M., Valero, P. y Vélez, M. (2006). *La interacción persona-ordenador*. de. Lorés J. Libro electrónico editado por AIPO. Disponible en <http://griho.udl.es/ipo/>). Recuperado el 12 de febrero de 2012.
- ABET (2011). *Criteria for Accrediting Engineering Programs. Engineering Accreditation Commission. Effective for Reviews During the 2012-2013 Accreditation Cycle. Disponible en http://www.abet.org/uploadedFiles/Accreditation/Accreditation_Process/Accreditation_Documents/Current/eac-criteria-2012-2013.pdf*. Recuperado el 10 de mayo de 2012.
- Abud, F. (2009). MeISE: Metodología de Ingeniería de Software Educativo. *Revista Internacional de Educación en Ingeniería*, 2 (1).
- Aceves, L., Argüello, J. , del Río, S.. (2009). *Caso de estudio sobre la usabilidad en periódicos en línea en México*. Disponible en <http://www.slideshare.net/RevistaSG/caso-de-estudio-sobre-la-usabilidad-en-peridicos-en-lnea-en-mxico>. Recuperado el 25 de febrero de 2012.
- Adell, J, Castellet, J. M. y Pascual, J. (2004). *Selección de un entorno virtual de enseñanza/aprendizaje de código fuente abierto para la Universitat Jaume I. Centro de Educación y Nuevas Tecnologías de la UJI*. Disponible en http://cent.uji.es/doc/eveauji_es.pdf . Recuperado el 14 de mayo de 2011.
- ADL (2011). <http://www.adlnet.gov/capabilities/scorm#tab-main>
- ADL. (2002). *Emerging and Enabling Technologies for the design of Learning Object Repositories Report. Advanced Distributed Learning Initiative*. Disponible en <http://xml.coverpages.org/ADLRepositoryTIR.pdf>. Recuperado el 15 de marzo de 2011.
- Albion, P. R. (1999). Heuristic evaluation of educational multimedia: from theory to practice. In *Proceeding of 16th Annual conference of the Australasian Society for Computers in Learning in Tertiary Education, ASCILITE (1999)*. Disponible en <http://www.usq.edu.au/users/albion/papees/ascilite99.html>. Recuperado el 20 de mayo de 2012
- Alva, M.(2005). *Metodología de Medición y Evaluación de la Usabilidad en Sitios Web Educativos*, Tesis Doctoral, Universidad de Oviedo, 2005.
- Amó, A. y Martínez, N. (2005). *Introducción a la Ingeniería del software*. Delta Publicaciones.

- Anido Riffon, L., Rodríguez Artacho, M. (2002). *Traducción del estándar LOM*. Disponible en <http://www.gist.uvigo.es/~lanido/LOMes/>. Recuperado el 18 de julio de 2011.
- APROA. (2005). *Aprendiendo con Repositorio de Objetos de Aprendizaje*. APROA Comunidad – FAQ: Sobre Objetos de Aprendizaje Disponible en http://146.83.43.182/aproa/1116/article-68380.html#h2_1. Recuperado el 15 de junio de 2010.
- Ausubel, D. (1983). *Psicología educativa: Un punto de vista cognoscitivo*. Editorial Trillas. México.
- Aycart, D., Ginestá, M. y Hernández, M. (2007). *Ingeniería de Software en Entornos de SL*. Barcelona: Universitat Oberta de Catalunya..
- Bangor, A., Kortum, F. & Miller, T. (2008): An Empirical Evaluation of the System Usability Scale, International. *Journal of Human-Computer Interaction*, 24, (6), 574-594. Disponible en <http://dx.doi.org/10.1080/10447310802205776>. Recuperado el 2 de junio de 2012.
- Barajas, A., Muñoz, J. y Álvarez, F. (2007). Modelo Instruccional para el Diseño de Objetos de aprendizaje: Modelo MIDOA. *Virtual Educa 2007*. São José dos Campos - São Paulo, Brazil. Disponible en <http://videoseducativos.org/comtics/164-ABS.pdf>. Recuperado el 10 de mayo de 2011.
- Barker, P. (2005). *What is IEEE Learning Object Metadata / IMS Learning Resource Metadata?* CETIS standards briefings series, the Centre for Educational Technology Interoperability Standards. Disponible en <http://metadata.cetis.ac.uk/guides/WhatIsLOM.pdf>. Recuperado el 18 de julio de 2011.
- Bartek, V. & Cheatham, D. (2003). *Experience remote usability testing, Part 1: Examine study results on the benefits and downside of remote usability testing*. Disponible en <http://www.ibm.com/developerworks/library/wa-rmustr1/>.
- Becta Quality (2007). Principles for digital learning resources. Summary Information. Disponible en http://archive.teachfind.com/becta/industry.becta.org.uk/content_files/industry/resources/Articles/Learning_resource_development/quality_principles.doc. Recuperado el 26 de enero de 2012.
- Bednar, A.K., Cunningham, D., Duffy, T.M., Perry, J.P. (1995). Theory into practice: How do we link? In G.J. Anglin (Ed.), *Instructional technology: Past, present and future*. Englewood, CO: Libraries Unlimited, Inc.
- Berggren, A., Burgos, D., Fontana, J.M., Hinkelman, D., Hung, V., Hursh, A. & G. Tielemans. (2005). Practical and Pedagogical Issues for Teacher Adoption of IMS Learning Design Standards in Moodle LMS. *Journal of Interactive Media (Special issue on Learning Design)*. Disponible en <http://jime.open.ac.uk/2005/2>. Recuperado el 21 de agosto de 2011.
- Bevan N (1999). Design for Usability. In *Proceedings of HCI International 1999*.

- Disponibile en <http://www.nigelbevan.com/papers/desuse.pdf>. Recuperado el 12 de setiembre de 2012.
- Bevan, N. (1995). Measuring usability as quality of use. *Software Quality journal*, 4, 115-150. NPL Usability Services. National Physical Laboratory, Teddington – Middx. TW11 0LW. Disponible en <http://www.nigelbevan.com/papers/qusab95.pdf>. Recuperado el 8 de junio de 2012.
- Bevan, N. & Macleod, M.(1994).Usability measurement in context. *National physical laboratory, Behaviour and information technology*, 13, 132-145. Teddington, Middlesex, UK. Disponible en www.nigelbevan.com/papers/music94.pdf. Recuperado el 8 de junio de 2012.
- Bevan, N., & Curson, I. (1999). Planning and implementing user-centred design. In *CHI'99 Extended Abstracts on Human Factors in Computing Systems*, 137-138. New York: ACM Press. Disponible en <http://www.nigelbevan.com/papers/ucdtut97.pdf>. Recuperado el 20 de octubre de 2011.
- Bhola, H. S. (1990). *Evaluating "Literacy for development" projects, programs and campaigns:Evaluation planning, design and implementation, and utilization of evaluation results*. UNESCO Institute for Education German Foundation for International . Hamburg, Germany, Bonn, Germany. Disponible en <http://unesdoc.unesco.org/images/0008/000877/087728eo.pdf>. Recuperado el 15 de abril de 2012.
- Bias, R. (1994). The pluralistic walkthrough: Coordinated empathies. In *J. Nielsen & R. Mack (Eds.), Usability inspection methods*, 63–76, New York. Wiley .
- Bias, R. G. & Mayhew, D. J. (2005). *Cost-Justifying Usability: An Update for the Internet Age*. San Francisco: Morgan Kaufmann Publishers.
- Black,P. & Wiliam, D. (1998) Assessment and Classroom Learning. *Assessment in Education* 5, (1). Disponible en http://www.ericfacility.net/databases/ERIC_Digests/ed470206.html. Recuperado el 15 de mayo de 2011.
- Blackmon, M. H. (2004). Cognitive Walkthrough. In *W. S. Bainbridge (Ed.), Encyclopedia of Human- Computer Interaction*, 2 ,(1), 104–107. Great Barrington, MA: Berkshire Publishing Group. Disponible en <http://www.colorado.edu/Papers/CognitiveWalkA0290000.doc>. Recuperado el 10 de febrero de 2012.
- Blecken, A., & Marx, W. (2010). Usability Evaluation of a Learning Management System. In *Proceedings of the 43rd Hawaii International Conference on system Sciences*, 1-9. Disponible en <http://doi.ieeecomputersociety.org/10.1109/HICSS.2010.422>. Recuperado el 2 de junio de 2012.
- Bojko, A. & Schumacher, R. (2008) Eye Tracking and Usability Testing in Form Layout Evaluation. In *Proceedings of the 38th International Symposium of Business Forms Management Association (BFMA) 2008*. Disponible en www.bfma.org/resource/resmgr/Articles/08_32.pdf. Recuperado el 19 de febrero de 2012.

- Boneu, J.M. (2007). Plataformas abiertas de e-learning para el soporte de contenidos educativos abiertos. En: *Contenidos educativos en abierto (monográfico en línea)*. *Revista de Universidad y Sociedad del Conocimiento (RUSC)*, 4, (1). Disponible en <http://www.uoc.edu/rusc/4/1/dt/esp/boneu.pdf>. Recuperado el 20 de mayo de 2011.
- Booch, G. (1991). *Object Oriented Design with Applications*, The Benjamin/Cumming Publishing Company, Inc., Redwood City.
- Borges de Barros Pereira, H. (2002). *Análisis experimental de los criterios de evaluación de usabilidad de aplicaciones multimedia en entornos de educación y formación a distancia*. Universitat Politecnica de Catalunya, Barcelona. Disponible en <http://tdx.cat/handle/10803/6542> .Recuperado el 25 de octubre de 2011.
- Borrero Caldas, M.C., Cruz García, E., Mayorga Muriel, S. y Ramírez González, K. (2010). *Una metodología para el diseño de objetos de aprendizaje. La experiencia de la DINTEV de la Universidad del Valle*. Disponible en http://objetos.univalle.edu.co/files/articulo_AMED.pdf. Recuperado el 10 de mayo de 2011.
- Bou Bauzá, G. (2003). *El guión multimedia*. España: Anaya Multimedia.
- Bradley, C., Haynes, R., Cook, J., Boyle, T., & Smith, C. (2009). Design and development of multimedia learning objects for mobile phones. In M. Ally (Ed.), *Mobile learning: Transforming the delivery of education and training* (pp. 157–182).Edmonton, AB: Athabasca University Press.
- Brooke, J. (1996). SUS: A Quick and Dirty Usability Scale. In: P.W. Jordan, B. Thomas, B.A. Weerdmeester & I.L. McClelland (Eds.), *Usability Evaluation in Industry*. London: Taylor & Francis. Disponible en [www.itu.dk /courses/U/E2005/litteratur/sus.pdf](http://www.itu.dk/courses/U/E2005/litteratur/sus.pdf). Recuperado el 8 de junio de 2012
- Brooks, F. (1995). *The Mythical Man-Month: Essays on Software Engineering*. Anniversary (Ed.) Reading, MA: MIT Press
- Bruner, J. S. (1988). *Desarrollo Cognitivo y Educación. España*. Ediciones Morata.
- Cabero, J. (2004). La red como instrumento de formación. Bases para el diseño de materiales didácticos. *Píxel-Bit. Revista de Medios y educación*, 22, 5-23. Disponible en <http://www.sav.us.es/pixelbit/articulos/n22/n22art/art2201.htm>. Recuperado el 25 de abril de 2012.
- Calero, C., Moraga, M. y Piatini, M. (2010). *Calidad del producto y proceso de software*. Editorial Ra-Ma.
- Callaos, N. and Callaos, B. (1993). Designing with Systemic Total Quality. In Callaos, N & Sánchez, B. (eds.), *International Conference on Information Systems*. (pp. 548-560). International Institute of Informatics and Systemics, Orlando.
- Canabal, M., Sarasa, A., & Sacristán, J. C. (2008). *LOM-ES: Un perfil de aplicación de LOM*. V Simposio Pluridisciplinar sobre Diseño y Evaluación de Contenidos

- Educativos Reutilizables SPDECE (2008). . Disponible en http://www.web.upsa.es/spdece08/contribuciones/110_Spedece2008-lom-es.pdf. Recuperado el 17 de julio de 2011.
- CANAIRE (2002). *A Report on Learning Object Repositories, Review and Recommendations for a Pan-Canadian Approach to Repository Implementation in Canada*. Disponible en http://www.colombiaaprende.edu.co/html/mediateca/1607/articles-172726_archivo.pdf. Recuperado el 20 de mayo de 2011.
- Cañas, J.J., Antolí, A., & Quesada, J.F. (2001). The role of working memory on measuring mental models of physical systems. *Psicologica*, 22, 25-42. Disponible en <http://www.ugr.es/Eergocogn/articulos/mentamodelsandworkingmemory.pdf>. Recuperado el 13 de febrero de 2012.
- Cañas, J.J., Granollers, A., Lorés Vidal, J. (2005). *Diseño de sistemas interactivos centrados en el usuario*. Editorial UOC, S.L. Disponible en <http://www.grihohcitoools.udl.cat/mpuia/>
- Carroll, J. (1995). Scenario-Based Design: Envisioning Work and Technology .In Carroll, J. (Ed), *Introduction: The Scenario Perspective on System Development*, John System Development.Wiley & Sons, New York.
- Carroll, J. (2000). *Making use: Scenario-based design of human-computer interactions*. MIT Press.
- Casanova, E. M. (2010). *Vídeo digital y alfabetización audiovisual en la formación universitaria*. Tesis Doctoral. Universidad de Barcelona.
- Cassola, E. (2004). Elaboración de Material Educativo para la Formación de Profesionales en Desarrollo de Software. *Memorias Congreso Iberoamericano de Educación Superior en Computación (CIESC), Conferencia Latinoamericana de Informática (CLEI2004)*. Perú.
- Cataldi, Z. (2006). El ciclo de vida y la matriz de actividades como base para el diseño y desarrollo metodológico de software educativo. *Revista Ingeniería Informática*, 13. Disponible en <http://www.inf.udec.cl/revista>. Recuperado el 1 de noviembre de 2011.
- Cataldi, Z., Lage, F., Pessacq, R. y García Martínez, R. (2003). Metodología extendida para la creación de software educativo desde una visión integradora. *Revista Latinoamericana de Tecnología Educativa, RELATEC, Universidad de Extremadura*, 2, (1). Disponible en http://dialnet.unirioja.es/servlet/fichero_articulo?codigo=1252631. Recuperado el 25 de octubre de 2011.
- Cataldi, Zulma. (2000). *Metodología de diseño, desarrollo y evaluación de software educativo*. Tesis de Maestría. Universidad Nacional de la Plata. Disponible en <http://www.fi.uba.ar/laboratorios/lsi/cataldi-tesisdemagistereninformatica.pdf>. Recuperado el 20 de octubre de 2011.
- Chan, M. E. (2002). *Objetos de aprendizaje: una herramienta para la innovación educativa*.

- Revista Apertura Innova*. 2, (3).Universidad de Guadalajara.
- Chan, M.E. (2002). Objetos de aprendizaje: Una herramienta para la innovación educativa. *Apertura No. 2*, Centro de Recursos de Innovación Educativa.
- Chiappe Laverde, A. (2009). Acerca de lo pedagógico en los objetos de aprendizaje-reflexiones conceptuales hacia la construcción de su estructura teórica. *Estudios Pedagógicos XXXV*, 1.
- Chin, J. P., Diehl, V. A. and Norman, K. L. (1988). Development of an instrument measuring user satisfaction of the human-computer interface. *In Proceedings of ACM CHI '88*, Washington, DC, 213-218. Disponible en <http://dl.acm.org/citation.cfm?id=57203>_Recuperado el 19 de febrero de 2012.
- Chisnell, D., Redish, J. C., and Lee, A. (2006) New heuristics for understanding older adults as web users, *Technical Communication*, 53, (1), 39-59. Disponible en http://www.usabilityworks.net/resources/chisnell_redish_lee_heuristics.pdf . Recuperado el 13 de febrero de 2012.
- Churchill, D., & Hedberg, J. 2008. Learning Object Design Considerations for Small-Screen Handheld Devices. *Computers & Education*, 50, (3), 881-893. Disponible en [http://www.cbilt.soton.ac.uk/multimedia/PDFs/Learning object design considerations for small screen handheld devices.pdf](http://www.cbilt.soton.ac.uk/multimedia/PDFs/Learning_object_design_considerations_for_small_screen_handheld_devices.pdf) . Recuperado el 20 de octubre de 2011.
- Codina, Luís (2000). *El libro digital y la www*. Madrid:Tauro Pro.
- Coll, C., Martín, E., Mauri, T., Miras, M., Onrubia, J., Solé, I. & Zabala, A. (1999). *El constructivismo en el aula*. Barcelona: Editorial Graó.
- CONFEDI (2006). *Desarrollo de competencias en la enseñanza de la Ingeniería Argentina*. Documento del Consejo Federal de Decanos de Ingeniería. Argentina. Disponible en <http://www.confedi.org.ar/sites/files/privado/DESARROLLO%20DE%20COMPETENCIAS%20GENERICAS.pdf>. Recuperado el 21 de marzo de 2009.
- Constantine, L. (1994). Collaborative usability inspections for software. In *Software Development '94 Proceedings*. San Francisco: Miller Freeman. Disponible en <http://www.blacksunimages.com/usability/handout.htm>. Recuperado el 2 de junio de 2012.
- Conte, T., Vaz, V., Massolar, J., Mendes, E., Travassos, G. (2009). Improving a Web Usability Inspection Technique using Qualitative and Quantitative Data from an Observational Study. In *XXIII Simpósio Brasileiro de Engenharia de Software-SBES 2009* .Fortaleza, Brazil. <http://www.lbd.dcc.ufmg.br:8080/colecoes/sbes/2009/023.pdf>. Recuperado el 21 de febrero de 2012.
- Contessa, G. (2007). Representation, Interpretation, and Surrogate Reasoning. *Philosophy of Science*, 74, (1), 48-68. Disponible en http://philsci-archive.Pitt.Edu/3291/1/Paper_Representation_Interpretation_and_Surrogate_Reasoning_

- (PhilSci).pdf. Recuperado el 1 de noviembre de 2011.
- Costa, A. P., Loureiro, M. J., Reis, L. P., Sá, P., Guerra C. y Vieira, R. M. (2009). Courseware Sere: Technical and Didactic Evaluation. In *V International Conference on Multimedia and ICT in Education (m-ICTE2009)*. Lisboa. Disponible en http://www.ludomedia.pt/uploads/artigos/mICTE2009_Article_2009.pdf. Recuperado el 2 de noviembre de 2011.
- Cruz Meléndez, A., Alfaro Rivera, J. A., Ramírez Montoya, M. S. (2012). Objeto de aprendizaje abierto para la formación docente orientado a desarrollar competencias de pensamiento crítico con énfasis en habilidades cognitivas. *RIED. Revista Iberoamericana de Educación a Distancia*, 15, (1). Disponible en <http://www.utpl.edu.ec/ried/images/pdfs/vol15-1/objetodeaprendizaje.pdf>. Recuperado el 24 de febrero de 2012.
- Davenport, T. & Prusak, L. (1998). *Working Knowledge: How Organizations Manage What They Know*. Harvard Business School Press.
- De Graff, E. & Kolmos, A. (2007). *Management of Change. Implementation of Problem-Based and Projec-Based learning in Engineering*. Sense Publishers
- Del Rìo Lugo, N (1999).Bordando sobre la zona de desarrollo próximo. *Educar*, 9. Mexico. Disponible en http://cmap.upb.edu.co/rid=1186179924796_526735513_1416. Recuperado el 3 de febrero de 2012.
- Delval, J. (2001). Hoy todos son constructivistas. *Revista Educere*, 15.
- Desire Project Team. (2000). *2.1 Quality selection - DESIRE Information gateways handbook*. Disponible en <http://www.desire.org/handbook/2-1.html>. Recuperado el 26 de enero de 2012.
- Díaz Barriga, F. y Hernández, G. (2002). *Estrategias docentes para un aprendizaje significativo. Una interpretación constructivista* (2ª. ed.). México: McGraw Hill.
- Díaz, M. (2002). *Propuesta de una metodología de desarrollo y evaluación de software educativo bajo un enfoque de calidad sistémica*. Tesis de Especialista en Informática Educativa. Universidad Simón Bolívar. Caracas, Venezuela. Disponible en http://www.academia-interactiva.com/te-sis_evaluacion_software.pdf. Recuperado el 1 de noviembre de 2011.
- Díaz, M.G., Pérez, M.A., Grimán, A. y Mendoza, L. (2002). Instrumento de evaluación de software educativo bajo un enfoque sistémico. En Lamas M., Fernández M., Anido L. (Eds.) *Congreso Iberoamericano, 4to. Simposio Internacional, 7mo. Taller Internacional de Software Educativo*. Universidad de Vigo. Disponible en www.academia-interactiva.com/evaluacion.pdf. Recuperado el 2 de noviembre de 2011.
- Dijkstra, E. and Feijen, W. (1988). *A Method of Programming*. Addison-Wesley, Boston, USA.

- Dix A., Finlay, J., Abowd G., Beale R. (1998). *Human Computer Interaction*. Prentice Hall, Englewood Cliffs, NJ (2nd edition).
- Dix, A., Ramduny, D., Rayson P., Onditi, V., Sommerville, I. & Mackenzie, A. (2003). Finding Decisions Through Artefacts. *Human-Computer Interaction: theory and Practice (volume 1)*. Lawrence Erlbaum Associates. Disponible en www.alandix.com/academic/paper/FHCII2003-artefacts. Recuperado el 15 de febrero de 2012.
- Downes, S. (2004). *The Learning Marketplace. Meaning, Metadata and Content Syndication in the Learning Object Economy*. Moncton, New Brunswick. Disponible en <http://www.downes.ca/files/book3.htm>. Recuperado el 20 de mayo de 2011.
- Dray, S.& Siegel, D. (2004). Remote possibilities? International usability testing at a distance. *Interactions*, 11, (2), 10-17. ACM Press.
- Du Boulay. J. (1986). Some difficulties of learning to program. *Journal of Educational Computing Research*, 2, (1), 57-73 Disponible en <http://www.cogs.susx.ac.uk/users/bend/papers/diffsofprogramming.pdf>. Recuperado el 15 de octubre de 2011
- Durán, A. (2000). *Un Entorno Metodológico de Ingeniería de Requisitos para Sistemas de Información*. Tesis doctoral. Universidad de Sevilla. Disponible en <http://www.lsi.us.es/~amador/publicaciones/tesis.pdf.zip>. Recuperado el 15 de abril de 2012.
- Duval, E., Hodgins, W., Sutton, S. & Weibel, S. L. (2002). Metadata Principles and Practicalities, *D-Lib Magazine*, 8, (4). Disponible en <http://www.dlib.org/dlib/april02/weibel/04weibel.html>. Recuperado el 17 de julio de 2011.
- Earthy, J, Sherwood B & Bevan N. (2001). The Improvement of Human-Centred Processes - facing the challenge and reaping the benefit of ISO 13407. *International Journal of Human-Computer Studies*, 55, (4), 553-585 Disponible en <http://www.sciencedirect.com/science/article/pii/S1071581901904938>. Recuperado el 4 de noviembre de 2011.
- Ellis,A.L, Wagner, E.D., and LongMire, W.R. (1999) *Managing Web-Based Training*. , Alexandria, VA.
- EOE Foundation (2003) *Educational Objects Economy: Building Communities that Build Knowledge*, Disponible en <http://www.eoe.org>. Recuperado el 23 de agosto de 2011.
- Ertmer P. & Newby T.(1993) : *Conductismo, cognitivismo y constructivismo: una comparación de los aspectos críticos desde la perspectiva del diseño de instrucción*. Disponible en http://ares.unimet.edu.ve/programacion/psfase3/modII/biblio/conductismo-cognitivismo_constructivismo.pdf. Recuperado el 21 de agosto de 2011.
- Ertmer, P. A., Newby, T. J. (1993). Behaviorism, cognitivism, constructivism: Comparing

- critical features from an instructional design perspective. *Performance Improvement Quarterly*, 6, (4).
- Feldstein, M. (2002) .What is usable e-learning?. *eLearn Magazine* , 9.Disponible en <http://elearnmag.acm.org/featured.cfm?aid=581845#body-1>. Recuperado el 8 de febrero de 2012.
- Fernández-Manjón, B., Moreno-Ger, P., Sierra, J.L. y Martínez-Ortiz, I. (2007). Uso de estándares aplicados a TIC en Educación. *Informe N° 16*. Instituto de Tecnología Educativas (ITE) Centro Nacional de Información y Comunicación Educativa (CNICE), Madrid: Ministerio de Educación. España.
- Fernández-Manjón,B. , Sierra, J.,Martínez-Ortiz, I., Moreno-Ger,P. (2009). Estandarización y Modelado Educativo. *Informe N° 20*. Instituto de Tecnología Educativas (ITE), Centro Nacional de Información y Comunicación Educativa, (CNICE), Madrid: Ministerio de Educación. España.
- Fernández-Pampillón, A., Domínguez Romero, E., de Armas Ranero, I. (2012) *Herramienta de evaluación de la Calidad de Objetos de Aprendizaje (COdA): guía del usuario*. Disponible en eprints.ucm.es/12533/1/COdA_v1_0_definitivo.pdf. Recuperado el 21 de febrero de 2012.
- Fernandez, C. y Salinero, M. (1999). El diseño de un plan de formación como estrategia de desarrollo empresarial: estructura, instrumentos y técnicas. *Revista Complutense de educación*, 10,181-242. <http://http://revistas.ucm.es /index.php/RCED/article /view/RCED9999120181A /1 7245>. Accedido el 22 de mayo de 2011.
- Fernández, J. & Millán J. (2003) .CGRAPHIC: Educational Software for Learning the Foundations of Programming. *Computer Applications in Engineering Education*, 11, (4), 167-178. Wiley and Sons, Inc. Disponible en www.lcc.uma.es/~afdez/Papers/ cae2004.pdf. Recuperado el 6 de octubre de 2011
- Ferran, N., Guerrero-Roldan, A.E., Mor, E. & Minguillon, J. (2009): User centered design of a learning object repository. In Kurosu, M. (Ed.) *Proceedings of the 1st International Conference on Human Centered Design, HCI International. Lecture Notes in Computer Science*, (pp.679-688).
- Ferré X., Moreno A.M., Integración de la IPO en el Proceso de Desarrollo de la Ingeniería del Software: Propuestas existentes y temas a resolver. *Actas de Interacción 2004*. Universidad de Lleida.
- Ferré, X.(2003). Incrementos de Usabilidad al Proceso de desarrollo Software. *Actas de las VIII. Jornadas Ingeniería del Software y Bases de Datos, JISBD2003*. Alicante.
- Figueroa Martínez,J., Morales L. y Castillo L. (2008). Modificación del Módulo SCORM para el Soporte Básico del IMS-LD. *MoodleMoot 2008*, Barcelona, España. Disponible en www.moodlemoot.net/2008/file.php/1/abstracts/126-223-1-PB.pdf. Recuperado el 21 de agosto de 2011.
- Floría Cortés, A. (2000). *Recopilación de Métodos de Usabilidad*. SIDAR. Disponible en

<http://www.sidar.org/recur/desdi/traduc/es/visitable/Herramientas.htm>

- Foix, C. & Zavando, S. (2002). *Estándares e-learning. Estado del Arte*. Corporación de Investigación Tecnológica de Chile. INTEC.
- Friesen, N., Fischer, S. & Roberts, A. (2004). *CanCore Guidelines for the Implementation of Learning Object Metadata* (IEEE 1484.12.1-2002) v.2. Disponible en <http://cancore.athabasca.ca/en/guidelines.html>. Recuperado el 29 de agosto de 2011.
- Galindo, M., Simó M., Prieto, J. y Segret, R. (2010). *Escaneando la informática*. Universitat Oberta de Catalunya. Disponible en http://www.certified-easy.com/aa.php?isbn=ISBN:8497881109&name=Escaneando_la_inform%C3%A1tica. Recuperado el 15 de noviembre de 2011.
- Galvis A. (1996): Software educativo multimedia aspectos críticos no seu ciclo de vida. *Revista Brasileira de Informática no Educação. Sociedade Brasileira de Computação*, 9-18. Disponible en www.janus.ufse.br:1085/revista/nr1/galvis_p.htm. Recuperado el 15 de noviembre de 2011.
- Galvis, A., (2000). *Ingeniería de software educativo*. 2da. reimpresión. Universidad de Los Andes. Ediciones UNIANDES. Colombia.
- Galvis, P., Gómez, C. y Mariño, D. (1998). Ingeniería de software educativo con modelaje orientado por objetos: un medio para desarrollar micromundos interactivos. *Informática Educativa*, 11, (1).
- García Álvarez, J.L. (2006). El Paradigma del Modelo Prototipo de la Ingeniería del Software, una opción viable para el desarrollo de software educativo. En *XX Simposio Internacional de Computación en Educación*. Mexico. Disponible en www.somece.org.mx/simposio06/memorias/contenido/grupo3/pdf/6_GarciaAlvarezJoseLuis.pdf. Recuperado el 15 de noviembre de 2011.
- García Aretio, L. (2001). *La educación a distancia. De la teoría a la práctica*. Barcelona: Ariel.
- García Aretio, L. (2005). *Objetos de aprendizaje. Características y repositorios*. Disponible en <http://ares.unimet.edu.ve/programacion/cursos/tobj/articObj2.pdf>. Recuperado el 22 de mayo de 2010.
- García Aretio, L., Ruíz, M. y Domínguez, D. (Coord) (2007). *De la educación a distancia a la educación virtual*. Barcelona: Ariel.
- García E. y Sicilia, M. (2006). *Evaluación de la Usabilidad. Técnicas y Métodos de Evaluación*. Disponible en <http://es.scribd.com/doc/6671230/Tema4-Cuarta-Parte>. Recuperado el 10 de febrero de 2012.
- Gasson, S. (2003). Human-centered vs. user-centered approaches to information system design. In *Journal of Information Technology Theory and Application (JITTA)*, 5, (2), 29-46. Disponible en http://findarticles.com/p/articles/mi_qa4008/

is_200301/ai_n9192785/. Recuperado el 15 de noviembre de 2011.

- Gedica, G., Hamborg, K. & Duentzsch, I.(1999): The IsoMetrics usability inventory: An operationalization of ISO 9241-10. Supporting summative and formative evaluation of software systems. *Behaviour & Information Technology* 18, (3), 151-164. Disponible en <http://ftp.dcs.ed.ac.uk/teaching/cs4/www/hci/questionnaires/manu115e.pdf>. Recuperado el 8 de junio de 2012.
- Gerken, J., Bak, P., Jetter, J., Klinkhammer, D. & Reiterer, H. (2008). How to use interaction logs effectively for usability evaluation. In *CHI 2008 Workshop BELIV'08: Beyond time and errors - novel evaluation methods for Information Visualization*. Disponible en <http://www.inf.uni-konstanz.de/pubsys/publishedFiles/GeBaJe08.pdf>. Recuperado el 15 de febrero de 2012.
- González, P., Lorés, J. y Pascual, A. (2002) *Evaluación Heurística*. Capítulo del libro "La interacción persona ordenador", libro electrónico de la Asociación Interacción Persona Ordenador (AIPO). Disponible en la dirección web: http://www.aipo.es/libro/libroe.php#evaluacion_heuristica.
- Good, T. L. & Brophy, J. E. (1990). *Educational psychology: A realistic approach*. (4th ed.). White Plains, NY: Longman
- Granollers, T. (2004). *MPIu+a. Una metodología que integra la Ingeniería del Software, la Interacción Persona-Ordenador y la Accesibilidad en el contexto de equipos de desarrollo multidisciplinares*. Tesis Doctoral. Universitat de Lleida.
- Granollers, T., Perdrix, F. y Lorés, J. (2004). Incorporación de Usuarios en la Evaluación de la Usabilidad por Recorrido Cognitivo. *Interacción 2004, Lleida. AIPO*. Disponible en www.aipo.es/articulos/3/42.pdf. Recuperado el 10 de febrero de 2012.
- Hadad G., Kaplan G., Oliveros A., Leite J.C.S.P., (1997). Construcción de Escenarios a partir del Léxico Extendido del Lenguaje. In *Proceedings 26 JAIIO, Sociedad Argentina de Informática y Comunicaciones*. Argentina.
- Hamel, C. J., and Ryan-Jones, D. (2002). Designing Instruction with Learning Objects. *International Journal of Educational Technology*, 3, (1). Disponible en <http://www.ascilite.org.au/ajet/ijet/v3n1/hamel/>. Recuperado el 20 de noviembre de 2011.
- Harper, P. D. and Norman, K. L. (1993). Improving user satisfaction: The questionnaire for user interaction satisfaction version 5.5. *The 1st Annual Mid-Atlantic Human Factors Conference*, Virginia Beach, VA, 224-228. Disponible en <http://lap.umd.edu/QUIS/publications/harper1993.pdf>. Recuperado el 26 de febrero de 2012.
- Harris, D. (1996). *Crating a Knowledge Centric Informacion Technology Enviroment*, Harris Training & Consulting Services Inc., Seattle, WA,
- Harrison, G. & McKinnon, J. (1999). Cross-cultural research in management control

- systems design: A review of the current state. *Accounting, Organizations and Society*, 24, 483–506. Disponible en <http://directory.umm.ac.id/Data%20Elmu/jurnal/A/Accounting,%20Organizations%20and%20Society/Vol24.Issue56.July199/215.pdf>. Recuperado el 9 de febrero de 2012.
- Hart, S. G., Staveland, L. E. (1988). Development of Nasa-Tlx (Task Load Index): Results Of Empirical And Theoretical Research. In P. A. Hancock Y N. Meshkati (Eds.), *Human Mental Workload*. North-Holland, Amsterdam (pp. 139-183). Disponible en <http://humansystems.arc.nasa.gov/groups/TLX/doblonadas/NASA-TLXChapter.pdf>. Recuperado el 3 de junio de 2012.
- Hassan, Y. (2002). Diseño Hipermedia centrado en el usuario. En: No Solo Usabilidad, n° 1. Disponible en <http://www.nosolousabilidad.com/articulos/hipermedia.htm>. Recuperado el 20 de marzo de 2012.
- Hassan Y. y Ortega, S. (2009). *Informe APEI sobre usabilidad*. APEI, Asociación Profesional de Especialistas en Información. Disponible en <http://www.nosolousabilidad.com/manual/>
- Hassan Y., Fernández, F. y Iazza, G. (2004). Diseño Web Centrado en el Usuario: Usabilidad y Arquitectura de la Información. *Hipertext.net*, 2. Disponible en www.hipertext.net/web/pag206.htm. Recuperado el 20 de marzo de 2012.
- Hernández, E. (2005). *Curso estándares y especificaciones e-learning*. Universidad UNIACC, Santiago, Chile.
- Hernández, L., Iriarte, L., Arroche, A., Marco, M., Pernías P. y Rodríguez, I. (2006). Utilización de estructuras de aprendizajes IMS-LD en la enseñanza de la programación. In *Proceedings of the 4th International Conference on Multimedia and Information and Communication Technologies in Education*, (pp. 2105-2112), Sevilha, España. Disponible en <https://www.cisuc.uc.pt/publication/show/323>.
- Herrera, M., (2004). Las nuevas tecnologías en el aprendizaje constructivo. *Revista Iberoamericana de Educación*, 34, (4). Disponible en <http://www.rieoei.org/deloslectores/821Herrera.PDF>. Recuperado el 25 de abril de 2012.
- Hewett, T, Baecker, R., Card, S., Carey, T., Gasen, J., Mantei, M., Perlman, G., Strong, G. & Verplank, W. (1997). Curricula for Human Computer Interaction. *ACM SIGCHI*. Disponible en <http://www.acm.org/sigchi/cdg>. Recuperado el 18 de marzo de 2011.
- Hilera, J. R. y Hoya R. (2010). *Estándares de e-learning: Guía de consulta*. Alcalá de Henares: Universidad de Alcalá. Disponible en www.cc.uah.es/hilera/GuiaEstandares.pdf. Recuperado el 5 de setiembre de 2011.
- Hix, D. & Hartson, H. (1993). *Developing User Interfaces: Ensuring Usability Through Product and Process*. John Wiley and Sons, New York (NY), USA.
- Hom, J. (1998). *The usability methods toolbox handbook*. Disponible en: <http://usability.jameshom.com>. Recuperado el 3 de febrero de 2011.

- Horila, M., Nokelainen, P., Syvänen, A. & Överlund, J. (2002). *Criteria for the pedagogical usability, version 1.0*. Hämeenlinna, Finland: *Häme Polytechnic and University of Tampere*. Disponible en http://ccita2010.uca.es/c/document_library/get_file?uuid=03428cb8-f542-40b3-adcd-192fb36334fe&groupId=14125_
Recuperado el 27 de febrero de 2012.
- Humphrey, Watts S. (1995). *A Discipline for Software Engineering*, Reading, Mass., Addison-Wesley
- IEEE Standards Collection (1993). *Software Engineering, IEEE Standard 610.12-1990*.
- IEEE Std. 610.12 -1990 (1990). *IEEE Standard Glossary of Software Engineering Terminology*. IEEE Computer Society. Disponible en <http://www.idi.ntnu.no/grupper/su/publ/ese/ieee-se-glossary-610.12-1990.pdf>. Recuperado el 4 de noviembre de 2011.
- IMS Global *Learning Consortium Inc. Overview of Specifications*. Disponible en <http://www.imsglobal.org> . Recuperado el 6 de setiembre de 2011.
- IMS_LD (2003). *Estándar IMS Learning Design*. Disponible en <http://www.imsglobal.org/learningdesign/> .Recuperado el 6 de setiembre de 2011.
- IMS-CC (2011). *Estándar IMS common cartridge*. Disponible en <http://www.imsglobal.org/commoncartridge.html>. Recuperado el 6 de setiembre de 2011.
- IMS-CP (2007). *Estándar IMS content packaging*. Disponible en <http://www.imsglobal.org/content/packaging/>.Recuperado el 7 de setiembre de 2011.
- IMS-LIS (2009). *Especificación de servicios de información del aprendizaje*. Disponible en <http://www.imsglobal.org/lis.cfm>. Recuperado el 7 de setiembre de 2011.
- IMS-LTI (2009). *Estándar destinado a la Interoperabilidad de herramientas de aprendizaje*. Disponible en <http://www.imsglobal.org/ti/index.html>. Recuperado el 7 de setiembre de 2011.
- IMS-QTI (2008). *Estándar IMS Question & Test Interoperability Specification*. Disponible en <http://www.imsglobal.org/question/>. Recuperado el 7 de setiembre de 2011.
- IMS-SS (2003). *Estándar de secuenciación IMS*. Disponible en <http://www.imsglobal.org/simplesequencing/index.html>. Recuperado el 7 de setiembre de 2011.
- Instone, K. (1996). *Site usability heuristics for the web*. Disponible en <http://instone.org/heuristics>
- ISO 13407 (1999) - *Human-centred design processes for interactive systems*. Disponible en http://www.iso.org/iso/catalogue_detail.htm?csnumber=21197. Recuperado el 15 de noviembre de 2011.

- ISO 9241-11 (1998). *Ergonomic requirements for office work with visual display terminals (VDTs). Part 11: Guidance on usability*. Disponible en <http://www.it.uu.se/edu/course/homepage/acsdvt09/ISO9241part11.pdf> Recuperado el 15 de noviembre de 2011.
- ISO 9241-210 (2010) - *Ergonomics of human-system interaction -- Part 210: Human-centred design for interactive systems*. Disponible en http://www.iso.org/iso/catalogue_detail.htm?csnumber=52075<http://url/>. Recuperado el 15 de noviembre de 2011.
- ISO/IEC 19788 (2011). *Information technology for learning, education and training. Metadata for Learning Resource*. International Standards Organization, (Working Draft).
- ISO/IEC 9126-1 (2001) *Software Engineering - Product quality - Part 1: Quality model*. Disponible en http://webstore.iec.ch/preview/info_isoiec9126-1{ed1.0}en.pdf. Recuperado el 14 de febrero de 2012.
- ISO/IEC 9126-2 (2003). *Software engineering — Product quality — Part 2: External metrics* Disponible en http://webstore.iec.ch/preview/info_isoiec9126-2%7Bed1.0%7Den.pdf. Recuperado el 4 de junio de 2012.
- Jacobson, I., Booch, G. & Rumbaugh, Y. (2000). *El proceso unificado de desarrollo de software*. Ed. Addison-Wesley Iberoamericana, España
- Jenkins T. (2001). The motivation of students of programming. In *Proceedings of the 6th annual conference on Innovation and technology in computer science education Kingdom 2001*, Canterbury, United. Disponible en <http://bioinfo.uib.es/~joemiro/semDOC/p53-jenkins.pdf>. Recuperado el 19 de setiembre de 2011.
- Jonassen, D. H. (1991) Objectivism versus constructivism: do we need a new philosophical paradigm?. *Educational Technology Research and Development*, 39, (3).
- Jonassen, D., Mayes, T. & McAleese, R. (1993). A manifesto for a constructivist approach to uses of technology in higher education. In T.M. Duffy, J. Lowyck, and D.H. Jonassen (Eds.), *Designing environments for constructive learning*. Heidelberg, Germany: Springer-Verlag Berlin.
- JORUM + Project (2004). *The JISC Online Repository for (learning and teaching) Materials*. Disponible en http://www.jorum.ac.uk/docs/Vol1_Fin.pdf. Recuperado el 21 de mayo de 2011.
- Kay, J., Barg, M., Fekete, A., Greening, T., Hollands, O., Kingston, J. & Crawford, K. (2000). Problems-based learning for foundation computer science courses. *Computer Science Education*, 10, (2), 109-128. Disponible en http://www.cs.usyd.edu.au/~judy/PBL/tr_cse_pbl99.pdf. Recuperado el 19 de de setiembre de 2011.
- Kioska.Net. (2008). *Métodos rápidos (RAD, XP)*. Disponible en

- <http://es.kioskea.net/contents/genie-logiciel/>. Recuperado el 11 de noviembre de 2011.
- Kirakowski, J., Claridge, N. & Whitehand, R. (1998): Human centered measures of success in Web site design. *In Proceedings Human Factors and the Web Workshop, Basking Ridge, NJ*. Disponible en <http://zing.ncsl.nist.gov/hfweb/att4/proceedings/kirakowski/index.html>. Recuperado el 8 de junio de 2012.
- Kölling, M. (1999). The Problem of Teaching Object Oriented Programming. *Journal of Object Oriented Programming* 11, (8). Disponible en http://kar.kent.ac.uk/21879/2/the_problem_of_teaching_object-oriented_kolling_1.pdf. Recuperado el 19 de de setiembre de 2011.
- Kort, J., Steen, M.G.D., de Poot, H., ter Hofte, H., & Mulder, I. (2005). *Studying usage of complex applications*. In *Proceeding Measuring Behav, 2005*, Noldus Information Technology, (pp.266-269). Disponible en https://doc.telin.nl/dsweb/Get/File-51916/Kort_et_al_2005.pdf. Recuperado el 13 de febrero de 2012.
- Kreitzberg, C. (2008). The LUCID Framework. In *Cognetics Corporation*, Disponible en <http://www.leadersintheknow.biz/Portals/0/Publications/Lucid-Paper-v2.pdf>. Recuperado el 11 de noviembre de 2011.
- Kruchten, P (1996). A rational development process. *Crosstalk*, 9, (7), 11-16. Disponible en http://www.unix.eng.ua.edu/~crutcher/class/cs600/papers/A_Rational_Development_Process.pdf. Recuperado el 11 de noviembre de 2011.
- Kruchten, P. (2003). *The Rational Unified Process— An Introduction* (3ed). Boston: Addison-Wesley. Disponible en <http://www.mypearsonstore.com/bookstore/product.asp?isbn=0321197704>. Recuperado el 13 de noviembre de 2011.
- Kuniavsky, M. (2003). *Observing The User Experience: A Practitioner's Guide to User Research*. San Francisco: Elsevier.
- Kurilovas, E., Birenienė, V. & Serikovien, S. (2011) . Methodology for evaluating quality and reusability of learning objects. *Electronic Journal of e-Learning* 9(1), 39–51. Disponible en <http://www.ejel.org/issue/download.html?idArticle=166>. Recuperado el 25 de febrero de 2012.
- Kurland, D., Pea, R., Clement, C. & Mawby, R. (1986). A Study of the development of programming ability and thinking skills in High school students. *Journal educational computing research* 2, (4) . Disponible en http://telearn.archives-ouvertes.fr/docs/00/19/05/39/PDF/A29_Kurland_etal_86.pdf. Recuperado el 6 de octubre de 2011
- L'Allier, J. (1998). *NETg's Precision Skilling: The linking of occupational skills descriptors to training interventions*. Disponible en <http://www.netg.com/research/pskillpaper.htm>. Recuperado el 22 de mayo de 2011.
- Leacock, T. L. & Nesbit, J. C. (2007). A framework for evaluating the Quality of multimedia learning resources. *Educational Technology and Society*, 10, (2), 44-59.

Disponible en www.ifets.info/journals/10_2/5.pdf. Recuperado el 18 de febrero de 2012.

Leal Fonseca, D.E. (2009). *Objetos de Aprendizaje: cuatro años después. Objetos de Aprendizaje - Prácticas y perspectivas educativas*. Pontificia Universidad Javeriana – Cali. Disponible en [http // portales.puj.edu.co/ javevirtual/ portal/ Documentos/ Publicaciones/ Publicacion_2009.pdf](http://portales.puj.edu.co/javevirtual/portal/Documentos/Publicaciones/Publicacion_2009.pdf). Recuperado el 30 de setiembre de 2011.

Leite, J.C.S.P., Hadad, G.D.S., Doorn, J.H. & Kaplan, G.N. (2000), A Scenario Construction Process. *Requirements Engineering Journal*, 5. Disponible en <http://Fwww-di.inf.puc-rio.br/~julio/lct-pub/rej2000.pdf>. Recuperado el 20 de abril de 2012.

Lewis, C. & Rieman, J.(1993).*Task-centered user interface design. A Practical Introduction*. Disponible en [http:// uow.ico5.janison.com/ ed/ subjects/ EDGI957/resources/HCIComplete.pdf](http://uow.ico5.janison.com/ed/subjects/EDGI957/resources/HCIComplete.pdf). Recuperado el 6 de febrero de 2012.

Lewis, J. R. (1994). Sample sizes for usability studies: Additional considerations. *Human Factors*, 36, (2), 368-378. Disponible en [http:// drjim.0catch.com/ SampleSizesForUsabilityStudie.pdf](http://drjim.0catch.com/SampleSizesForUsabilityStudie.pdf). Recuperado el 23 de febrero de 2012.

Liber, O. & Corley, L. (2003). Learning Technology Standards - Where's The Pedagogy?. In A. Méndez-Vilas et al. (eds.), *Proceedings 2nd International Conference on Multimedia and Information & Communication Technologies in Education*, (pp 855-859). Consejería de Educación, Ciencia y Tecnología de la Junta de Extremadura.

Line Kolås (2010) *Implementation of pedagogical principles into the software design process of e-learning applications*. Doctoral Thesis. Department of Computer and Information Science Faculty of Information Technology, Mathematics and Electrical Engineering Norwegian University of Science and Technology.

Linn, M., & Dalbey, J. (1989). Cognitive consequences of programming instruction. In E. Soloway & J. C. Spohrer (Eds.). *Studying the novice programmer*, Hillsdale, NJ: Lawrence Erlbaum (pp. 57-81) Disponible en [http:// edr.sagepub.com/ content /14 /5/14.full.pdf](http://edr.sagepub.com/content/14/5/14.full.pdf). Recuperado el 18 de de setiembre de 2011.

Lister, R., Adams E.S., Fitzgerald, S., Fone, W., Hamer, J., Lindholm, M., McCartney, R., Moström, J.E., Sanders, K., Seppällä, O., Simon, B. & Thomas, L. (2004). A Multi-National Study of Reading and Tracing Skills in Novice Programmers. *SIGSCE Bulletin*, 36, (4), 119-150. Disponible en [http://citeseerx.ist.psu.edu/ viewdoc/ download?doi=10.1.1.110.5796&rep=rep1&type=pdf](http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.110.5796&rep=rep1&type=pdf). Recuperado el 20 de junio de 2012

Llamosa, R., Guarín I., Moreno, G., & Baldiris S. (2009). *Sistema Hipermedia Adaptativo Para la enseñanza de los Conceptos Básicos de la Programación Orientada a Objetos*. Disponible en [http://lsm.dei.uc.pt/ribie/docfiles/ txt2003326195840A016.pdf](http://lsm.dei.uc.pt/ribie/docfiles/txt2003326195840A016.pdf). Recuperado el 17 de junio de 2012

LOM (2002). *Learning Object v1 Metadata Final Draft*. Institute of Electrical and

- Electronics Engineers, New Cork, USA. Disponible en http://ltsc.ieee.org/doc/wg12/LOM_1484_12_1_v1_Final_Draft.pdf . Recuperado el 21 de mayo de 2011.
- LOM-ES (2008). Perfil de Aplicación LOM-ES v1. GT9 / SC 36. *Disponible en* www.xtec.es/~jvivanco/lom-es_v1.pdf. Recuperado el 7 de setiembre de 2011.
- Longmire, W. (2000). *A primer on learning objects*. ASTD Learning Circuits. Disponible en <http://www.learningcircuits.org/mar2000/primer.html> . Recuperado el 10 de marzo de 2011.
- López-Gil, J., Navarro-Molina, C., García, R. y Aleixandre-Benavent, R . (2010). Análisis de la arquitectura de webs mediante tests de estrés de navegación, de usabilidad y eye tracking. *Revista El Profesional de la Informacion* ,19. (4) ,359 – 367. Disponible en <http://repositori.udl.cat/bitstream/handle/10459.1/41447/fulltext.pdf>. Recuperado el 13 de febrero de 2012.
- López, C. (2005). *Los Repositorios de Objetos de Aprendizaje como soporte a un entorno e-learning* Tesis doctoral. Universidad de Salamanca. Disponible en <http://www.biblioweb.dgsca.unam.mx/libros/repositorios/index.htm>. Recuperado el 10 de marzo de 2011.
- López, J. (2008). *Algoritmos y Programación en la Educación Escolar*. Disponible en <http://www.eduteka.org/pdfdir/FGPUPonenciaAlgoritmos.pdf>. Recuperado el 10 de setiembre de 2011.
- Loucopoulos P. y Karakostas V. (1995). *System Requirements Engineering*, McGraw-Hill International series in Software Engineering,
- Lowe, D. (2000). A skills development framework for learning computing tools in the context of engineering practice. *European Journal of Engineering Education* 25, 45-56. Disponible en <http://services.eng.uts.edu.au/~dbl/archive/2000-Low00d.pdf>. Recuperado el 19 de junio de 2012.
- Luna-Pérez, H., Mezura-Godoy, C., Benítez-Guerrero, E., García-Gaona. A. (2001). Modelado de escenarios colaborativos para e-learning. En *Memorias del XXII Congreso Nacional y VIII Congreso Internacional de Informática y Computación (CNCIIC 2009)*. Ensenada, México. Disponible en <http://www.somece.org.mx/simposio/memorias/documentos/F140.doc>. Recuperado el 25 de abril de 2012.
- Lynch, P. & Horton, S. (1999). *Web style guide. Basic design principles for creating web sites*. Yale University Press.
- Macleod, M. & Rengger, R.(1993). *The Development of DRUM: A Software Tool for Video-assisted Usability Evaluation*. In *People and Computers VII*, Cambridge University Press. *Disponible en* www.nigelbevan.com/papers/drum93.pdf. Recuperado el 22 de febrero de 2012.
- MADI -UNED. Protocolo Evaluación Materiales Impresos. IUED. Disponible en

http://www.uned.es/iued/subsitio/html/documentos/Protocolo_MADI.pdf.
Recuperado el 26 de enero de 2012

Manchón, E. (2003). *Evaluación heurística (o por expertos) de la usabilidad*. Alzado.org. Disponible en http://www.alzado.org/articulo.php?id_art=74. Recuperado el 20 de mayo de 2012.

Maner, W.(1997). *Prototipado*. Disponible en <http://www.sidar.org/recur/desdi/traduc/es/visitable/maner/Prototipado.htm>

Mayhew, D. J. (1999). *The Usability Engineering Lifecicle. A practioner's handbook for user interface desing. San Francisco. Morgan Kaufmann.*

Mayhew, D.J. (1999). *The Usability Engineering Lifecycle: A practitioner's Handbook for User Interface Design*. Morgan Kaufman.

McCracken, M., Almstrum, V., Diaz, D., Guzdial, M., Hagan, D., Kolikant, Y.B.,Laxer, C., Thomas, L., Utting, I. & Wilusz, T. (2001). A multi-national, multi-institutional study of assessment of programming skills of first-year CS students. *SIGCSE Bulletin*, 33, (4), 125-180. Disponible en <http://www.deepdyve.com/lp/acm/a-multi-national-multi-institutional-study-of-assessment-of-R6JwiLwJXz?key=citeulike>. Recuperado el 15 de octubre de 2011.

McLean, N. & Lynch, C. (2003). *Interoperability between Information and Learning Environments: Bringing the Gaps*. Disponible en http://www.imsglobal.org/DLims_white_paper_publicdraft_1.pdf. Recuperado el 10 de marzo de 2011.

Medina, J. M., López. M. G. (2006). LOCOME: Metodología de Construcción de Objetos de Aprendizaje. *Actas del III Simposio Pluridisciplinar sobre Objetos de Aprendizaje y Diseños de Aprendizaje apoyados en las tecnologías (SPDECE06)*. Oviedo, España.

MELT. (2007). Metadata Ecology for Learning and Teaching project web site. *Disponible en* http://info.melt-project.eu/shared/data/melt/melt_brochure30_5final.pdf. Recuperado el 26 de febrero de 2012.

Mendoza, L., Pérez, M. y Grimán, A.(2005). Prototipo de Modelo Sistémico de Calidad (MOSCA) del Software. *Computación y Sistemas Revista Iberoamericana de Computación*, VIII (3), Mexico.

Menéndez , V. y Prieto M. (2010). Sistemas de Gestión Integral de Objetos de Aprendizaje. *IEEE RITA Revista Iberoamericana de Tecnologías del Aprendizaje* 5, 56- 62. Disponible en <http://rita.det.uvigo.es/201005/uploads/IEEE-RITA.2010.V5.N2.pdf#page=18>. Recuperado el 11 de noviembre de 2011.

Mergel, B. (1998). *Instructional design and learning theories*. University of Saskatchewan, College of Education. Disponible en <http://www.usask.ca/education/coursework/802papers/mergel/brenda.htm> Recuperado el 22 de agosto de 2011.

- Moisil, I., Domnariu, C. y Rogozea, L. (2009) Web-based Citizens Education for a Healthy Life-Style. In *Proceedings of the 8th WSEAS Int. Conf. on Artificial intelligence, knowledge engineering & data bases (AIKED '09)*. Disponible en <http://www.wseas.us/e-library/conferences/2009/cambridge/AIKED/AIKED06.pdf>. Recuperado el 19 de noviembre de 2011.
- Molich, R., Ede, M., Kaasgaard, K., and Karyukin, B (2004). Comparative Usability Evaluation. *Behaviour & Information Technology*, 23, (1), 65-74.
- Molich, R.& Nielsen, J. (1990). Heuristic evaluation of user interfaces. In *Proceedings of the Conference on Human factors in Computing Systems: Empowering people (SIGCHI 1990)* Seattle: ACM Press.
- Molpeceres, A. (2002). *Procesos de desarrollo RUP, XP y FDD*. AT Javahispano. Disponible en <http://www.javahispano.org/contenidos>. Recuperado el 19 de noviembre de 2011.
- Mondragón Donés, S., Vergara Monedero, M., Company Calleja, P. (2005) Diferencial Semántico: una herramienta al servicio del diseño emocional de máquinas herramientas. *International Journal of Industrial Ergonomics*, 35 (11), 1021-1029. Disponible en www3.uji.es/~pcompany/MVC06.pdf. Recuperado el 4 de junio de 2012.
- Mor, M., Garreta, M, y Galofré, M. (2007). Diseño Centrado en el Usuario en Entornos Virtuales de Aprendizaje, de la Usabilidad a la Experiencia del Estudiante. *Actas del IV Simposio Pluridisciplinar sobre Diseño, Evaluación y Desarrollo de Contenidos Educativos Re utilizables (SPDECE'07)*. Disponible en <http://spdece07.ehu.es/actas/Mor.pdf>. Recuperado el 31 de marzo de 2011.
- Morales, E. M., Gómez, D.A. y García Peñalvo, F.J. (2008). HEODAR.: Herramienta para la Evaluación de Objetos Didácticos de Aprendizaje Reutilizables. *Actas del X Simposio Internacional de Informática Educativa (SIIE'08)*. Salamanca.J, Velázquez Iturbide A., García Peñalvo, F. y Gil González, A. (Eds.) Colección Aquilafuente, ediciones Universidad de Salamanca.
- Morales, E. M., Muñoz, C., Conde, M. Á, Peñalvo F. J. (2010). Resultados de la aplicación de la Herramienta de Evaluación de Objetos Didácticos de Aprendizaje Reutilizables (HEODAR) en Moodle. *Acta del Simposio Pluridisciplinar sobre Diseño y Evaluación de Contenidos Digitales Educativo (SPDECE 2010)*. Universidad de Cádiz
- Moran T. P. (1981). The command language grammar: a representation for the user interface of interactive systems. *International Journal of man-machine studies*, 15.
- Moursund, D. (2008). Computational Thinking and Math Maturity: Improving Math Education. K-8 Schools. Disponible en <http://pages.uoregon.edu/moursund/Books/Math/K8-Math.pdf>. Recuperado el 10 de setiembre de 2011
- MUMMS (2003). *Human Factors Research Group: Measuring the usability of multimedia systems*. University College Cork, Ireland. Disponible en

<http://www.ucc.ie/hfrg/questionnaires/mumms/info.html>

Nagamachi M. (1995) Kansey engineering: a new ergonomic consumer-orientated technology for consumer development. *Journal. of Industrial Ergonomics* 15, 3-11.

Negroponte N. (1994). *Being Digital*. Vintage books, Nueva York, NY.

Nesbit, J. ,Belfer, K. & Leacock , T. (2009) . *Learning Object Review Instrument (LORI) User Manual* .Disponible en <http://www.elera.net/eLera/Home/Articles/LORI%20manual>. Recuperado el 21 de febrero de 2012.

Nesbit, J.C. & Leacock, T.L. (2009). Collaborative argumentation in learning resource evaluation. In *Handbook of research on learning design and learning objects: issues, applications and technologies* . L. Lockyer, S. Bennet, S. Agostinho, y B. Harper (pp 574-588). Disponible en <http://www.sfu.ca/~tleacock/Publications/NesbitLeacockPrePrint-LOHandbook.pdf>. Recuperado el 26 de enero de 2012.

Nielsen, J. (1993). *Usability Engineering* . *Academic Press Professional, Boston, MA*.

Nielsen, J. (1994a). *Ten Usability Heuristics*. Disponible en http://www.useit.com/papers/heuristic/heuristic_list.html

Nielsen J. (1994b) Heuristic evaluation. In Nielsen, j. And Mack, R. I. (Eds.) *Usability Inspection Methods*, John Wiley & Sons. Disponible en <http://www.useit.com/jacob/inspectbook.html>

Nielsen, J. (1994c). *Usability Labs*. Disponible en <http://www.useit.com/papers/uselabs.html>

Nielsen, J. (2000). *Why you only need to test with 5 users*. Disponible en <http://www.useit.com/alertbox/20000319.html>

Nielsen J. (2006). *Quantitative Studies: How Many Users to Test?* .Disponible en http://www.useit.com/alertbox/quantitative_testing.html

Nielsen J. and Landauer, T (1993). A mathematical model of the finding of usability problems. In *Proceedings of CHI '93, Proceedings of the INTERACT '93 and CHI '93 conference on Human factors in computing systems* .

Nielsen, J & Pernice, K. (2010). *Técnicas de Eyetracking para usabilidad web*. Anaya Multimedia.

Nielsen, J. & Mack, R.L. (1994). *Usability Inspection Methods* . John Wiley & Sons, New York, NY.

Nokelainen, P. (2006). An empirical assessment of pedagogical usability criteria for digital learning material with elementary school students. *Educational Technology & Society*. 9, (2), 178-197. Disponible en www.ifets.info/journals/9_2/15.pdf. Recuperado el 2 de junio de 2012.

- Norman, D. A. & Draper, S. W. (1986). *User Centered System Design*. Lawrence Erlbaum . Associates, Hillsdale, N.J.
- O'Malley, C., Draper, S. & Riley, M. (1984). Constructive interaction: A method for studying user-computer-user interaction. In *Proceedings of Interact '84*. London
- Olsen, A., Smolentzov, L. & Strandvall, T. (2009). *Retrospective Think Aloud and Eye Tracking Comparing the value of different cues when using the retrospective think aloud method in web usability testing*. Disponible en http://www.tobii.com/Global/Analysis/Training/WhitePapers/Tobii_RTA_and_EyeTracking_WhitePaper.pdf. Recuperado el 18 de febrero de 2012.
- Olsina, L., Godoy, D., Lafuente, G. & Rossi, G, (1999). Specifying Quality Characteristics and Attributes for Websites . In *ICSE '99 Web Engineering Workshop*. Los Angeles, US. Disponible en http://gidis.ing.unlpam.edu.ar/downloads/pdfs/Olsina_WebE.pdf. Recuperado el 19 de mayo de 2011.
- Orjuela, A. y Rojas, M. (2008). La metodología de desarrollo ágil como una oportunidad para la ingeniería del software educativo. *Revista Avances en Sistemas e Informática*, 5, (2). Disponible en www.revista.unal.edu.co/index.php/avances/article/view/10037. Recuperado el 22 de noviembre de 2011.
- Ortega M., Pérez, M. & Rojas T. (2000). A Model for Software Product Quality with a Systemic Focus. In *The 4th World Multiconference on Systemics, Cybernetics and Informatics SCI 2000 and The 6th International Conference on Information Systems, Analysis and Synthesis ISAS 2000*. Orlando, USA. Disponible en <http://www.lisi.usb.ve/publicaciones>. Recuperado el 22 de noviembre de 2011.
- Paganelli, L. & Paternò, F. (2002). Intelligent analysis of user interactions with web applications. In *International Conference on Intelligent User Interfaces*. ACM Press (2002), (pp. 111-118. Disponible en http://giove.isti.cnr.it/attachments/publications/2002-A2-72_0.pdf. Recuperado el 10 de febrero de 2012.
- Parra Castrillón. E. (2011). Propuesta de metodología de desarrollo de software para objetos virtuales de aprendizaje -MESOVA. *Revista Virtual Universidad Católica del Norte*, 34. Disponible en <http://revistavirtual.ucn.edu.co/index.php/RevistaUCN/article/view/332>. Recuperado el 31 de marzo de 2011.
- Pérez, M., Rojas T., Ortega, M., & Alvarez, C. (1999). *Toward Systemic Quality: Case study*. In *4Th Squad Meeting, Squad*. Porlamar, Venezuela,
- Pérez, M., Mendoza, L., Rojas, T. & Grimán, A. (2001) .Systemic Quality Model for System Development Process: Case Study. In *Seventh Americas Conference on Information Systems AMCIS 2001*. Disponible en <http://www.lisi.usb.ve/publicaciones>. Recuperado el 22 de noviembre de 2011.
- Perkins, D. N. (1991). Technology meets constructivism: Do they make a marriage? *Educational Technology*, 5, 18-23.

- Pohl, K. (1997). Requirements Engineering: An Overview. *Encyclopedia of Computer Science and Technology*, 36. Disponible en <http://sunsite.informatik.rwth-aachen.de/CREWS/reports96.htm>. Recuperado el 10 de abril de 2012.
- Pohl, K.(1993). The three dimensions of requirements engineering . *En Proc. 5th Intl. Conf. on Advanced Information Systems Engineering. Aachen, Germany.* Disponible en <http://citeseerx.ist.psu.edu/viewdoc/download/doi/D10.1.1.43.9761>. Recuperado el 15 de abril de 2012.
- Pons, D., Hilera, J.R., Pagés, C. (2011). ISO/IEC 19788 MLR: Un Nuevo Estándar de Metadatos para Recursos Educativos". *IEEE-RITA* , 6, (3), 140-145. Disponible en <http://rita.det.uvigo.es/201108/uploads/IEEE-RITA.2011.V6.N3.A8.pdf>. Recuperado el 9 de setiembre de 2011.
- Powell, Thomas A. (2001). *Diseño de sitios Web, Manual de referencia*. McGraw Hill, Madrid.
- Preece, Y. ,Rogers, H. ,Sharp, D. ,Benyon, S. , Holland, T. (1994) . *Human- Computer Interaction*. Addison Wesley, Harlow, England.
- Pressman, R. (2002) *Ingeniería del Software. Un enfoque práctico*. 5ta edición, McGraw-Hill Interamericana, Madrid, España.
- Proyecto CANALS. Instituto de Tecnologías educativas. Ministerio de Educación de España. Disponible en <http://recursostic.educacion.es/canals/web>
- Proyecto CUIP2 (2009). *Una solución integral al problema de enseñar y aprender a programar*. Universidad de los Andes. Disponible en http://www.colombiaaprende.edu.co/html/mediateca/1607/articles-205832_recurso_1.pdf. Recuperado el 28 de diciembre de 2011.
- Proyecto Tuning (2007). *Reflexiones y perspectivas de la Educación Superior en América Latina. Informe Final*. Proyecto Tuning América Latina. 2004-2007. Disponible en http://tuning.unideusto.org/tuningal/index.php?option=com_docman&task=docclick&Itemid=191&bid=54&limitstart=0&limit=5. Recuperado el 19 de junio de 2012.
- Q4R. (2007). *Quality for Reuse project web site*. Disponible en <http://www.q4r.org>
- Quinn, C.N. (1996). Pragmatic evaluation: lessons from usability. In *Proceedings of 13th Annual Conference of the Australasian Society for Computers in Learning in Tertiary Education*, Disponible en <http://www.ascilite.org.au/conferences/adelaide96/papers/18.html>. Recuperado el 1 de junio de 2012.
- QUIS (2001). *Questionnaire for User Interaction Satisfaction* . University of Maryland. Disponible en http://cognitron.umd.edu/QUIS/PaperQUIS_85463/agreement.html
- Redish, J. (2007). Expanding usability testing to evaluate complex systems. *Journal of Usability Studies*. 2, (3), 102-111. Disponible en http://216.92.41.4/upa_publications/jus/2007may/utesting-complex.pdf. Recuperado el 23 de febrero

de 2012.

- Redish, J., Bias, R. G., Bailey, R., Molich, R., Dumas, J., & Spool, J. M. (2002). Usability in practice: formative usability evaluations - evolution and revolution. In *CHI '02 Extended Abstracts on Human Factors in Computing Systems*. Disponible en www.itu.dk/~hulda/Usability/artikler/p885-redish.pdf. Recuperado el 5 de abril de 2012.
- Reeves, T. , Benson, L. . Elliott,D., Grant, M. Holschuh, D. , Kim, B., Kim, H., Lauber, E. & Loh, C. (2002) . Usability and Instructional Design Heuristics for E-Learning Evaluation. In *Proceedings of World Conference on Educational Multimedia, Hypermedia & Telecommunications (ED-MEDIA 2002)*, (pp. 1615-1621). Disponible en <http://www.eric.ed.gov/ERICWebPortal/contentdelivery/servlet/ERICServlet?accno=ED477084>. Recuperado el 2 de junio de 2012
- Reeves, T. (1994). Evaluating what really matters in computer-based education. In *M. Wild & D. Kirkpatrick (Eds.) Computer education: new perspectives*, (pp. 219-246). Perth, Australia: MASTEC, Disponible en <http://www.eduworks.com/Documents/Workshops/EdMedia1998/docs/reeves.html>. Recuperado el 3 de junio de 2012.
- Restrepo Jaramillo, J. (2008). Valoración de algunos instrumentos para evaluar la satisfacción del usuario en el contexto Web. *Revista Kepes. 4*. Universidad de Caldas. Colombia. Disponible en http://200.21.104.25/kepes/downloads/Revista%204_10.pdf. Recuperado el 8 de junio de 2012
- Reyes, J.y Libreros F. (2011) Método para la evaluación integral de la usabilidad en sistemas e-learning. *Revista Educación en Ingeniería, 12*, 69-79. Disponible en http://www.acofi.edu.co/revista/Revista12/2011_II_30.pdf. Recuperado el 20 de enero de 2012.
- Riihiaho, S. (2002). The pluralistic usability walk-through method. *Ergonomics in Design, 10* (3), 23–27.
- Robins, A., Rountree, J. & Rountree, N. (2003). Learning and Teaching Programming: A Review and Discussion. *Computer Science Education, 13*, (2), 137-172. Disponible en home.cc.gatech.edu/csed/uploads/2/robins03.pdf. Recuperado el 20 de junio de 2012.
- Rodriguez, F. y Cardona, P. (2005). *Metodología para el desarrollo de cursos virtuales basados en objetos de aprendizaje*. Universidad Autónoma de Aguascalientes. Disponible en www.willydev.net/InsiteCreation/v1.0/prev/METODOVIRTUAL.pdf. Recuperado el 10 de mayo de 2011.
- Rosenbaum, S. (2002). Usability in Practice: Field Studies. In *Proceedings of CHI 2002 Conference, Usability in Practice session Field Studies—Evolution and Revolution*. Disponible en <http://www.teced.com/PDFs/Chi2002.pdf>. Recuperado el 15 de febrero de 2012.
- Rosenfeld, L.& Morville, P. (1998). *Information architecture for the world wide web*. Cambridge: O'Reilly.

- Rosson, M.B. & Carroll, J.M. (2002). *Usability Engineering: scenario-based development of HCI*. Morgan Kaufmann.
- Rovira, C. (2002). Estructuras de navegación para e-learning. *Revista El profesional de la información*, 11, (6). Disponible en <http://dialnet.unirioja.es/servlet/articulo?codigo=287149&orden=115878&info=link>. Recuperado el 20 de mayo de 2012.
- Rovira, P. (2007). *Usabilidad y Analítica Web*. Disponible en http://www.alzado.org/articulo.php?id_art=684. Recuperado el 12 de febrero de 2012.
- Royo, J. (2004). *Diseño Digital*. Ediciones Paidós Ibérica.
- Ruiz González R., Muñoz Arteaga J., Álvarez Rodríguez (2007). *Evaluación de objetos de aprendizaje a través del aseguramiento de competencias*. Universidad Autónoma de Aguas calientes, Centro de Ciencias. México. Disponible en <http://e-spacio.uned.es/fez/eserv.php?pid=bibliuned:19233&dsID=n03ruizgonz07.pdf>. Recuperado el 24 de febrero de 2012.
- Saettler, P. (2004). *The Evolution of American Educational Technology*. USA: Information Age Publishing Inc.
- Sánchez, J., Gil, R. y Oliva, M. (2010). Más allá del cuchillo de palo. Hacia una herramienta integrada para un verdadero diseño centrado en el usuario. En *XI Congreso Internacional de Interacción Persona Ordenador (Interacción 2010)*. Valencia, España. Disponible en <http://jordisan.net/blog/2011/mas-alla-del-cuchillo-de-palo-articulo-sobre-dcu-en-interaccion-2010>. Recuperado el 25 de octubre de 2011.
- Sánchez, J. (2011). En busca del Diseño Centrado en el Usuario (DCU): definiciones, técnicas y una propuesta. *No Solo Usabilidad*, 10. Disponible en <http://www.nosolousabilidad.com/articulos/dcu.htm>.
- Santiso, M.R., González .B. (2005). *Diseño multimedia en e-learning para el ámbito universitario*. Disponible en http://www.nosolousabilidad.com/articulos/multimedia_elearning.htm.
- Santos, O.C., Martin, L., Del Campo, E., Saneiro, M., Mazzone, E., Boticario, J.G. & Petrie, H. (2009). *User-centered design methods for validating a recommendations model to enrich learning management systems with adaptive navigation support*. Disponible en <http://www.easy-hub.org/workshops/umap2009/doc/paper7.pdf>. Recuperado el 25 de octubre de 2011.
- Schuman, L. (1996). *Perspectives on instruction*. Disponible en <http://edweb.sdsu.edu/courses/edtec540/Perspectives/Perspectives.html>. Recuperado el 9 de setiembre de 2011.
- SCORM 2004 4th Edition Specification (2009). *Disponible en* <http://www.adlnet.gov/capabilities/scorm/scorm-2004-4th> Recuperado el 9 de setiembre de 2011.

- Senge, P. (1990): *The Fifth Discipline*. Doubleday. Nueva York (versión en castellano: La Quinta Disciplina) Ed. Granica, S.A. Barcelona. Disponible en http://portal.unesco.org/es/ev.php-url_id=16044&url_do=do_topic&url_section=201.html. Recuperado el 15 de marzo de 2011.
- Sergeev, A. (n.d.). *User Interfaces Design*. Disponible en http://ui-designer.net/interface_design.htm
- Shneiderman B. (1986). *Eight Golden Rules of Interface Design* Disponible en <http://faculty.washington.edu/jtenenbg/courses/360/f04/sessions/schneidermanGold enRules.html>. Recuperado el 27 de febrero de 2012.
- Sicilia, M., y Sánchez Alonso, S. (2009). *Learning objects y learning designs: conceptos. Diseño y Evaluación de contenidos y actividades educativas reutilizables*. Information Engineering Research Unit. Universidad de Alcalá.
- Sicilia M. A., Lytras Miltiadis D. (2005). Scenario-oriented reusable learning object characterizations. *International Journal of Knowledge and Learning*, 1, 332-341. Disponible en http://www.cc.uah.es/msicilia/papers/Sicilia_IJKL_2005.pdf. Recuperado el 25 de abril de 2012.
- Sicilia, M.A. (2005). Reusabilidad y reutilización de objetos didácticos: mitos, realidades y posibilidades. *RED. Revista de Educación a Distancia, número monográfico II*. Disponible en <http://www.um.es/ead/red/M2/sicilia46.pdf> Recuperado el 15 de febrero de 2011.
- Sicilia, M.A. & García, E. (2003). On the Concepts of Usability and Reusability of Learning Objects . *International Review of Research in Open and Distance Learning* 4, (2). Disponible en <http://www.irrodl.org/content/v4.2/sicilia-garcia.html>. Recuperado el 15 de abril de 2012.
- Sierra, J. (2005). *Estudio de la influencia de un entorno de simulación por ordenador en el aprendizaje por investigación de la Física en Bachillerato*. Ministerio de Educación y Ciencia de España.
- SIG-CHI (2001) .Notes from E-learning Special Interest Group (SIG) . *Discussion at CHI 2001. Magazine eLearn*, 6. Disponible en <http://elearnmag.acm.org/archive.cfm?aid=566948>. Recuperado el 4 de junio de 2012
- Sommerville I. (2005). *Ingeniería de software*. 7ª edición, Prentice Hall – Pearson Educación, México.
- Spencer, D. & Warfel, T. (2007). *Card sorting: a definitive guide. Boxes and Arrows*. Disponible en http://www.boxesandarrows.com/view/card_sorting_a_definitive_guide . Recuperado el 10 de febrero de 2012.
- Spohrer, J., & Soloway, E. (1986). Analyzing the high-frequency bugs in novice programs. In E. Soloway, & S. Iyengar (Eds.), *Empirical studies of programmers*, (pp.230-251). Washington, DC, USA: Ablex Publishing Corporation.

- Squires, D. (1999). Usability and Educational Software Design: Special Issue of Interacting with Computers. *Interacting with Computers 11*, (5), 463-466. Disponible en <http://www.sciencedirect.com/science/article/pii/S0953543898000629>.
- Squires, D. & Preece, J. (1999). Predicting quality in educational software: Evaluating for learning, usability and the synergy between them. *Interacting with Computers. 11*, (5), 467-483. Disponible en <http://www.irit.fr/recherches/ICS/projects/twintide/upload/446.pdf>. Recuperado el 17 de mayo de 2012.
- Stephen Downes (2001) Learning Objects: Resources For Distance Education Worldwide. *International Review of Research in Open and Distance Learning 2*, (1). Disponible en http://www.downes.ca/files/Learning_Objects.doc. Recuperado el 7 de setiembre de 2011.
- Tognazzini, B. (2007). *First principles of interaction design*. Disponible en <http://www.asktog.com/basics/firstPrinciples.html>.
- Travis, D. (2011). ISO 13407 is dead. Long live ISO 9241-210!. Disponible en <http://www.userfocus.co.uk/articles/iso-13407-is-dead.html>
- Tullis, T. & Stetson, J. (2004). A Comparison of Questionnaires for Assessing Website Usability. *UPA Annual Conference*. Disponible en <http://home.comcast.net/~tomtullis/publications/UPA2004TullisStetson.pdf>. Recuperado el 4 de junio de 2012
- Turner, C. W., Nielsen, J., & Lewis, J. R. (2002). Current issues in the determination of usability test sample size: How many users is enough? In *Usability Professionals' Association 2002 Conference Proceedings*. Chicago:UPA. Disponible en drjim.0catch.com/samsize-upa02.pdf. Recuperado el 20 de febrero de 2012.
- Tuya, J., Ramos, I. y Dolado, J. (eds.) (2007). *Técnicas Cuantitativas para la Gestión en Ingeniería del Software*. Netbiblo, S.L.
- UsabilityNet (2006a). *What is usability?*. Disponible en http://www.usabilitynet.org/management/b_what.htm
- UsabilityNet (2006b). *ISO 13407*. Disponible en <http://www.usabilitynet.org/tools/13407stds.htm>
- van Amstel, J. (1984). Teaching Programming at various levels of formality. In *Proceedings of the IFIP TC3 Working Conf. on The role of programming in teaching informatics*, (pp. 143 – 148). Disponible en <http://dl.acm.org/citation.cfm?id=4906.4922&coll=DL&dl=GUIDE&CFID=94456549&CFTOKEN=46734615>. Recuperado el 19 de setiembre de 2011.
- Van der Veer G. (1990). *Human computer interaction: learning, individual differences and design recommendations*. Tesis Doctoral, Universidad de Vrije, Amsterdam,

- Vargas Cerdán, M. D. ,Rojano Cáceres, J. R. y García Gaona, R. M. (2009).Diseño de un Software Educativo para la Integración de Niños con Hipoacusia en Preescolar. *Revista Diseño e Interacción- FAZ*, 3.
- Varlamis, I. & Apostolakis, I. (2006). The present and future of standards for e-learning technologies. *Interdisciplinary Journal of Knowledge and Learning Objects*, 2, 59-76. Available at <http://ijklo.org/Volume2/v2p059-076Varlamis.pdf>
- Veenendaal, E. (1998). *Questionnaire based usability testing*. In Conference Proceeding European Software Quality Week, *Brussels*. Disponible en <http://www.erikvanveenendaal.nl/NL/files/Questionnaire%20based%20usability%20testing.pdf> .Recuperado el 8 de junio de 2012
- Velázquez, I. y Sosa, M. (2009). La usabilidad del *software* educativo como potenciador de nuevas formas de pensamiento. *Revista Iberoamericana de Educación*, 50, (4), 1-12. Disponible en www.rieoei.org/deloslectores/3032Sosa.pdf. Recuperado el 5 de febrero de 2011.
- Virzi, R. A. (1992). Refining the test phase of usability evaluation: how many subjects is enough?. *Human Factors*, 34,(4), 457-468.Disponible en <http://coursesite.uhcl.edu/hsh/PeresSc/Classes/PSYC5911www/Refiningthetestphase.pdf>. Recuperado el 10 de mayo de 2012.
- Vygotsky, S. (1978). *Pensamiento y lenguaje*. Buenos Aires: Pléyade.
- Vygotsky, S. (1979). *El desarrollo de los procesos psicológicos superiores*. Barcelona: Grijalbo.
- Weitzenfeld, A. (2005). *Ingeniería de software orientada a objetos con UML, Java e Internet*. Cengage Learning Editores.
- Wharton, C. (1992). *Cognitive Walkthroughs: Instructions, Forms and Examples*. Institute of Cognitive Science. Technical Report CU-ICS-92-17. University of Colorado, Boulder.
- Wharton, C., Rieman, J., Lewis, C., & Polson, P. (1994). *The cognitive walkthrough method: A practitioner's guide*. In J. Nielsen & R. L. Mack. (Eds.), *Usability inspection methods*, (pp.105–140). New York: Wiley.
- Wiley, D. (2009). Evaluating Open Educational Resources. *In the Open Education Conference 09: Crossing the chasm*, Vancouver, Canadá. Disponible en <http://www.ustream.tv/recorded/1983441>. Recuperado el 30 de setiembre de 2011
- Williams D. (2000) Evaluation of learning objects and instruction using learning objects. *In D. A. Wiley (Ed.), The instructional use of LOs*, Disponible en <http://reusability.org/read/chapters/williams.doc>. Recuperado el 2 de marzo de 2011.
- Wirth, N. P (1971). Program Development by Stepwise Refinement. *Communications of the ACM* 14, (4), 221 – 227. Disponible en <http://sunnyday.mit.edu/16.355/wirth->

refinement.html

- Wixon D., Jones S., Tse L. & Casaday G. (1994). Inspections and design re-views: framework, history, and reflection. In *Usability Inspection Methods*, Nielsen J. & Mack R. L. (Ed.), (pp. 79-104). John Wiley & Sons, NY.
- Wright, P. & Monk, A. F. (1989). Evaluation for design. In *People and Computers*, (pp. 345-358. Sutcliffe V, A. & Macaulay, L. Eds. Cambridge University Press, New York.
- Zapata, M. (2005). Secuenciación de contenidos y objetos de aprendizaje. *RED. Revista de Educación a Distancia, IV (Número monográfico II)*. Murcia (España). Disponible en <http://www.um.es/ead/red/M2/zapata47.pdf>. Recuperado el 23 de mayo de 2010.
- Zhu, H. & Zhou, M. (2003). Methodology First and Language Second: A Way to Teach Object Oriented Programming. In *Companion of the 18th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications OOPSLA'03*, (pp.140-147). Anaheim, CA. Disponible en <http://www.nipissingu.ca/faculty/haibinz/research/oopsla03.pdf>. Recuperado el 18 de de setiembre de 2011.
- Zijlstra, F.(1993). *Efficiency in work behaviour: A design approach for modern tools*. Delft:Delft University Press. Disponible en http://repository.tudelft.nl/assets/uuid:d97a028b-c3dc-4930-b2ab-a7877993a17f/ide_zijlstra_19931123.PDF. Recuperado el 23 de febrero de 2012
- Zülch G. & Stowasser, S. (2001). Usability evaluation of universal user interfaces with the computer-aided evaluation tool PROKUS. In *Proceedings HCI 2001*, (pp.645-649). <http://www.mmi-interaktiv.de/uploads/media/stowasser.pdf>. Recuperado el 27 de febrero de 2012.

APENDICE A. MATERIAL DEL OA

A.1. Escenarios

Componente	Descripción
Nombre	Video 1: ejemplo variables simples
Objetivo	Proporcionar información por diversos medios: texto, imagen y sonido sobre el pasaje de parámetros por valor utilizando variables simples.
Contexto	El usuario debe tener un cuenta Moodle del curso y una computadora con lo siguiente: <ul style="list-style-type: none"> • Complemento de Adobe Flash Player 10.0.22 o posterior. • Firefox 1.1 o posterior, Safari 1.0 o posterior, Google Chrome u Opera • Conexión de banda ancha de un mínimo de 500 Kbps .
Recursos	video
Actores	Estudiante
Set de episodios	<ol style="list-style-type: none"> 1. El usuario hace clic en el link que hace referencia al video ejemplo del pasaje de parámetros – variables simples. 2. Se despliega una ventana pop-up que presenta el objetivo del video y debajo el video embebido. 3. El usuario clickea en el video y comienza la reproducción. El video tiene controles para avanzar, retroceder, parar y pasar a pantalla completa. 4. El video presenta el problema sencillo “swap” a resolver que trabaja con variables simples. 5. El video tiene la pantalla dividida en a) el código fuente Pascal que resuelve el problema, b) la recreación de la memoria c) La pantalla de salida del programa. 6. El usuario observa como se ejecutan cada una de las sentencias del programa, las cuales están señaladas con un color. Simultáneamente en la otra parte de la pantalla se van mostrando los cambios en la memoria y en la pantalla de salida. 7. El usuario escucha a un locutor que explica los pasos más relevantes. 8. El usuario observa que los momentos claves son señalados además con globos explicativos. 9. Se resalta con el relato del locutor y visualmente el tipo de pasaje de parámetros por valor, en el sector de la memoria se destaca que no cambian los valores de los parámetros actuales cuando cambian los valores de los formales y que no se resuelve el problema con este tipo de pasaje de parámetros. 10. Se repiten los pasos 6 a 9 en donde el usuario tiene la libertad de “avanzar”, “retroceder” o “parar” el video. 11. El usuario presiona el botón “parar” que finaliza la reproducción del video.
Casos alternativos	Si el usuario no tiene los complementos necesarios, el sistema se lo indicará, podrá descargarlos y luego ver el video.
Dudas:	<ul style="list-style-type: none"> • Enunciado del problema a resolver. • Momentos claves a destacar en el programa Pascal. • Metáforas para recrear la memoria

Tabla A.1. Escenario 1: video explicando pasaje de parámetros por valor con variables simples

Componente	Descripción
Nombre	Video 2: ejemplo variables simples
Objetivo	Proporcionar información por diversos medios: texto, imagen y sonido sobre el pasaje de parámetros por referencia utilizando variables simples.
Contexto	El usuario debe tener un cuenta Moodle del curso y una computadora con lo siguiente: <ul style="list-style-type: none"> • Complemento de Adobe Flash Player 10.0.22 o posterior. • Firefox 1.1 o posterior, Safari 1.0 o posterior, Google Chrome u Opera • Conexión de banda ancha de un mínimo de 500 Kbps ..
Recursos	video
Actores	Estudiante
Set de episodios	<ol style="list-style-type: none"> 1. El usuario hace clic en el link que hace referencia al video ejemplo del pasaje de parámetros – variables simples. 2. Se despliega una ventana pop-up que presenta el objetivo del video y debajo el video embebido. 3. El usuario clickea en el video y comienza la reproducción. El video tiene controles para avanzar, retroceder, parar y pasar a pantalla completa. 4. El video presenta el problema sencillo “swap” a resolver que trabaja con variables simples. 5. El video tiene la pantalla dividida en a) el código fuente Pascal que resuelve el problema, b) la recreación de la memoria c) La pantalla de salida del programa. 6. El usuario observa como se ejecutan cada una de las sentencias del programa, las cuales están señaladas con un color. Simultáneamente en la otra parte de la pantalla se van mostrando los cambios en la memoria y en la pantalla de salida. 7. El usuario escucha a un locutor que explica los pasos más relevantes. 8. El usuario observa que los momentos claves son señalados además con globos explicativos. 9. Se resalta con el relato del locutor y visualmente el tipo de pasaje de parámetros por referencia, en el sector de la memoria se destaca que cambian los valores de los parámetros actuales cuando cambian los valores de los formales y se resuelve el problema con este tipo de pasaje de parámetros. 10. Se repiten los pasos 6 a 9 en donde el usuario tiene la libertad de “avanzar”, “retroceder” o “parar” el video. 11. El usuario presiona el botón “parar” que finaliza la reproducción del video.
Casos alternativos	Si el usuario no tiene los complementos necesarios, el sistema se lo indicará, podrá descargarlos y luego ver el video.
Dudas:	<ul style="list-style-type: none"> • Enunciado del problema a resolver. • Momentos claves a destacar visual y auditivamente en el programa Pascal. • Metáforas para recrear la memoria

Tabla A.2. Escenario 2: video explicando pasaje de parámetros por referencia con variables simples

Componente	Descripción
Nombre	Video 3: ejemplo variables estructuradas
Objetivo	Proporcionar información por diversos medios : texto , imagen y sonido sobre el pasaje de parámetros por valor utilizando variables estructuradas.
Contexto	El usuario debe tener un cuenta Moodle del curso y una computadora con lo siguiente: <ul style="list-style-type: none"> • Complemento de Adobe Flash Player 10.0.22 o posterior. • Firefox 1.1 o posterior, Safari 1.0 o posterior, Google Chrome u Opera • Conexión de banda ancha de un mínimo de 500 Kbps .
Recursos	video
Actores	Estudiante
Set de episodios	<ol style="list-style-type: none"> 1. El usuario hace clic en el link que hace referencia al video ejemplo del pasaje de parámetros – variables estructuradas. 2. Se despliega una ventana pop-up que presenta el objetivo del video y debajo el video embebido. 3. El usuario clickea en el video y comienza la reproducción. El video tiene controles para avanzar, retroceder, parar y pasar a pantalla completa. 4. El video presenta el problema sencillo para modificar elementos de un vector. 5. El video tiene la pantalla dividida en a) el código fuente Pascal que resuelve el problema, b) la recreación de la memoria c) La pantalla de salida del programa. 6. El usuario observa como se ejecutan cada una de las sentencias del programa, las cuales están señaladas con un color. Simultáneamente en la otra parte de la pantalla se van mostrando los cambios en la memoria y en la pantalla de salida. 7. El usuario escucha a un locutor que explica los pasos más relevantes. 8. El usuario observa que los momentos claves son señalados además con globos explicativos. 9. Se resalta con el relato del locutor y visualmente el tipo de pasaje de parámetros por valor, en el sector de la memoria se destaca que no cambian los valores de los parámetros actuales cuando cambian los valores de los formales y que en realidad no se resuelve el problema con este tipo de pasaje de parámetros. 10. Se repiten los pasos 6 a 9 en donde el usuario tiene la libertad de “avanzar”, “retroceder” o “parar” el video. 11. El usuario presiona el botón “parar” que finaliza la reproducción del video.
Casos alternativos	Si el usuario no tiene los complementos necesarios, el sistema se lo indicará, podrá descargarlos y luego ver el video.
Dudas:	<ul style="list-style-type: none"> • Enunciado del problema a resolver. • Momentos claves a destacar visual y auditivamente en el programa Pascal. • Metáforas para recrear la memoria

Tabla A.3.Escenario 3: video explicando pasaje de parámetros por valor con variables estructuradas

Componente	Descripción
Nombre	Video 4: ejemplo variables estructuradas
Objetivo	Proporcionar información por diversos medios : texto , imagen y sonido sobre el pasaje de parámetros por referencia utilizando variables estructuradas.
Contexto	El usuario debe tener un cuenta Moodle del curso y una computadora con lo siguiente: <ul style="list-style-type: none"> • Complemento de Adobe Flash Player 10.0.22 o posterior. • Firefox 1.1 o posterior, Safari 1.0 o posterior, Google Chrome u Opera • .Conexión de banda ancha de un mínimo de 500 Kbps .
Recursos	video
Actores	Estudiante
Set de episodios	<ol style="list-style-type: none"> 1. El usuario hace clic en el link que hace referencia al video ejemplo del pasaje de parámetros – variables estructuradas. 2. Se despliega una ventana pop-up que presenta el objetivo del video y debajo el video embebido. 3. El usuario clickea en el video y comienza la reproducción. El video tiene controles para avanzar, retroceder, parar y pasar a pantalla completa. 4. El video presenta el problema sencillo para modificar elementos de un vector . 5. El video tiene la pantalla dividida en a) el código fuente Pascal que resuelve el problema, b) la recreación de la memoria c) La pantalla de salida del programa. 6. El usuario observa como se ejecutan cada una de las sentencias del programa, las cuales están señaladas con un color. Simultáneamente en la otra parte de la pantalla se van mostrando los cambios en la memoria y en la pantalla de salida. 7. El usuario escucha a un locutor que explica los pasos más relevantes. 8. El usuario observa que los momentos claves son señalados además con globos explicativos. 9. Se resalta con el relato del locutor y visualmente el tipo de pasaje de parámetros por valor, en el sector de la memoria se destaca que cambian los valores de los parámetros actuales cuando cambian los valores de los formales y se resuelve el problema con este tipo de pasaje de parámetros. 10. Se repiten los pasos 6 a 9 en donde el usuario tiene la libertad de “avanzar”, “retroceder” o “parar” el video. 11. El usuario presiona el botón “parar” que finaliza la reproducción del video.
Casos alternativos	Si el usuario no tiene los complementos necesarios, el sistema se lo indicará, podrá descargarlos y luego ver el video.
Dudas:	<ul style="list-style-type: none"> • Enunciado del problema a resolver. • Momentos claves a destacar visual y auditivamente en el programa Pascal. • Metáforas para recrear la memoria

Tabla A.4. Escenario 4: video explicando pasaje de parámetros por referencia con variables estructuradas

Componente	Descripción
Nombre	Problemas resueltos
Objetivo	Facilitar el aprendizaje del “Pasaje de parámetros” por medio de la comprensión de las condiciones de un problema y la asimilación de la solución mediante la visualización paso a paso de la misma.
Contexto	El usuario debe tener un cuenta Moodle del curso y una computadora con lo siguiente: <ul style="list-style-type: none"> • Complemento de Adobe Flash Player 10.0.22 o posterior. • Firefox 1.1 o posterior, Safari 1.0 o posterior, Google Chrome u Opera • .Conexión de banda ancha de un mínimo de 500 Kbps .
Recursos	Archivo pdf
Actores	Estudiante
Set de episodios	<ol style="list-style-type: none"> 1. El usuario hace clic en el link que hace referencia a problemas resueltos. 2. Se despliega una ventana pop-up que contiene un mensaje indicando que se presentarán varios problemas resueltos y debajo un botón para la descarga indicando el tamaño del archivo. El archivo tiene formato pdf. 3. El usuario descarga el archivo. 4. El documento presenta un índice con enlaces a cada uno de los ejemplos. 5. El ejemplo presenta a) análisis top-down b) estructura jerárquica del programa con un diagrama c) solución con la explicación detallada de cada elemento y su función (tipos de datos, variables, ctes, parámetros, procedimientos y funciones) y las sentencias ejecutables. 6. Al final del ejemplo hay un botón que permite volver al principio del documento. 7. Se repiten los pasos 4 a 6 en donde el usuario tiene la libertad de recorrer el documento. 8. El usuario cierra el documento.
Casos alternativos	Si el usuario no tiene los complementos necesarios, el sistema se lo indicará, podrá descargarlos y luego ver el documento pdf.
Dudas:	<ul style="list-style-type: none"> • Enunciado de los ejemplos resueltos. • Momentos claves a destacar visualmente en el programa Pascal.

Tabla A.5.Escenario 5: Problemas resueltos

Componente	Descripción
Nombre	Actividad individual : autoevaluación
Objetivo	Brindar una herramienta que permita al estudiante evaluar su propio aprendizaje
Contexto	El usuario debe tener un cuenta Moodle del curso y una computadora con lo siguiente: <ul style="list-style-type: none"> • Complemento de Adobe Flash Player 10.0.22 o posterior. • Firefox 1.1 o posterior, Safari 1.0 o posterior, Google Chrome u Opera • .Conexión de banda ancha de un mínimo de 500 Kbps
Recursos	Cuestionario
Actores	Estudiante

Set de episodios	<ol style="list-style-type: none"> 1. El estudiante lee cada consigna del cuestionario. 2. Para cada consigna se presentan al menos tres o cuatro respuestas posibles, siendo una sola de ellas correcta. 3. El estudiante selecciona la opción que considera correcta (Multiple-choice). Tiene 2 intentos. 4. Aparecerá una corrección automática, de acuerdo a la respuesta. 5. Se repiten los pasos 1a 4 hasta finalizar con el cuestionario. 6. El estudiante presiona el botón “respuestas propuestas”, mediante el cual podrá conocer el resultado de su evaluación.
Casos alternativos	
Dudas:	<ul style="list-style-type: none"> • Enunciado de las consignas • Respuestas correctas de cada consigna • Distractores.

Tabla A.6. Escenario 6: Actividad individual: autoevaluación

Componente	Descripción
Nombre	Actividad grupal
Objetivo	Fomentar el trabajo colaborativo para la resolución de un problema que involucra “Pasaje de Parámetros” y fortalecer el aprendizaje de una mayor cantidad de estudiantes.
Contexto	El usuario debe tener un cuenta Moodle del curso y una computadora con lo siguiente: <ul style="list-style-type: none"> • Complemento de Adobe Flash Player 10.0.22 o posterior. • Firefox 1.1 o posterior, Safari 1.0 o posterior, Google Chrome u Opera. • Conexión de banda ancha de un mínimo de 500 Kbps .
Recursos	<ul style="list-style-type: none"> • Herramienta de comunicación “CHAT” del Curso Moodle. • Herramienta de colaboración WIKI por grupos en el Curso Moodle.
Actores	Grupo de Estudiantes y Profesor
Set de episodios	<ol style="list-style-type: none"> 1. Se forman equipos de tres a cinco estudiantes para resolver el problema. 2. Antes de iniciar la resolución del problema los estudiantes leen cuidadosamente las orientaciones específicas para trabajar con la WIKI que se encuentra en el curso Moodle. 3. Para la resolución de la consigna los estudiantes conversan y/o discuten la mejor alternativa de solución, pero sólo entre los integrantes del equipo. 4. Los integrantes del grupo utilizan la Wiki para la creación colectiva de la solución. 5. Los integrantes van editando el documento ,guardan los cambios , hasta llevar a una solución consensuada por todos los integrantes del problema.. 6. Al finalizar la actividad se envía un correo al profesor para su evaluación y devolución.
Casos alternativos	
Dudas:	<ul style="list-style-type: none"> • Enunciado del problema a resolver.

Tabla A.7. Escenario 7: Actividad grupal

Componente	Descripción
Nombre	Actividad de pares
Objetivo	Fomentar el trabajo colaborativo para la resolución de un problema que involucra “Pasaje de Parámetros” y fortalecer el aprendizaje de una mayor cantidad de estudiantes a través del trabajo de a pares.
Contexto	El usuario debe tener un cuenta Moodle del curso y una computadora con lo siguiente: <ul style="list-style-type: none"> • Complemento de Adobe Flash Player 10.0.22 o posterior. • Firefox 1.1 o posterior, Safari 1.0 o posterior, Google Chrome u Opera . • Conexión de banda ancha de un mínimo de 500 Kbps .
Recursos	<ul style="list-style-type: none"> • Herramienta colaborativa de documentos. • Tarea “subida avanzada de archivos” en el Curso Moodle.
Actores	Grupo de pares de Estudiantes y Profesor
Set de episodios	<ol style="list-style-type: none"> 1. Se forman equipos de dos estudiantes para resolver el problema. 2. Antes de iniciar la resolución del problema los estudiantes leen cuidadosamente las orientaciones específicas para trabajar en la resolución del problema. 3. Cada compañero enviará al otro su resolución por ejemplo vía e-mail. 4. Adicionalmente pueden trabajar con un documento en un espacio colaborativo como GoogleDocs. 5. Cada uno corrige el trabajo del compañero. 6. Entre ambos construyen una versión consensuada. 7. Al finalizar la actividad se sube el archivo a una Tarea creada a tal fin en el curso Moodle.. 8. El profesor realiza la evaluación de la tarea y envía una devolución al grupo de estudiantes.
Casos alternativos	
Dudas:	<ul style="list-style-type: none"> • Enunciado del problema a resolver.

Tabla A.8.Escenario 8: Actividad de pares

A.2. Introducción teórica: pasaje de parámetros

Subprogramas: Procedimientos y funciones

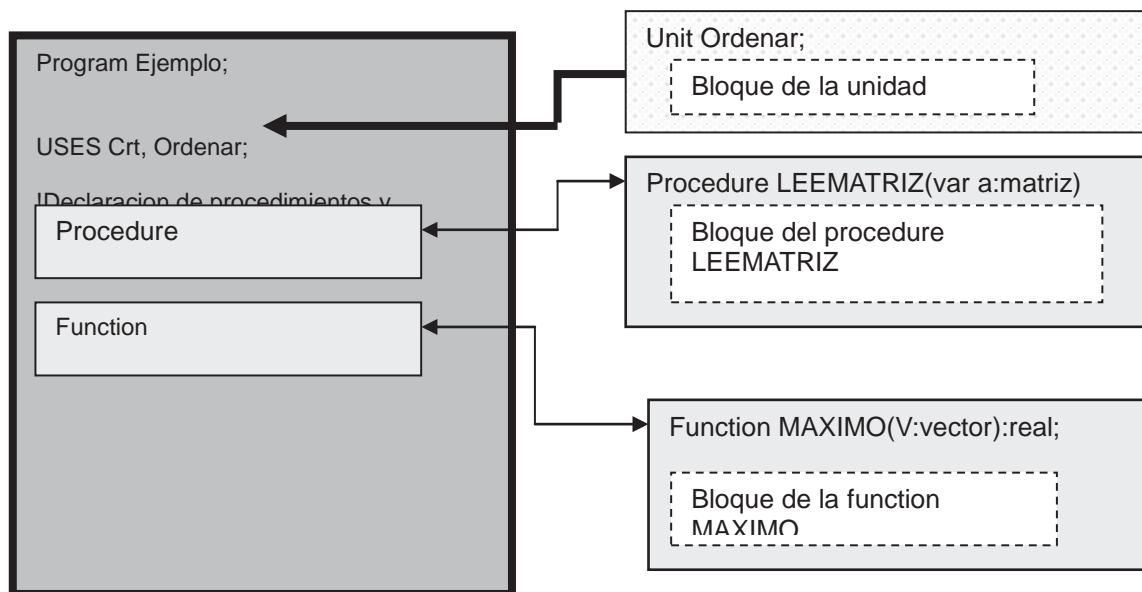
1 Introducción

Un programa PASCAL, hasta el capítulo anterior, se ha concentrado en la búsqueda de **un** algoritmo apropiado para resolver el cálculo deseado. La sección ejecutable del programa tiende a crecer considerablemente. Este estilo de programación nos lleva a programas extensos si el problema es suficientemente complejo. Los programas extensos son difíciles de corregir, modificar y mantener.

En el análisis top-down, el programador puede dividir cada tarea compleja en varias subtareas más simples, definiendo así algunos algoritmos (posteriores programas) para resolver cada una de ellas. A partir de ahora, esas subtareas serán tratadas como subprogramas.

Un **subprograma** es una sección del programa (idealmente autónoma), que realiza una tarea definida, la cual puede ser llamada (activada, invocada) por otras secciones del programa, siempre que se necesite, a fin de ejecutar la tarea asignada.

A continuación vemos un esquema de un programa Pascal con subprogramas. En el bloque que figura a la izquierda, se presenta las secciones de declaración de procedimientos y funciones, ya sea declarados explícitamente o incluidos a partir de unidades (USES).



A la derecha figuran los subprogramas procedimientos (**procedure**) y funciones (**function**) que forman parte del programa Ejemplo y dos módulos pre-compilados

(unidades Pascal conteniendo otros procedimientos, funciones y definiciones de tipos) llamados Crt y Ordenar.

En el presente capítulo veremos los procedimientos y funciones, y en el capítulo siguiente estudiaremos las unidades (Unit).

2 Procedimientos

2.1 Declaración de los procedimientos

En la sección de declaración de procedimientos se escribe el código fuente (en negrita) correspondiente a un algoritmo que soluciona una tarea específica.

PROGRAM principal;

CONST
Definición de constantes

TYPE
Declaración de tipos

VAR
Declaración de variables

PROCEDURE procedim1 (A,B,C: integer);

! Sección de declaraciones del procedimiento

.....

! Sección ejecutable del procedimiento

Begin

.....

.....

end;

BEGIN
;Sentencias ejecutables del programa

.....

END.

Identificador del procedimiento

Lista de parámetros formales
(la cantidad de parámetros formales puede ser 0,1,2,...n)

Como se aprecia en el esquema, el procedimiento replica el formato de un programa con la excepción de la palabra clave **procedure** en lugar de **program**.

Los *parámetros formales* son nombres de variables de determinado tipo cuyo objetivo es especificar la comunicación del subprograma con el programa (al cual generalmente pasa a llamarse programa principal). Estos parámetros (en otros lenguajes se los llama “mudos”) no actúan como variables comunes, i.e., no se les asignan valores.

2.2 Llamado a los procedimientos

Para lograr que un subprograma se ejecute, este debe ser llamado (invocado) desde la sección ejecutable del programa (para nuestro ejemplo). Para invocarlo se debe escribir el identificador del procedimiento seguido de parámetros actuales entre paréntesis.

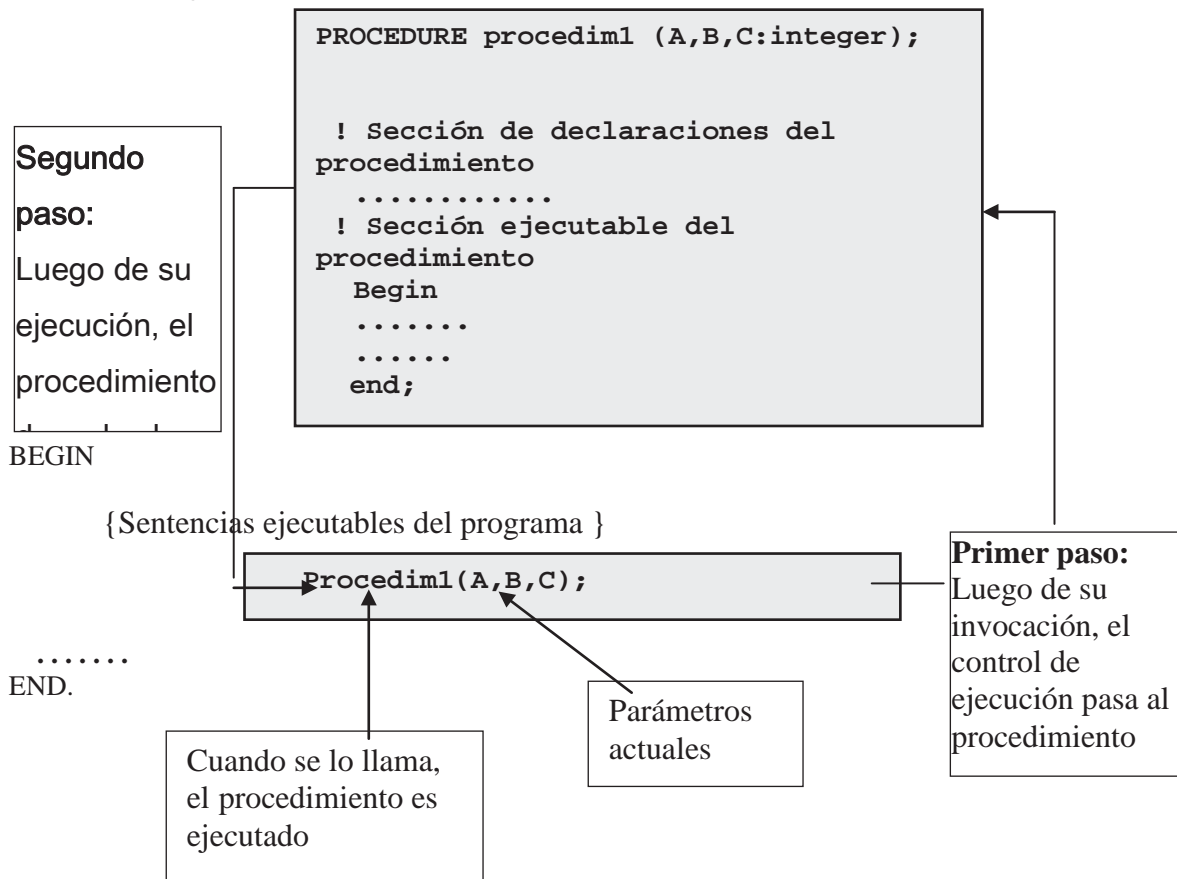
Los *parámetros actuales* son expresiones (entre ellas variables) que contienen la información necesaria para la ejecución del procedimiento. Estos parámetros se activan cuando el procedimiento es invocado para su ejecución.

PROGRAM principal;

CONST
 ¡Definición de constantes

TYPE
 {Declaración de tipos}

VAR
 {Declaración de variables}
 A,B,C:integer;

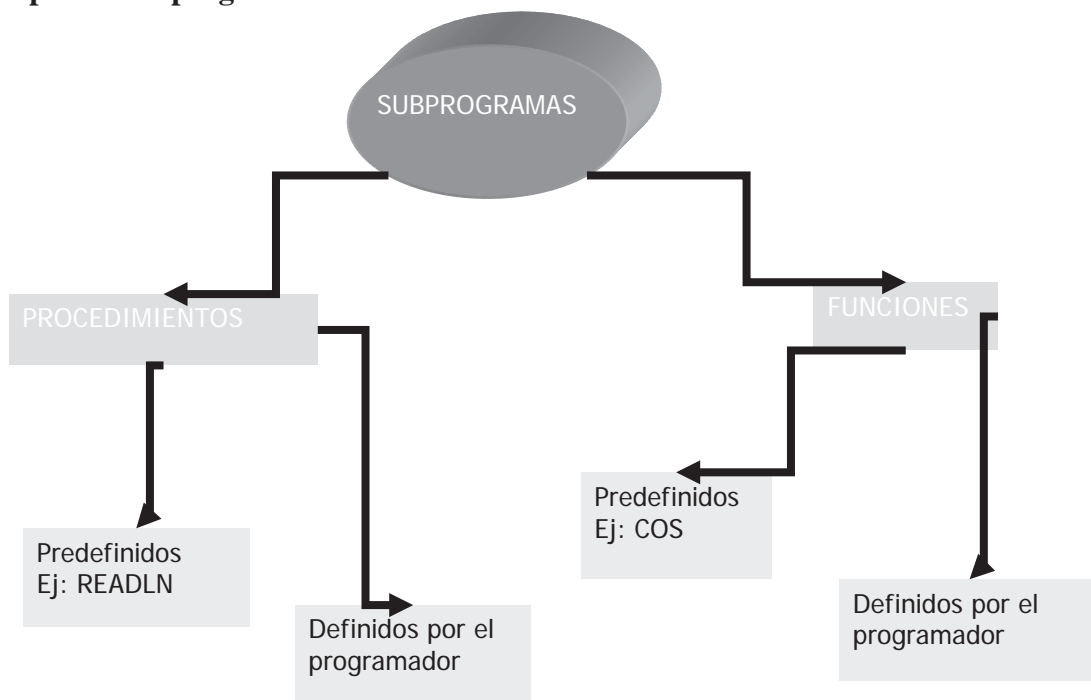


Cuando, durante la ejecución del programa Principal, el procedimiento es llamado el control de ejecución pasa al procedimiento. Este es ejecutado completamente, resolviendo un problema específico, enviando la solución al programa principal a través de los parámetros. Inmediatamente, el control de ejecución pasa a la sección ejecutable del programa, específicamente a la sentencia siguiente de la invocación del procedimiento.

2.3 Ventajas del uso de subprogramas

1. AHORRO DE CÓDIGO: tareas que se repiten pueden ser escritas en PASCAL solo una vez en un subprograma, luego se le llama tantas veces como sea necesario.
2. MODULARIDAD: los problemas son divididos en subproblemas (estrategia top-down) en forma conveniente; luego cada uno de ellos pueden ser llevados a subprogramas.
3. BUEN MANTENIMIENTO: los subprogramas pueden ser corregidos y modificados fácilmente. Es absolutamente conveniente que la comunicación entre el programa principal y el subprograma sea SÓLO a través de sus parámetros.

2.4 Tipos de subprogramas



Tanto los procedimientos como las funciones predefinidas ya fueron vistos previamente.

Los procedimientos predefinidos son invocados por su nombre (y a veces se le agregan parámetros). Su definición fue hecha previamente por el fabricante del software y no es accesible por el programador.

3. Su invocación (o llamado) crea una **sentencia** en la sección ejecutable.

Ejemplos:

```
clrscr; {procedimiento sin parámetros que figura en la unidad crt}
writeln('Este es un ejemplo',A,b, cos(a*b));
writeln es un procedimiento y os es una función, ambos poseen parámetros }
```

A continuación estudiaremos los subprogramas definidos por el programador.

2.5 Procedimientos del programador

Los **procedimientos definidos por el programador** son aquellos procedimientos creados por el programador (diseñados, programados y pre-compilados) escritos en una sección especial de declaración para luego ser invocados por su nombre (al que generalmente se le agregan parámetros). La invocación de los procedimientos crea una **sentencia** en la sección ejecutable.

2.5.1 Alcance global y local

- Definimos como *alcance* de un objeto (variable, tipo, etc.) a la esfera de acción de dicho objeto. Puede ser *global* o *local*.
- En la declaración de un programa (*programa principal*) encontramos la sección de definición de constantes, la sección de definición de tipos, la sección de declaración de variables, procedimientos y funciones que son *globales* al programa. Esto significa que son válidos en todo el programa.
- En la declaración de un procedimiento tenemos identificadores de parámetros (formales), la sección de definición de constantes, la sección de definición de tipos, la sección de declaración de variables, procedimientos y funciones que son **locales** al procedimiento declarado, i.e., *no se les conoce* fuera de él. Queda claro que los objetos globales siguen valiendo dentro del procedimiento.

2.5.2 Pasaje de parámetros

El mecanismo de comunicación entre el programa principal y los procedimientos se realiza a través del **pasaje de parámetros**.

- Para que exista comunicación entre programa principal y los procedimientos se deben usar parámetros (en caso contrario no hay comunicación o será considerada errónea, por ejemplo cuando se utilicen variables globales).
- Como ya se analizó, los parámetros que figuran en la declaración del procedimiento, se denominan parámetros formales. Debe indicarse el identificador y tipo del parámetro.
- Los parámetros que figuran en la sentencia de llamada al procedimiento se denominan parámetros actuales. Sólo debe indicarse su identificador (o expresión), que debe figurar en la sección de variables globales del programa principal.
- La cantidad entre parámetros actuales y formales debe coincidir. La correspondencia entre actuales y formales se determina por la posición en su respectiva lista.

Veremos el *pasaje de parámetros por valor* y el *pasaje de parámetros por variable*.

2.5.2.1 Pasaje de parámetros por valor

Veamos el siguiente ejemplo:

```
PROGRAM EjemploValor;

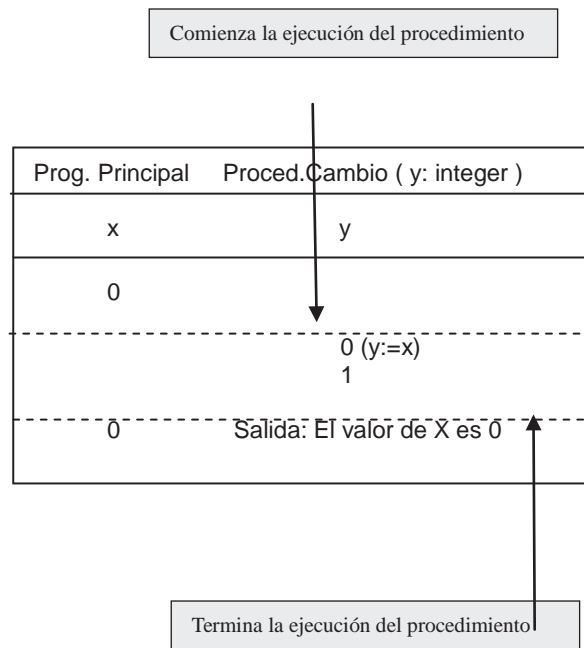
Var
  X:integer;
PROCEDURE cambio(y:integer);{y es un parámetro formal por valor}
  Begin
    Y:=1
  End;
BEGIN
  X:=0;
  Cambio(x);{x es un parámetro actual por valor}
  Writeln('El valor de X es', X)
END.
```

Las características del pasaje de parámetros por valor son:

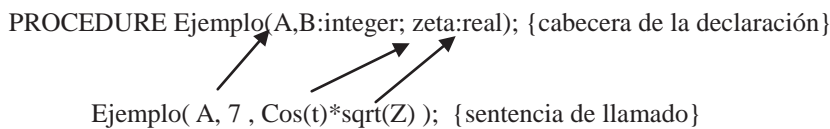
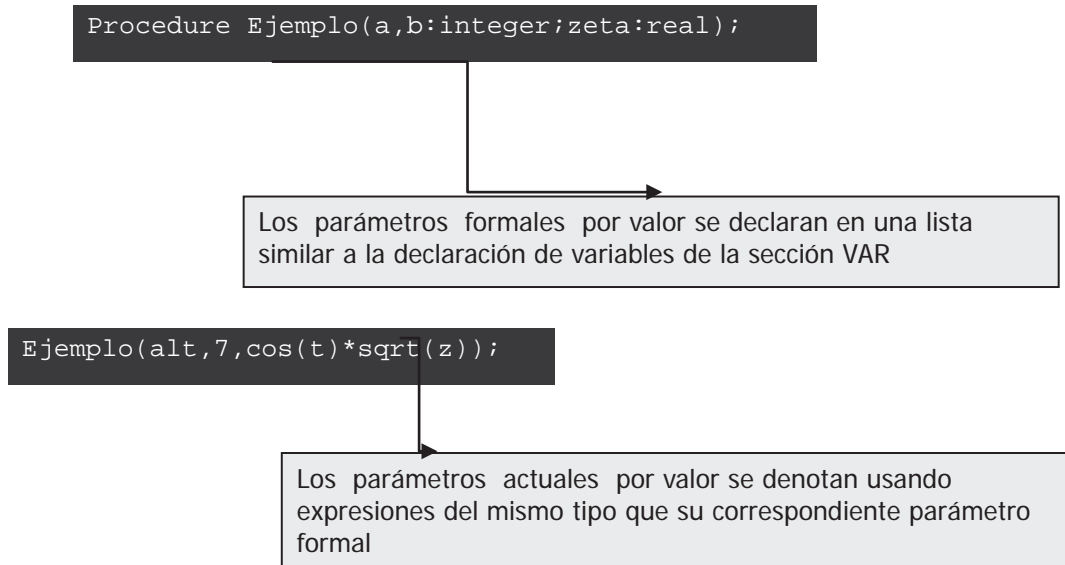
- 1.- Se establece una comunicación unidireccional (del programa principal hacia el procedimiento).
- 2.- Al parámetro formal se le asigna memoria y se le copia el valor del parámetro actual y se corta la comunicación entre programa principal y el procedimiento.
- 3.- Si se modifica el valor del parámetro formal dentro del procedimiento, el valor del parámetro actual NO CAMBIA.
- 4.- Si bien el parámetro formal debe ser una variable, el parámetro actual puede ser una expresión.

A continuación vemos una prueba de escritorio del programa EjemploValor:

```
PROGRAM EjemploValor;
Var
  x:integer;
PROCEDURE cambio(y:integer);
  Begin
    Y:=1
  End;
BEGIN
  x:=0;
  Cambio(x);
  Writeln('El valor de X es', x)
END.
```

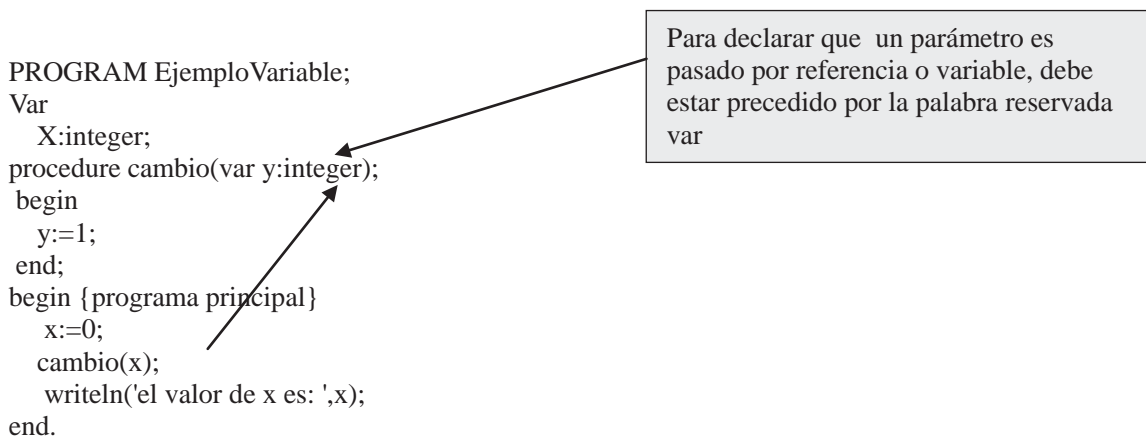


Sintaxis de la cabecera de un procedimiento y del llamado:



2.5.2.2 Pasaje de parámetros por variable

Veamos el siguiente ejemplo:



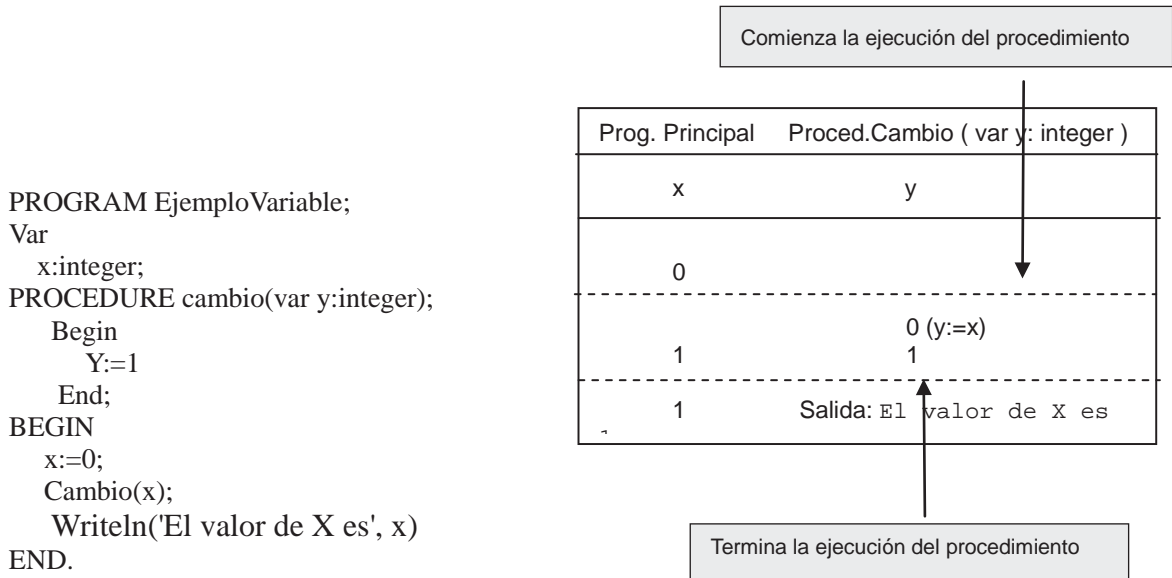
Las características del pasaje de parámetros por variable son:

- 1.- Se establece una comunicación bidireccional
- 2.- Cada vez que se modifica el parámetro formal, se modifica el parámetro actual. De esta forma, se pueden intercambiar datos. El programa principal le envía datos al procedimiento y el procedimiento le envía resultados al programa principal.

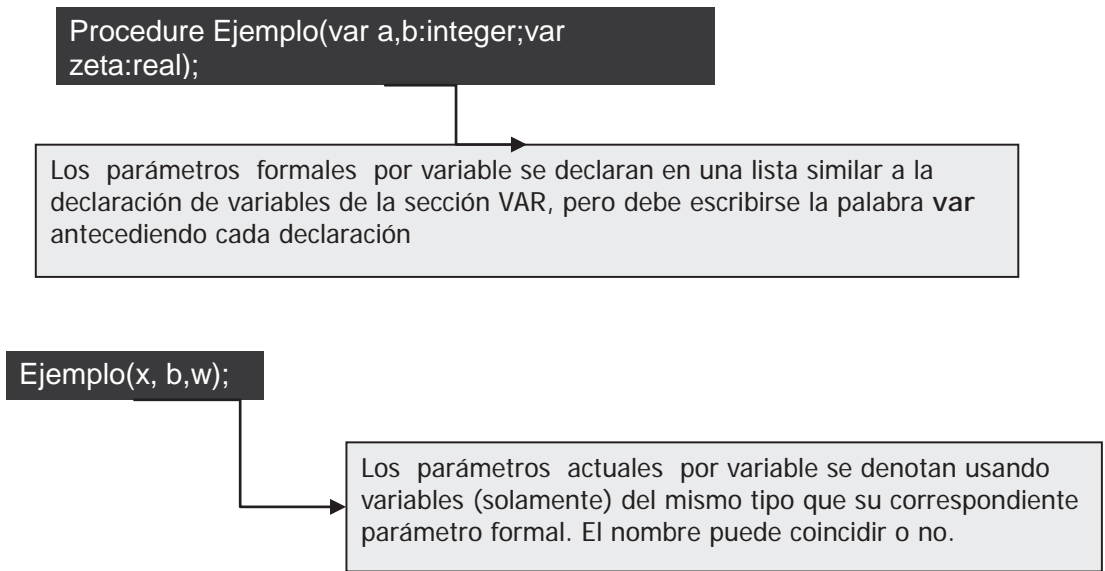
3.- No hay copia de valores. Se establece una referencia entre ellos, es decir, el parámetro actual le pasa la dirección donde él se encuentra, al parámetro formal.

4.- Los parámetros formales y actuales deben ser variables.

A continuación vemos una prueba de escritorio del programa EjemploVariable:



Sintaxis de la cabecera de un procedimiento y del llamado:



A.3. Enunciados de los problemas para los videos

- Variables simples: Ingresar dos valores enteros. Intercambiar los valores (Swap)
- Variables estructuradas: Ingresar un vector de tres componentes enteras. Modificar el contenido de la primera componente

A.4. Enunciados de los problemas para actividades

i) Actividad Grupal

1) Ingresar un conjunto de números reales en un vector. No se conoce la cantidad exacta de datos, pero se sabe que no superan los 100. No se conocen cotas para los datos. Informar el valor máximo, la cantidad de veces que aparece y la/s posición/es que ocupa.

2) Ingresar dos vectores a y b de igual dimensión n. El programa debe devolver el producto escalar p, es decir, $p = \sum(a[i] * b[i])$, para $0 \leq i \leq n-1$

3) Una empresa vende productos de clase A, B, C, D, E y F. Se pide diseñar un programa que permita ingresar la cantidad de productos vendidos de cada clase. Deberá existir una opción para poder las cantidades ordenar de mayor a menor y finalmente listarlas en forma ordenada indicando Clase y Cantidad. Para resolverlo es necesario utilizar dos vectores: uno para la cantidad de productos vendidos y otro para indicar en qué posición está cada clase de producto.

ii) Actividad de pares

Escribir un programa para simular el juego del CRAPS a \$1 por juego, donde el usuario comienza con una cifra de dinero (ingresada por él, máximo \$20) y termina cuando haya perdido todo su dinero o haya conseguido duplicar el mismo. Debe informársele al jugador el monto de dinero que posee luego de cada juego Nota: el juego de CRAPS tiene las siguientes reglas:

1. Si la primera tirada es 7 u 11, ganó.
2. Si la primera tirada es 2, 3 o 12, perdió.
3. Si en la primera tirada obtuvo otro resultado se sigue tirando hasta obtener un 7 (en cuyo caso se pierde) o nuevamente dicho número (en cuyo caso se gana.).

APENDICE B. PROBLEMAS RESUELTOS

Índice:

Ejemplo 1: multiplicar las cifras de un número positivo.

Ejemplo 2: determinar los mejores 4 promedios.

Ejemplo 3: juego de cartas

Ejemplo 1:

El problema que se plantea consiste en multiplicar las cifras de un número positivo.

a) Análisis Top-Down

Para ello, se hace la siguiente descomposición:

Obtener un número entero positivo numPosit;

Calcular el producto p de las cifras de numPosit;

Mostrar p.

A su vez, Calcular el producto p de las cifras de numPosit podría descomponerse así:

Repetir

Tomar una cifra c de numPosit;

Multiplicar por c el producto acumulado de las restantes cifras hasta que no queden más cifras

El programa consta de un Procedimiento **LeerNumPos**, que lee un número positivo distinto de cero y de una Función **ProductoCifras** que calcula el producto de sus cifras. Para ello, la función dispone de un Procedimiento local **QuitaUltimaCifra** que, dado un número, elimina su última cifra y la devuelve junto con el número modificado.

ProductoCifras llama repetidamente a **QuitaUltimaCifra** obteniendo las sucesivas cifras del número y calculando su producto. El número se va reduciendo hasta que no quedan más cifras, con lo que analizan las repeticiones. En la Figura 1 puede verse la estructura jerárquica del programa.

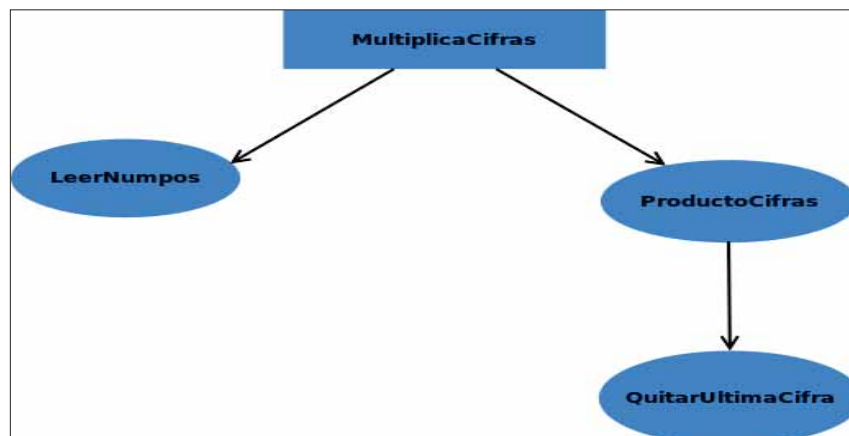


Figura 1. Estructura del Programa MultiplicaCifras

b) Solución

<pre> Program MultiplicaCifras ; var numPosit: integer; Procedure LeerNumPos (var n: integer); {Solicita un numero hasta obtener uno positivo} begin repeat write('Escriba un entero mayor que cero: '); readln(n) until n > 0 end; {LeerNumPos} Function ProductoCifras (numero: integer): integer; {Elimina la ultima cifra de numero y la almacena en ultima} var acumProd, cifrUnidades: integer; Procedure QuitarUltimaCifra (var n, ultima: integer); begin ultima:= n mod 10; n:= n div 10 end; {QuitarUltimaCifra} begin acumProd:= 1; repeat QuitarUltimaCifra(numero,cifrUnidades); acumProd:= acumProd * cifrUnidades until numero = 0; ProductoCifras:= acumProd end; {ProductoCifras} begin LeerNumPos (numPosit); writeln ('El producto de las cifras de ', numPosit,' es ', ProductoCifras (numPosit)) end. </pre>	<p>El programa utiliza una variable global numPosit, que es pasada como parámetro a los dos subprogramas, a LeerNumPos por referencia y a ProductoCifras por valor.</p> <p>El procedimiento LeerNumPos le da su valor inicial, que después se pasa a ProductoCifras para su procesamiento.</p> <p>La función ProductoCifras tiene dos variables locales acumProd (en la que se acumula el producto de las cifras del número) y cifrUnidades (donde se anotan los valores de dichas cifras). Estas variables se utilizan solamente dentro de ProductoCifras, no teniendo ninguna utilidad fuera de la función, por lo que se han declarado como locales. Quedan ocultas al programa principal y al procedimiento LeerNumPos, pero son visibles desde el procedimiento QuitaUltimaCifra para el que son no locales.</p> <p>La función tiene también un parámetro por valor llamado numero, a través del cual se recibe el dato inicial, y que actúa además como variable, que se va modificando al ir quitándole cifras.</p> <p>El procedimiento QuitaUltimaCifra se ha definido dentro del procedimiento ProductoCifras, por lo que es local a esta función y tiene sentido solo dentro de la misma, que es donde se necesita, quedando oculto al resto del programa. No tiene variables locales, pero utiliza los parámetros por referencia n y ultima.</p> <p>En el primero se recibe el número sobre el que operar y devuelve el número sin la última cifra, actuando como dato y como resultado. La cifra de las unidades se devuelve en el parámetro ultima, que representa solo este resultado. Ambos son locales por ser parámetros, y se pasan por referencia para enviar los resultados a ProductoCifras.</p> <p>Hay que observar que el identificador n se ha utilizado como parámetro de LeerNumPos y de QuitaUltimaCifra, sin provocar ningún conflicto.</p>
--	--



[VOLVER](#)

Ejemplo 2:

A fin de año se tiene la lista de alumnos de un curso (no más de 40 alumnos, aunque no se sabe a priori cuantos son), con la información de cada uno:

- Nombre (hasta 20 caracteres)
- Nota parcial 1
- Nota parcial 2
- Nota parcial 3

Se desea conocer los nombres y promedios de los 4 mejores alumnos de dicha comisión. OBSERVACION: SE utilizará el promedio de notas para determinar los 4 mejores alumnos

a) Estructura de datos

Un vector de 40 componentes de tipo String[20] para los nombres de los alumnos
Una matriz de 40x3 componentes de tipo real para las 3 notas de cada alumno
Un vector de 40 componentes de tipo real para los promedios de las notas

b) Análisis Top-Down

Para ello, se hace la siguiente descomposición:

Ingresar las notas y nombres
Hallar los 4 mejores alumnos
Mostrar las notas y nombres de los 4 mejores alumnos

A su vez, para hallar los mejores alumnos, la descomposición podría ser así:

Calcular las notas promedio
Ordenar las notas de mayor a menor;

El programa consta de un Procedimiento Ingreso, que lee los nombres y notas de los alumnos. Para calcular los promedios dispone de un Procedimiento NotasPromedio. Luego el Procedimiento Ordena cada promedio y acompaña ese mismo orden los nombres de los alumnos para que cada alumno conserve su promedio y su nombre. De esta manera en los primeros 4 lugares se encontrará la solución al problema. Finalmente el programa tiene un Procedimiento Muestra que presenta los promedios y nombres de los mejores 4 alumnos. En la Figura 2 puede verse la estructura jerárquica del programa.

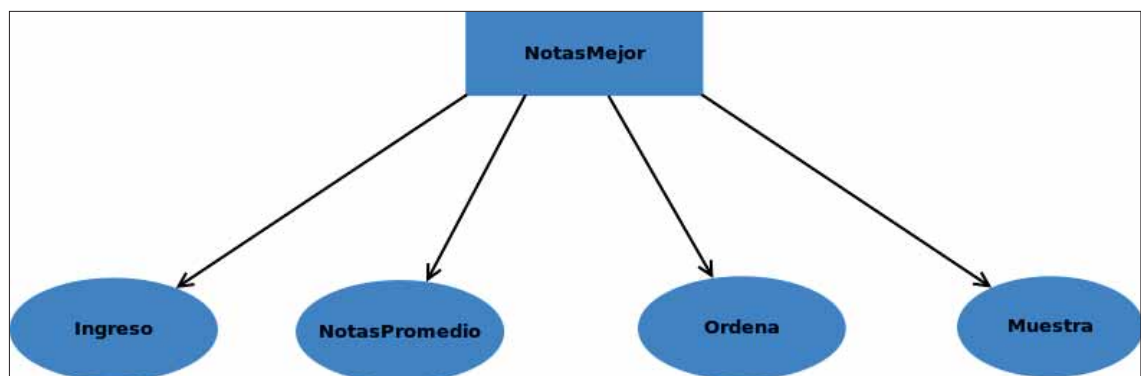
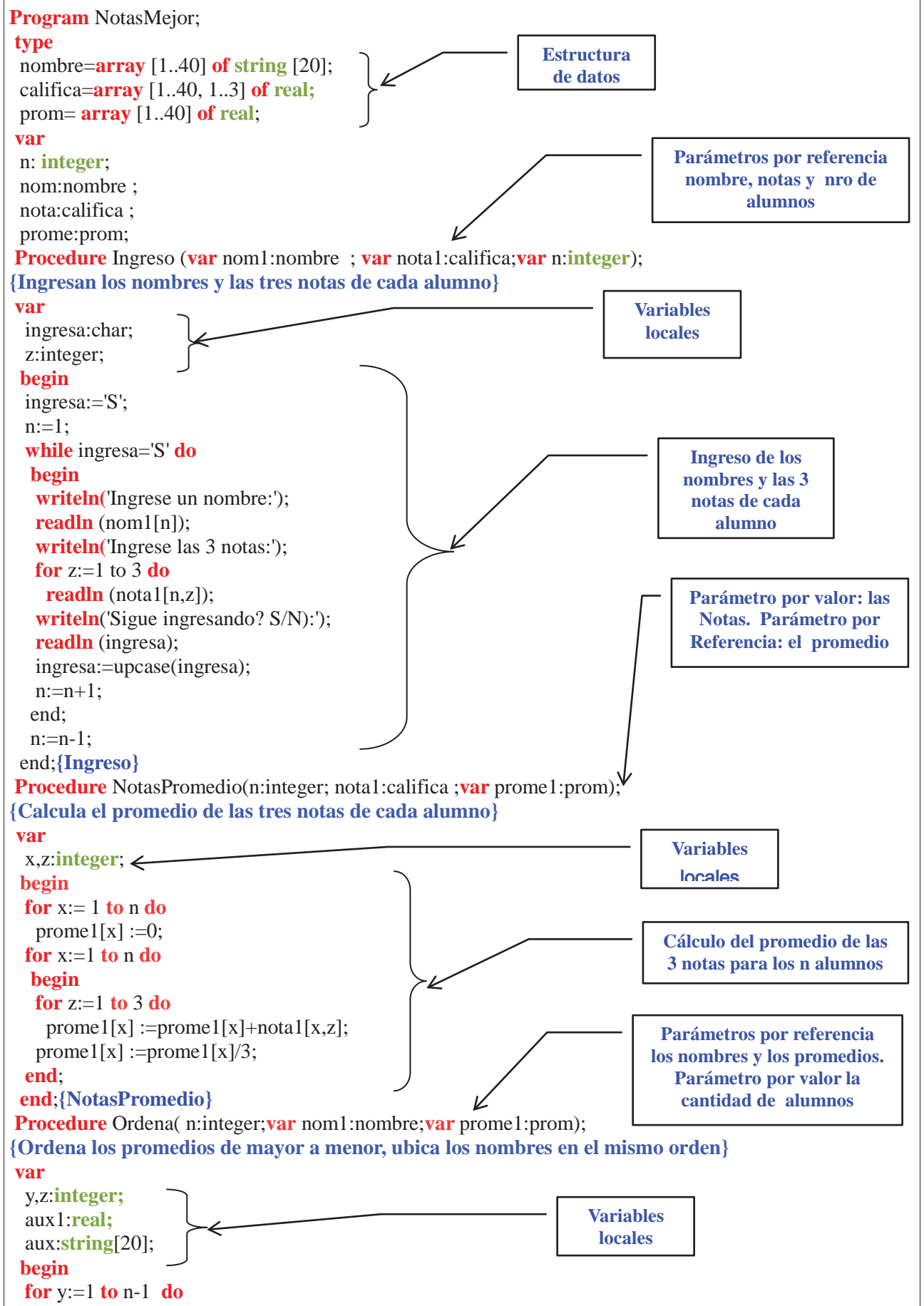
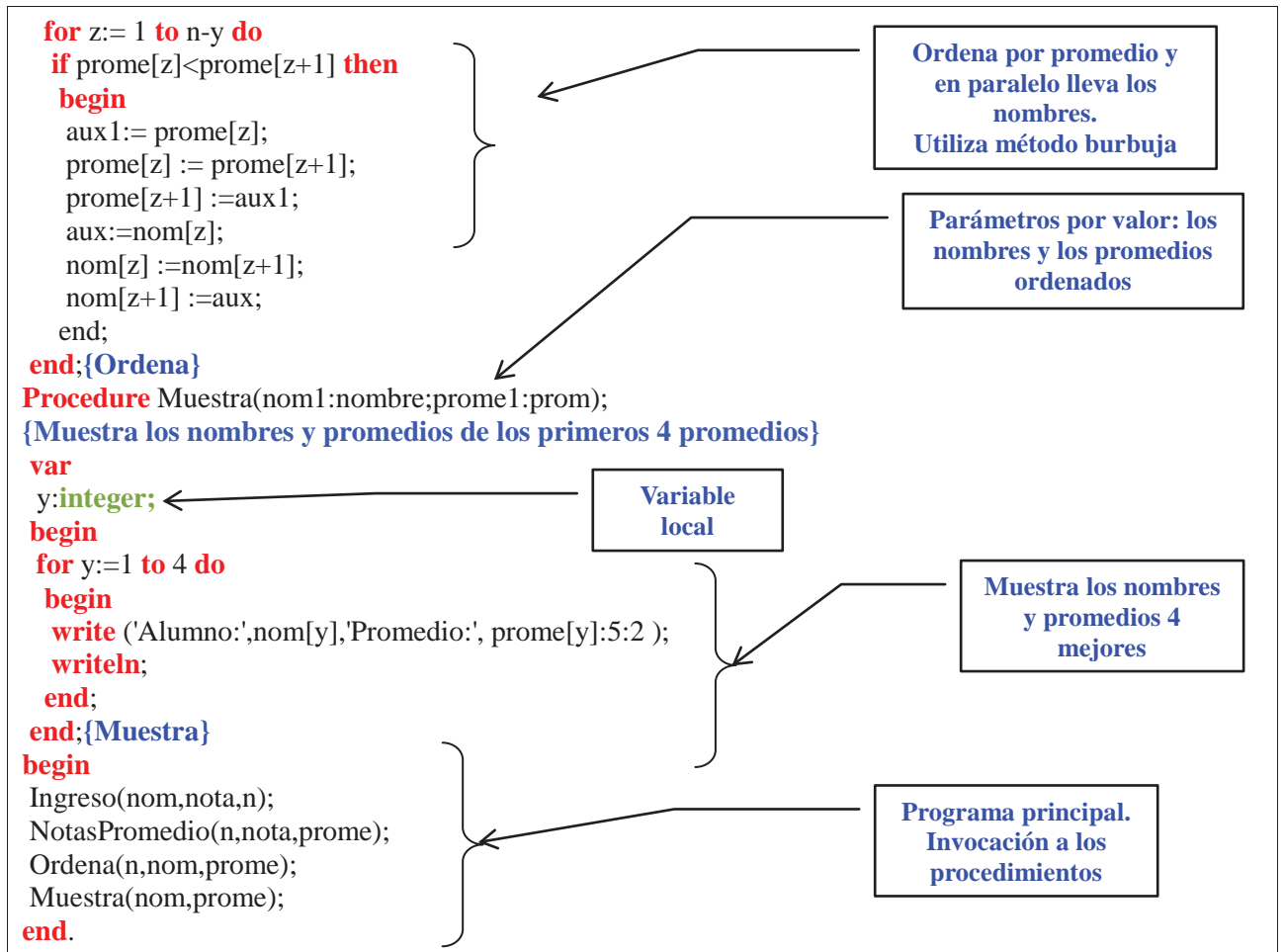


Figura 2. Estructura Jerárquica del Programa NotasMejor






VOLVER

Ejemplo 3

N jugadores reparten 4 cartas (mazo español) para cada uno. Se desea evaluar si alguno de ellos gana la mano.

Las cartas deben ingresarse agrupadas de 4 en 4 (por jugador y en el orden de la mano), cada una representada por un carácter (C, B, O, E) y un número de 1 a 12.

Gana quien tiene en sus cartas:

(en el orden que se indica y si hay más de uno, el que está primero o sea la “mano”)

El máximo puntaje impar (de los que suman más de 24) y si no hay ninguno, las 4 cartas de igual palo y si no hay ninguno, la mano no tiene ganadores.

Se pide realizar un programa Pascal para:

- ✓ Ingresar la información de los jugadores en una estructura de datos adecuada
- ✓ Mostrar las cartas del ganador y el motivo, o el cartel “no hay ganadores”

a) Ejemplo :

Numero de Jugadores: 5 (N)

B	E	O	E	O
B	C	E	B	B
B	C	C	C	C
B	E	C	O	E

12	10	12	4	6
1	8	1	2	6
4	11	10	2	6
5	5	2	12	6

las cartas del ganador por palo son

B 12
 B 1
 B 4
 B 5

b) Estructura de datos

De acuerdo al ejemplo, definieron los siguientes tipos:

Una matriz de 4xN componentes de tipo char para almacenar el palo de las 4 cartas de los N jugadores

Una matriz de 4xN componentes de tipo integer para almacenar el número de las 4 cartas de los N jugadores

Una matriz de 4x4 componentes de tipo string[2] para almacenar el palo y el número de las 4 cartas del ganador

Un vector de N componentes de tipo entero para almacenar la suma de puntos

c) Análisis Top-Down

De acuerdo a la estructura de datos definida, el algoritmo podría descomponerse de la siguiente forma:

Ingresar las cartas: palo y número de cada jugador

Calcular el ganador

Si hay ganador

Mostrar las cartas: palo y número del ganador

sino

Mostrar un cartel indicando que no hay ganador

A su vez, para calcular el ganador, la descomposición podría ser así:

Calcular la suma de los puntos

Almacenar las cartas de aquellos jugadores que suman 24

Ordenar las cartas de acuerdo a la suma de puntos

Encontrar suma mayor impar

si no hay suma mayor impar

encontrar el jugador cuyas cartas son del mismo palo

sino

no hay ganador

El programa consta de un Procedimiento **IngresoCartas**, que lee las 4 cartas (palo y número) de los N jugadores. Para calcular el ganador dispone de un Procedimiento **CalculaGanador**.

El Procedimiento **CalculaGanador** cuenta con el Procedimiento **SumaPuntos** que calcula

la suma de los puntos de las 4 cartas de los N jugadores y luego almacena sólo las cartas (palo y número) que suman 24.

Luego el Procedimiento **CalculaGanador** llama a otro Procedimiento **Ordena**, el mismo coloca las cartas (palo y número) de mayor a menor de acuerdo a la suma de puntos.

Finalmente el Procedimiento **CalculaGanador** busca entre todos los jugadores (con suma mayor a 24) las cartas de mayor puntaje impar y si no hay busca la carta entre todos los jugadores del mismo palo. Puede ocurrir que no haya ningún jugador que cumpla las condiciones.

A partir de ello el programa principal, muestra el ganador (si existe) y sino indica que no hay ganador.

En la Figura 3 puede verse la estructura jerárquica del programa.

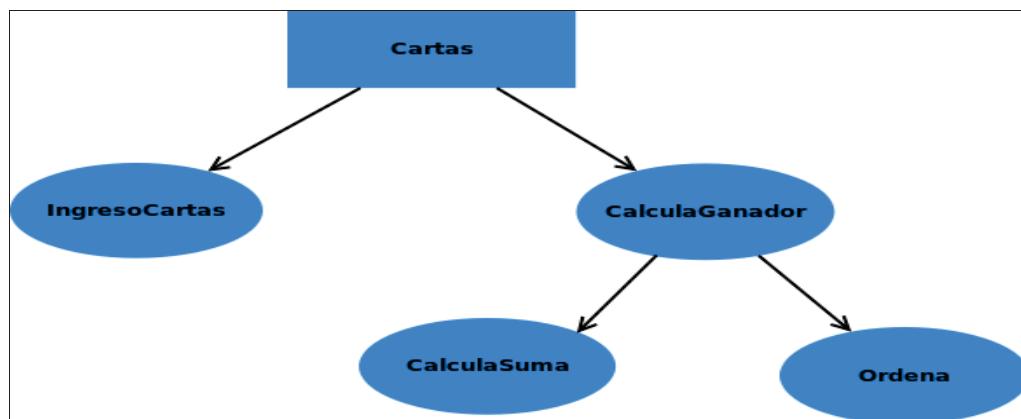
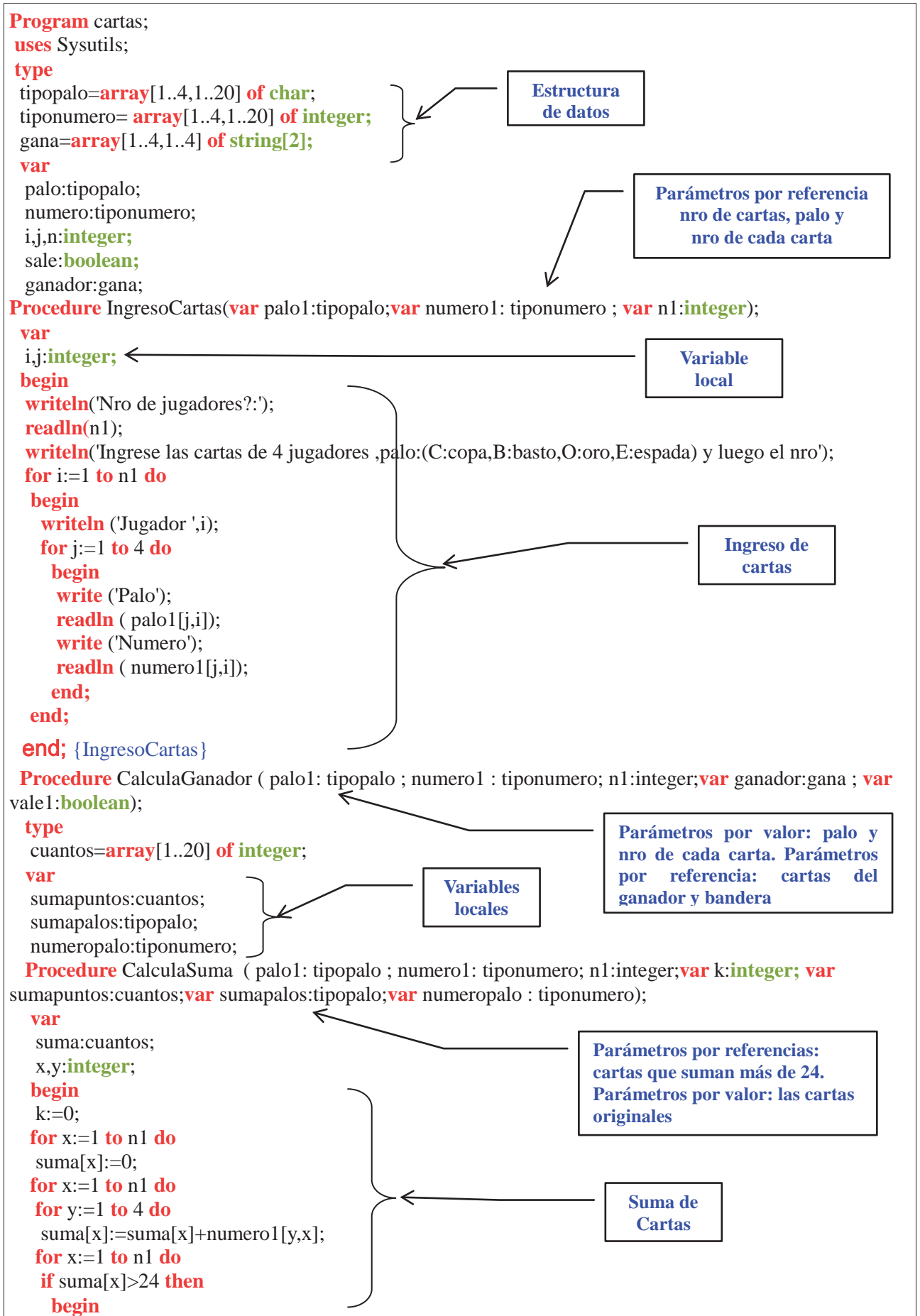
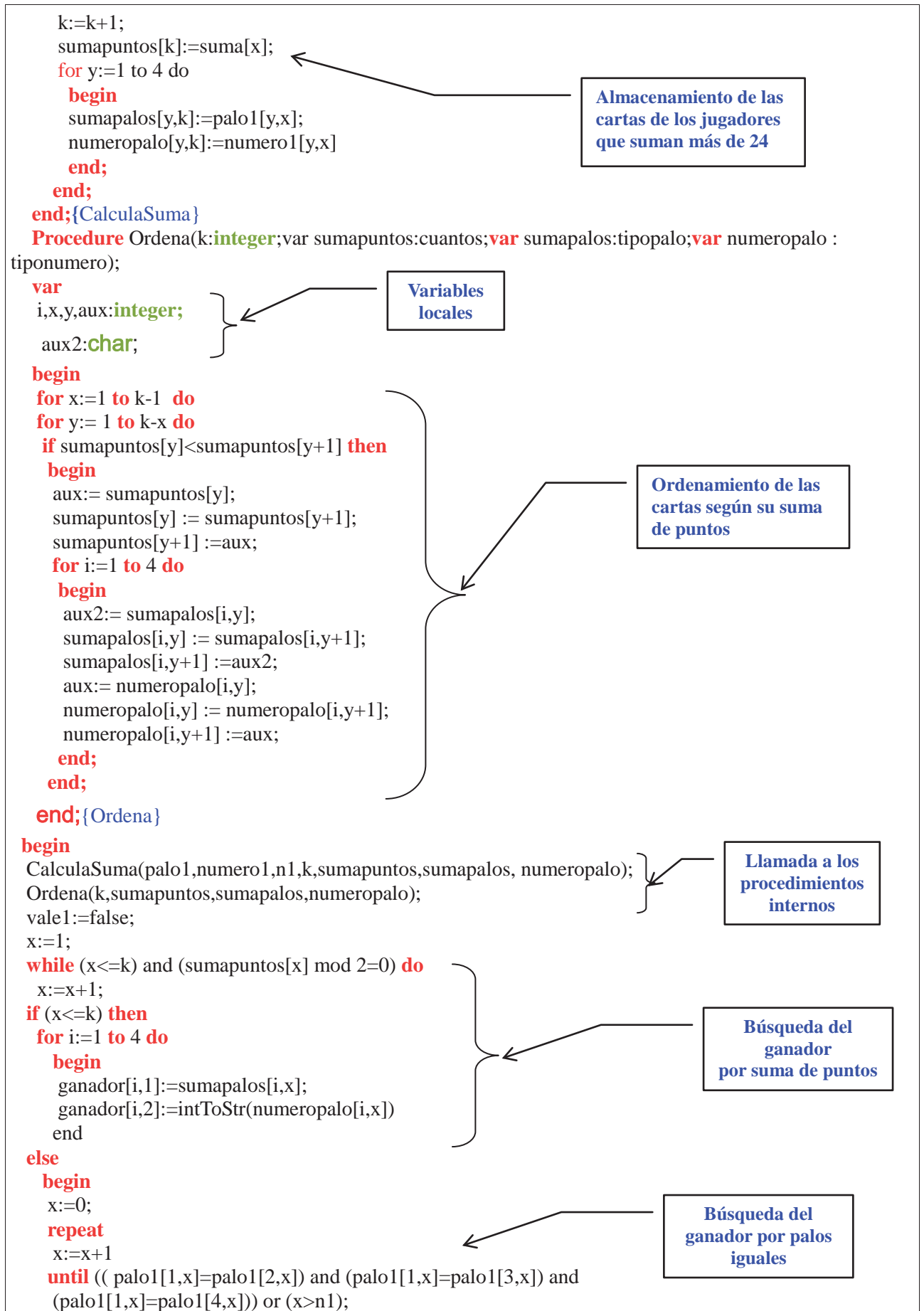


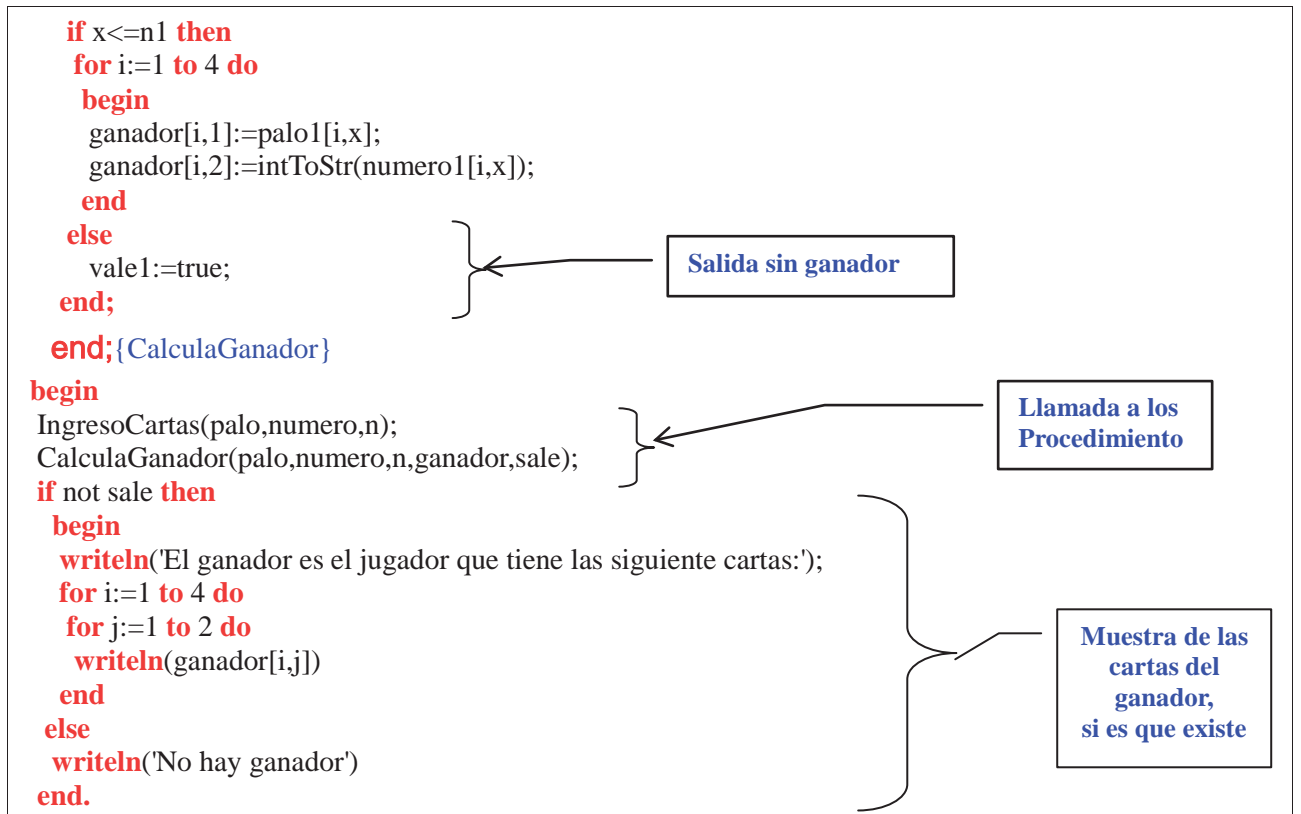
Figura 3. Estructura Jerárquica del Programa Cartas

c) Solución

La solución resulta sumamente compleja por el tipo de estructura de datos elegida (que es la conocida hasta el momento). Puede mejorarse eligiendo trabajar con arreglos de registros.







VOLVER

APENDICE C. EVALUACION DE EXPERTOS

C.1. Plantilla GEHOA (Guía de evaluación heurística para Objetos de Aprendizaje)

1. Criterios pedagógicos para evaluar objetos de aprendizaje.	
Nro	Categoría Significatividad psicológica
MOTIVACIÓN	
1	El contenido es relevante para los objetivos profesionales o personales e intereses del estudiante.
2	Hay elementos en el OA que suponen captar el interés del alumno pero resultan ser una distracción que interfiere con el aprendizaje.
COMPETENCIAS	
3	El OA está orientado al desarrollo de competencias genéricas (transversales) y/o específicas
4	El OA propone diferentes contenidos/actividades para cada tipo/nivel de competencia de alumno.
INTERACCIÓN E INTERACTIVIDAD	
5	Se propicia el intercambio con el tutor y/o en forma grupal.
6	Se induce a la participación directa por parte de los estudiantes en diversas actividades.
CONOCIMIENTOS PREVIOS	
7	La profundidad es adecuada según los conocimientos previos y el nivel de complejidad que el estudiante es capaz de comprender
8	Se fomenta la capacidad de relacionar conceptos ya aprendidos con los nuevos conceptos.
INNOVACIÓN Y AUTONOMÍA	
9	Se promueve la creación de nuevas ideas y la búsqueda de nuevos procedimientos /técnicas/métodos para la resolución de problemas o tareas.
10	El OA contiene pautas, guías o actividades que fomentan el desarrollo de habilidades metacognitivas y estrategias de aprendizaje que le permita al estudiante planificar, regular y evaluar su propia actividad intelectual.
COMENTARIOS GENERALES (Si considera que el objeto puede ser reutilizado en otras áreas o asignaturas, dé algunos ejemplos)	
2. Criterios pedagógicos para evaluar objetos de aprendizaje.	
Categoría Significatividad lógica	
OBJETIVOS	
1	Se indica lo que se espera sea aprendido.
2	Los objetivos están correctamente formulados.

3	Los objetivos son factibles de ser alcanzados.
	CONTENIDOS
4	El tiempo de duración estimado en el desarrollo del OA es excesivo.
5	Presenta información suficiente y adecuada al nivel educativo.
6	Los contenidos son coherentes con los objetivos propuestos.
7	Presenta el material de apoyo en un único formato (texto. audio. etc).
8	Permite interactuar con el contenido a través de enlaces.
9	Proporciona información complementaria para ayudar a los alumnos que deseen profundizar sus conocimientos.
10	Cuida que la información que presenta sea confiable (datos exactos. referencias bibliográficas. etc.).
11	El contenido no presenta errores, sesgos u omisiones que pudiera confundir o llevar a errores.
12	Se respetan los derechos de propiedad intelectual.
13	Los contenidos se introducen en forma secuenciada para ayudar a una mejor comprensión del mismo.
14	El lenguaje empleado en los contenidos es pertinente a los objetivos y al nivel educativo.
15	Se utilizan múltiples estrategias de enseñanza: resúmenes, ilustraciones, señalizaciones, gráficos, mapas conceptuales. organizadores textuales
	ACTIVIDADES
16	Las actividades del OA complementan las del curso presencial.
17	Se presentan variadas actividades: resolución de problemas, estudio de caso, glosario, mapa conceptual, etc.
18	Presenta actividades de evaluación y práctica.
19	La mayor parte de las actividades se corrigen adecuadamente de manera automática.
20	En las actividades no existe posibilidad de que el estudiante reciba feedback personalizado relativo a la calidad o corrección de sus respuestas.
	COMENTARIOS GENERALES
	3. Criterios técnicos para evaluar objetos de aprendizaje
	Diseño de interfaz
	LENGUAJE DE LOS USUARIOS
1	No se rompen los párrafos ni la continuidad de las ideas que se exponen en ellos.
2	Marca bloques de contenido a través de títulos o epígrafes
3	Las imágenes aclaran la información textual.
4	La presencia de las imágenes es superflua.
5	Las animaciones atraen la atención del usuario para destacar puntos claves e ideas significativas.
6	La presencia de audio es superflua. no proporciona algún tipo de valor añadido.
7	Los videos proporcionan algún tipo de valor añadido
	CONTROL Y LIBERTAD PARA EL USUARIO
8	Existe un vínculo que permite volver a la página de inicio o retroceder en una acción.
9	La interfaz no requiere tecnologías que necesitan versiones actualizadas de los navegadores o plug-ins externos. Si esto no es así. se describen los requisitos informáticos necesarios.
10	En los recursos multimediales (imágenes. sonido. video) se indica el tiempo de descarga
11	Las animaciones pueden ser controladas por el usuario (avanzar, detener,etc.)

	DISEÑO ESTÉTICO Y MINIALISTA
12	Las fuentes son legibles y tienen un tamaño adecuado
13	Los colores y tipos de letras aportan información por sí mismos.
14	Las fuentes utilizan colores con suficiente contraste con el fondo
15	Las imágenes son poco claras. no se ha cuidado su resolución.
16	Las animaciones proporcionan algún tipo de valor añadido.
17	En los videos. la imagen y el audio se presentan de forma clara.
	COMENTARIOS GENERALES
	4. Criterios técnicos para evaluar objetos de aprendizaje
	Estructura y navegación
	VISIBILIDAD DEL ESTADO DEL SISTEMA
1	La página de inicio aclara al usuario dónde se encuentra y el objetivo del sitio .
2	La página de Inicio presenta las principales áreas de contenido del sitio con enlaces para acceder a ella.
3	El usuario sabe dónde se encuentra en todo momento.
	CONSISTENCIA Y ESTÁNDARES
4	El diseño es inconsistente en todas las pantallas (tamaños. colores. iconos. tipos de letra. etc.).
5	Las áreas de navegación superior. laterales. herramientas de búsqueda y controles (botones. radio buttons. ...) siguen los estándares comunes del mercado.
6	Las mismas acciones llevan a los mismos resultados
	RECONOCIMIENTO MAS QUE MEMORIA
7	Utiliza el hipertexto para dividir información extensa en múltiples páginas
8	El sitio posee una estructura flexible y familiar que permite al usuario controlar su navegación.
9	Los títulos son breves y descriptivos.
10	Los enlaces son fácilmente reconocibles. están claramente diferenciados
	FLEXIBILIDAD Y EFICIENCIA DE USO
11	Se utiliza un diseño líquido para que la página se adapte a las diferentes resoluciones posibles que puede tener un usuario.
	NAVEGACION VISIBLE
12	Las pantallas dedican en gran parte espacio al contenido.
13	La información visible es la única esencial para realizar la acción. Las páginas no contienen información que es irrelevante o raramente necesaria.
	ERRORES
14	Cuando se produce un error. se informa de forma clara al usuario de lo ocurrido y de cómo solucionar el problema.
	COMENTARIOS GENERALES

Tabla C.1. Instrumento GEHOA

C.2. Resultados de la evaluación heurística

1. Criterios pedagógicos para evaluar objetos de aprendizaje.		valoración				
Nro	Categoría Significatividad psicológica	Exp_1	Exp_2	Exp_3	Exp_4	Exp_5
1	El contenido es relevante para los objetivos profesionales o personales e intereses del estudiante.	0	2	0	0	1
2	Hay elementos en el OA que suponen captar el interés del alumno pero resultan ser una distracción que interfiere con el aprendizaje.	2	0	0	1	0
3	El OA está orientado al desarrollo de competencias genéricas (transversales) y/o específicas.	0	0	0	0	0
4	El OA propone diferentes contenidos/actividades para cada tipo/nivel de competencia de alumno.	4	0	2	4	1
5	La profundidad es adecuada (según los conocimientos previos y el nivel de complejidad que el estudiante es capaz de comprender).	0	0	0	0	0
6	Se propicia el intercambio con el tutor y/o en forma grupal.	1	0	0	0	0
7	Se induce a la participación directa por parte de los estudiantes en diversas actividades.	2	1	0	0	0
8	Se fomenta la capacidad de relacionar conceptos ya aprendidos con los nuevos conceptos.	1	2	0	0	1
9	Se promueve la creación de nuevas ideas y la búsqueda de nuevos procedimientos/técnicas/métodos para la resolución de problemas o tareas.	0	0	0	4	0
10	El OA contiene pautas, guías o actividades que fomentan el desarrollo de habilidades metacognitivas y estrategias de aprendizaje que le permita al estudiante planificar, regular y evaluar su propia actividad intelectual.	1	1	0	0	1
	PUNTUACIÓN FINAL	11	6	2	9	4
	PROMEDIO	1.38	0.67	0.20	1.29	0.40
	DESVIO ESTANDAR	1.30	0.87	0.63	1.89	0.52
2. Criterios pedagógicos para evaluar objetos de aprendizaje.						
	Categoría Significatividad lógica					
1	Se indica lo que se espera sea aprendido.	0	0	0	0	0
2	Los objetivos están correctamente formulados.	0	0	0	0	0
3	Los objetivos son factibles de ser alcanzados.	0	1	0	0	0
4	El tiempo de duración estimado en el desarrollo del OA es excesivo.	1	0	0	4	4
5	Presenta información suficiente y adecuada al nivel educativo.	1	0	0	0	0
6	Los contenidos son coherentes con los objetivos propuestos.	0	0	0	0	0
7	Presenta el material de apoyo en un único formato (texto, audio, etc).	0	0	0	0	0
8	Permite interactuar con el contenido a través de enlaces.	1	0	0	0	0
9	Proporciona información complementaria para ayudar a los alumnos que deseen profundizar sus conocimientos.	2	2	0	4	0
10	Cuida que la información que presenta sea confiable (datos exactos, referencias bibliográficas, etc.).	3	1	0	0	0
11	El contenido no presenta errores, sesgos u omisiones que pudiera confundir o llevar a errores.	2	0	0	1	0

12	Se respetan los derechos de propiedad intelectual.	0	0	0	0	0
13	Los contenidos se introducen en forma secuenciada para ayudar a una mejor comprensión del mismo.	0	0	0	0	0
14	El lenguaje empleado en los contenidos es pertinente a los objetivos y al nivel educativo.	1	0	0	0	0
15	Las actividades del OA complementan las del curso presencial.	0	0	0	0	0
16	Se utilizan múltiples estrategias de enseñanza: resúmenes, ilustraciones, señalizaciones, gráficos, mapas conceptuales, organizadores textuales.	2	0	0	0	0
17	Se presentan variadas actividades : resolución de problemas, estudio de caso, glosario, mapa conceptual, etc.	1	0	1	0	1
18	Presenta actividades de evaluación y práctica.	1	0	0	0	0
19	La mayor parte de las actividades se corrigen adecuadamente de manera automática.	1	1	0	0	2
20	En las actividades no existe posibilidad de que el estudiante reciba feedback personalizado relativo a la calidad o corrección de sus respuestas.	1	4	0	0	0
	PUNTUACIÓN FINAL	17	9	1	9	7
	PROMEDIO	0.35	0.45	0.05	0.47	0.35
	DESVIO ESTANDAR	0.99	1.27	0.22	1.26	0.99
	3. Criterios técnicos para evaluar objetos de aprendizaje					
	Categoría Diseño de interfaz					
1	No se rompen los párrafos ni la continuidad de las ideas que se exponen en ellos.	1	0	0	0	0
2	Marca bloques de contenido a través de títulos o epígrafes	1	0	0	0	0
3	Las fuentes son legibles y tienen un tamaño adecuado	1	1	2	0	0
4	Los colores y tipos de letras aportan información por sí mismos.	2	1	0	0	1
5	Las fuentes utilizan colores con suficiente contraste con el fondo	1	1	2	1	0
6	La interfaz no requiere tecnologías que necesitan versiones actualizadas de los navegadores o plug-ins externos. Si esto no es así. se describen los requisitos informáticos necesarios.	0	0	0	1	0
7	Las imágenes son poco claras. no se ha cuidado su resolución.	1	3	0	0	0
8	Las imágenes aclaran la información textual.	1	0	0	0	0
9	La presencia de las imágenes es superflua.	1	0	0	0	0
10	En los recursos multimediales (imágenes. sonido. video) se indica el tiempo de descarga.	0	0	0	0	4
11	Las animaciones proporcionan algún tipo de valor añadido.	1	0	0	4	0
12	Las animaciones atraen la atención del usuario para destacar puntos claves e ideas significativas.	2	1	0	0	0
13	Las animaciones pueden ser controladas por el usuario (avanzar, detener, etc.)	0	0	0	0	0
14	La presencia de audio es superflua. no proporciona algún tipo de valor añadido.	1	0	0	0	0
15	Los videos proporcionan algún tipo de valor añadido.	1	0	0	2	0
16	En los videos. la imagen y el audio se presentan de forma clara.	0	1	2	0	1
17	Existe un vínculo que permite volver a la página de inicio o retroceder en una acción.	2	0	0	0	0

	PUNTUACIÓN FINAL	16	8	6	8	6
	PROMEDIO	1.00	0.47	0.35	0.47	0.38
	DESVIO ESTANDAR	0.63	0.80	0.79	1.07	1.02
	4. Criterios técnicos para evaluar objetos de aprendizaje					
	Categoría Estructura y navegación					
1	La página de inicio aclara al usuario dónde se encuentra y el objetivo del sitio.	1	1	0	0	0
2	La página de Inicio presenta las principales áreas de contenido del sitio con enlaces para acceder a ella.	1	0	0	4	0
3	Utiliza el hipertexto para dividir información extensa en múltiples páginas.	1	0	0	0	0
4	El diseño es inconsistente en todas las pantallas (tamaños. colores. iconos. tipos de letra, etc.).	1	0	0	0	0
5	El sitio posee una estructura flexible y familiar que permite al usuario controlar su navegación.	1	0	0	4	0
6	Las áreas de navegación superior, laterales, herramientas de búsqueda y controles (botones. radio buttons) siguen los estándares comunes del mercado.	1	0	0	0	0
7	Los títulos son breves y descriptivos.	0	0	0	0	1
8	Los enlaces son fácilmente reconocibles están claramente diferenciados.	1	1	0	0	0
9	Las mismas acciones llevan a los mismos resultados.	0	0	0	0	0
10	Cuando se produce un error. se informa de forma clara al usuario de lo ocurrido y de cómo solucionar el problema.	0	1	1	0	0
11	El usuario sabe dónde se encuentra en todo momento.	0	2	0	0	0
12	Las pantallas dedican en gran parte espacio al contenido.	0	0	0	0	0
13	La información visible es la única esencial para realizar la acción. Las páginas no contienen información que es irrelevante o raramente necesaria.	0	0	0	0	0
14	Se utiliza un diseño líquido para que la página se adapte a las diferentes resoluciones posibles que puede tener un usuario.	1	2	0	0	0
	PUNTUACIÓN FINAL	8	7	1	8	1
	PROMEDIO	0.62	0.54	0.07	0.67	0.08
	DESVIO ESTANDAR	0.51	0.78	0.27	1.56	0.29

Tabla C.2. Puntajes de expertos en el instrumento GEHOA

C.3. Estadísticos comparativos de los puntajes

		Significatividad psicológica	Significatividad lógica	Diseño de interfaz	Estructura y navegación
Consultor de exp. de usuario	Puntuación final	11	17	16	8
	Promedio	1.38	0.85	1.00	0.62
	Desvío estándar	1.30	0.74	0.63	0.51
Consultor de exp. de usuario y experto en contenido	Puntuación final	6	9	8	7
	Promedio	0.67	0.45	0.47	0.54
	Desvío estándar	0.87	1.27	0.80	0.78
Experto de software relacionado con las tecnologías Web	Puntuación final	2	1	6	1
	Promedio	0.20	0.05	0.35	0.07
	Desvío estándar	0.63	0.22	0.79	0.27
Experto en exp. de usuario	Puntuación final	9	9	8	8
	Promedio	1.29	0.47	0.47	0.67
	Desvío estándar	1.89	1.26	1.07	1.56
Especialista del área del contenido educativo	Puntuación final	4	7	6	1
	Promedio	0.40	0.35	0.38	0.08
	Desvío estándar	0.52	0.99	1.02	0.29
Puntuación total		32	43	44	25
Puntaje máximo catástrofe x 5 expertos		200	400	340	280
Puntaje Promedio x categoría		6.40	8.60	8.80	5.00
Desvio estándar x categoría		2.59	2.19	1.96	1.85
Puntaje catástrofe x categoría		40	80	68	56

Tabla C.3. Estadísticos comparativos en cada categoría para cada experto

C.4. Puntajes ponderados

	Puntaje ponderado
Consultor de experiencia de usuario	13.67
Consultor de experiencia de usuario y experto en contenido	7.77
Experto de software relacionado con las tecnologías Web	2.56
Experto en experiencia de usuario	8.49
Especialista del área del contenido educativo	4.85
Puntaje ponderado catástrofe	64.59

Tabla C.4. Puntajes ponderados

APENDICE D. EVALUACION POR ESTUDIANTES

D.1. Instrumento CUSEOA (Cuestionario de satisfacción de estudiantes de un Objeto de Aprendizaje)

Luego de utilizar el recurso, marca con una “X” la posición donde se encuentra tu opinión:

	1	2	3	4	5	6	7		No sabe/No contesta
difícil								fácil	
frustrante								satisfactorio	
aburrido								ameno	
rígido								flexible	

Tabla D.1. Parte A del instrumento CUSEOA

	Totalmente en Desacuerdo	En Desacuerdo	Indiferente	De acuerdo	Totalmente de acuerdo
Los objetivos indican lo que se espera que sea aprendido.					
El nivel de dificultad de los contenidos fue elevado para mis conocimientos previos.					
El material teórico me ayudó a comprender los conceptos.					
Las actividades han sido claras y significativas para mi aprendizaje.					
El sistema informa sobre mi progreso.					
Las pistas sobre los errores cometidos son inútiles.					
El texto es conciso y preciso.					
Los títulos son inadecuados , no se sabe cual es la acción que se debe realizar.					
Las imágenes empleadas me ayudaron a aclarar los contenidos.					
Me encontré perdido cuando recorría el recurso, no sabía dónde me encontraba.					
Los videos y las animaciones me ayudaron a aclarar los contenidos.					
La información está mal organizada.					
En general, los colores y el diseño de todo el recurso son adecuados.					
Recomendaría este recurso a otra persona.					

Tabla D.2. Ítems Parte B del instrumento CUSEOA

Para finalizar, te pedimos que nos des tu opinión sobre el cuestionario:

D.2. Puntajes parte a del instrumento CUSEOA

ID	difícil-fácil	frustrante-satisfactorio	aburrido-ameno	rígido-flexible
1	1	2	0	1
2	2	2	1	2
3	3	2	1	0
4	0	2	0	-2
5	1	3	3	-1
6	2	2	2	2
7	3	3	3	3
8	0	-3	-2	-2
9	-1	3	2	0
10	2	1	-1	0
11	-1	3	2	0
12	2	2	1	2
13	1	1	2	0
14	1	0	0	0
15	0	1	1	0
16	0	3	1	0
17	0	2	0	-2
18	3	1	3	3
19	1	2	0	0
20	2	2	1	2

Tabla D.3. Puntajes de la parte A

D.3. Valoración de ítems de la parte b del instrumento CUSEOA

ID	Los objetivos indican lo que se espera que sea aprendido	El nivel de dificultad de los contenidos fue elevado para mis conocimientos previos	El material teórico me ayudó a comprender los conceptos	Las actividades han sido claras y significativas para mi aprendizaje
1	De acuerdo	en Desacuerdo	De acuerdo	De acuerdo
2	De acuerdo	en Desacuerdo	De acuerdo	De acuerdo
3	De acuerdo	en Desacuerdo	Indiferente	De acuerdo
4	De acuerdo	Indiferente	Indiferente	Indiferente
5	De acuerdo	De acuerdo	Totalmente de acuerdo	Totalmente de acuerdo
6	De acuerdo	en Desacuerdo	De acuerdo	De acuerdo
7	De acuerdo	en Desacuerdo	De acuerdo	De acuerdo
8	Totalmente de acuerdo	Totalmente en Desacuerdo	Totalmente de acuerdo	Totalmente de acuerdo

9	De acuerdo	en Desacuerdo	De acuerdo	De acuerdo
10	De acuerdo	en Desacuerdo	De acuerdo	De acuerdo
11	De acuerdo	Totalmente en Desacuerdo	Indiferente	Indiferente
12	De acuerdo	Totalmente en Desacuerdo	De acuerdo	De acuerdo
13	De acuerdo	Indiferente	De acuerdo	Totalmente de acuerdo
14	De acuerdo	Indiferente	De acuerdo	De acuerdo
15	De acuerdo	en Desacuerdo	De acuerdo	De acuerdo
16	De acuerdo	en Desacuerdo	De acuerdo	De acuerdo
17	De acuerdo	Indiferente	De acuerdo	De acuerdo
18	Totalmente de acuerdo	Indiferente	Totalmente de acuerdo	Totalmente de acuerdo
19	De acuerdo	en Desacuerdo	De acuerdo	De acuerdo
20	De acuerdo	Indiferente	De acuerdo	De acuerdo

Tabla D.4a. Valoración de ítems parte B

ID	El sistema informa sobre mi progreso	Las pistas sobre los errores cometidos son inútiles	El texto es conciso y preciso	Los títulos son inadecuados , no se sabe cual es la acción que se debe realizar
1	Indiferente	en Desacuerdo	De acuerdo	en Desacuerdo
2	en Desacuerdo	Totalmente en Desacuerdo	Totalmente de acuerdo	De acuerdo
3	Indiferente	De acuerdo	Totalmente de acuerdo	Totalmente en Desacuerdo
4	en Desacuerdo	Totalmente en Desacuerdo	Totalmente de acuerdo	en Desacuerdo
5	De acuerdo	Indiferente	Totalmente de acuerdo	en Desacuerdo
6	Indiferente	Indiferente	De acuerdo	en Desacuerdo
7	De acuerdo	en Desacuerdo	De acuerdo	en Desacuerdo
8	Totalmente de acuerdo	Totalmente en Desacuerdo	Totalmente de acuerdo	Totalmente en Desacuerdo
9	Indiferente	en Desacuerdo	De acuerdo	en Desacuerdo
10	Indiferente	en Desacuerdo	Totalmente de acuerdo	en Desacuerdo
11	en Desacuerdo	De acuerdo	De acuerdo	Totalmente en Desacuerdo
12	en Desacuerdo	en Desacuerdo	Indiferente	en Desacuerdo
13	en Desacuerdo	en Desacuerdo	en Desacuerdo	Totalmente en Desacuerdo
14	en Desacuerdo	en Desacuerdo	en Desacuerdo	Indiferente
15	Indiferente	en Desacuerdo	De acuerdo	en Desacuerdo
16	Indiferente	Indiferente	Indiferente	en Desacuerdo
17	en Desacuerdo	Totalmente en Desacuerdo	Totalmente de acuerdo	Indiferente
18	De acuerdo	Indiferente	Totalmente de acuerdo	Totalmente en Desacuerdo
19	Indiferente	en Desacuerdo	De acuerdo	en Desacuerdo
20	De acuerdo	De acuerdo	Totalmente de acuerdo	Totalmente en Desacuerdo

Tabla D.4b. Valoración de ítems parte B

ID	Las imágenes empleadas me ayudaron a aclarar los contenidos	Me encontré perdido cuando recorría el recurso, no sabía dónde me encontraba	Los videos y las animaciones me ayudaron a aclarar los contenidos	La información está mal organizada
1	Totalmente de acuerdo	Totalmente en Desacuerdo	De acuerdo	en Desacuerdo
2	De acuerdo	Totalmente en Desacuerdo	De acuerdo	en Desacuerdo
3	De acuerdo	Totalmente en Desacuerdo	Indiferente	Totalmente en Desacuerdo
4	De acuerdo	Totalmente en Desacuerdo	De acuerdo	Totalmente en Desacuerdo
5	De acuerdo	en Desacuerdo	Totalmente de acuerdo	Totalmente en Desacuerdo
6	De acuerdo	en Desacuerdo	De acuerdo	en Desacuerdo
7	De acuerdo	Totalmente en Desacuerdo	De acuerdo	Totalmente en Desacuerdo

8	Totalmente de acuerdo	Totalmente en Desacuerdo	Totalmente de acuerdo	Totalmente en Desacuerdo
9	De acuerdo	en Desacuerdo	Totalmente de acuerdo	Totalmente en Desacuerdo
10	De acuerdo	en Desacuerdo	Totalmente de acuerdo	en Desacuerdo
11	De acuerdo	Totalmente en Desacuerdo	Totalmente de acuerdo	Totalmente en Desacuerdo
12	Indiferente	en Desacuerdo	De acuerdo	en Desacuerdo
13	Totalmente de acuerdo	Totalmente en Desacuerdo	Totalmente de acuerdo	Totalmente en Desacuerdo
14	De acuerdo	en Desacuerdo	Totalmente de acuerdo	en Desacuerdo
15	Totalmente de acuerdo	Totalmente en Desacuerdo	De acuerdo	en Desacuerdo
16	Totalmente de acuerdo	Totalmente en Desacuerdo	De acuerdo	en Desacuerdo
17	De acuerdo	Totalmente en Desacuerdo	De acuerdo	Totalmente en Desacuerdo
18	De acuerdo	Totalmente en Desacuerdo	De acuerdo	Totalmente en Desacuerdo
19	De acuerdo	Totalmente en Desacuerdo	De acuerdo	en Desacuerdo
20	Totalmente de acuerdo	en Desacuerdo	De acuerdo	Totalmente en Desacuerdo

Tabla D.4c. Valoración de ítems parte B

ID	En general, los colores y el diseño de todo el recurso son adecuados	Recomendaría este recurso a otra persona
1	Totalmente de acuerdo	Totalmente de acuerdo
2	Indiferente	De acuerdo
3	Indiferente	De acuerdo
4	Indiferente	De acuerdo
5	Indiferente	Totalmente de acuerdo
6	Totalmente de acuerdo	Totalmente de acuerdo
7	De acuerdo	De acuerdo
8	Totalmente de acuerdo	Totalmente de acuerdo
9	Indiferente	De acuerdo
10	Indiferente	De acuerdo
11	Totalmente de acuerdo	Indiferente
12	De acuerdo	De acuerdo
13	Totalmente de acuerdo	De acuerdo
14	Totalmente de acuerdo	De acuerdo
15	De acuerdo	De acuerdo
16	Indiferente	De acuerdo
17	Totalmente de acuerdo	De acuerdo
18	Totalmente de acuerdo	Totalmente de acuerdo
19	Totalmente de acuerdo	De acuerdo
20	De acuerdo	De acuerdo

Tabla D.4d. Valoración de ítems parte B

D.4. Distribución de frecuencias de ítems parte b

	Frecuencia	Porcentaje
De acuerdo	18	90,00%
Totalmente de acuerdo	2	10,00%
Total	20	100,00%

Tabla D.5. Los objetivos indican lo que se espera que sea aprendido

	Frecuencia	Porcentaje
Totalmente en Desacuerdo	3	15,00%
En desacuerdo	10	50,00%
Indiferente	6	30,00%
De acuerdo	1	5,00%
Total	20	100,00%

Tabla D.6. El nivel de dificultad de los contenidos fue elevado para mis conocimientos previos

	Frecuencia	Porcentaje
Indiferente	3	15,00%
De acuerdo	14	70,00%
Totalmente de acuerdo	3	15,00%
Total	20	100,00%

Tabla D.7. El material teórico me ayudó a comprender los conceptos

	Frecuencia	Porcentaje
Indiferente	2	10,00%
De acuerdo	14	70,00%
Totalmente de acuerdo	4	20,00%
Total	20	100,00%

Tabla D.8. Las actividades han sido claras y significativas para mi aprendizaje

	Frecuencia	Porcentaje
En desacuerdo	7	35,00%
Indiferente	8	40,00%
De acuerdo	4	20,00%
Totalmente de acuerdo	1	5,00%
Total	20	100,00%

Tabla D.9. El sistema informa sobre mi progreso

	Frecuencia	Porcentaje
Totalmente en Desacuerdo	4	20,00%
En desacuerdo	9	45,00%
Indiferente	4	20,00%
De acuerdo	3	15,00%
Total	20	100,00%

Tabla D.10. Las pistas sobre los errores cometidos son inútiles

	Frecuencia	Porcentaje
En desacuerdo	2	10,00%
Indiferente	2	10,00%
De acuerdo	7	35,00%
Totalmente de acuerdo	9	45,00%
Total	20	100,00%

Tabla D.11. El texto es conciso y preciso

	Frecuencia	Porcentaje
Totalmente en Desacuerdo	6	30,00%
En desacuerdo	11	55,00%
Indiferente	2	10,00%
De acuerdo	1	5,00%
Total	20	100,00%

Tabla D.12. Los títulos son inadecuados, no se sabe cual es la acción que se debe realizar

	Frecuencia	Porcentaje
Indiferente	1	5,00%
De acuerdo	13	65,00%
Totalmente de acuerdo	6	30,00%
Total	20	100,00%

Tabla D.13. Las imágenes empleadas me ayudaron a aclarar los contenidos

	Frecuencia	Porcentaje
Totalmente en Desacuerdo	13	65,00%
En desacuerdo	7	35,00%
Total	20	100,00%

Tabla D.14. Me encontré perdido cuando recorría el recurso, no sabía dónde me encontraba

	Frecuencia	Porcentaje
Indiferente	1	5,00%
De acuerdo	12	60,00%
Totalmente de acuerdo	7	35,00%
Total	20	100,00%

Tabla D.15. Los videos y las animaciones me ayudaron a aclarar los contenidos

	Frecuencia	Porcentaje
Totalmente en Desacuerdo	11	55,00%
En desacuerdo	9	45,00%
Total	20	100,00%

Tabla D.16. La información está mal organizada

	Frecuencia	Porcentaje
Indiferente	7	35,00%
De acuerdo	4	20,00%
Totalmente de acuerdo	9	45,00%
Total	20	100,00%

Tabla D.17. En general, los colores y el diseño de todo el recurso son adecuados

	Frecuencia	Porcentaje
Indiferente	1	5,00%
De acuerdo	14	70,00%
Totalmente de acuerdo	5	25,00%
Total	20	100,00%

Tabla D.18. Recomendaría este recurso a otra persona

D.5. Gráficos de sectores de ítems parte b

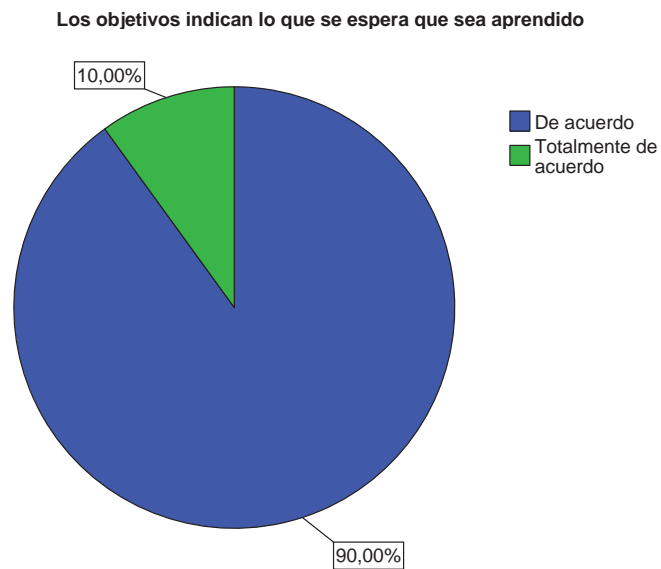


Gráfico D.1. Porcentaje de respuestas ítem 1

El nivel de dificultad de los contenidos fue elevado para mis conocimientos previos

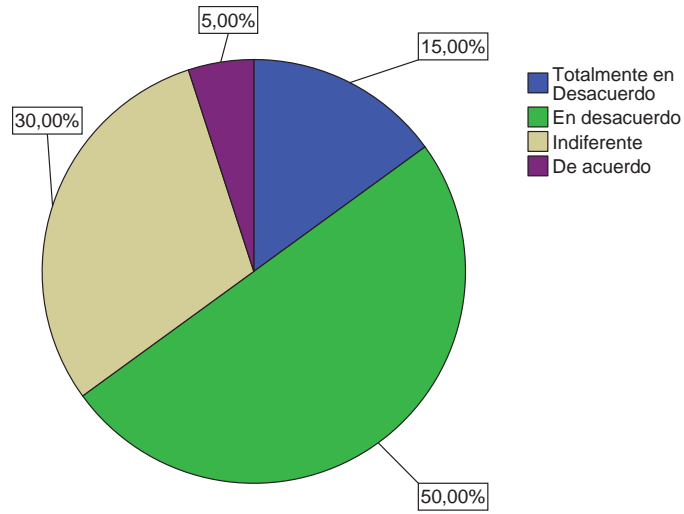


Gráfico D.2. Porcentaje de respuestas ítem 2

El material teórico me ayudó a comprender los conceptos

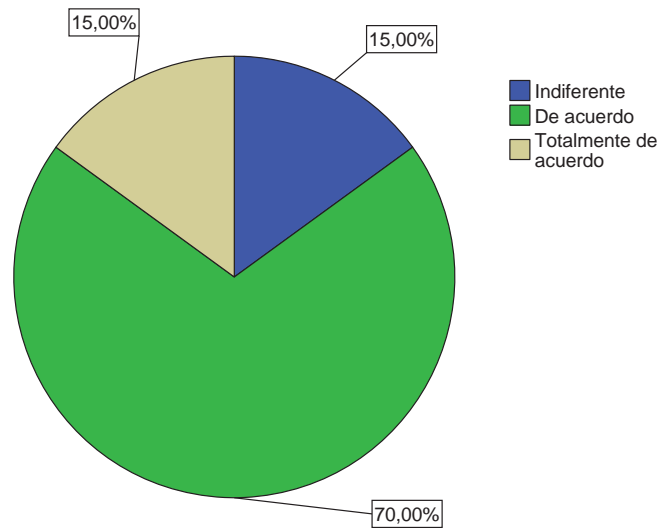


Gráfico D.3. Porcentaje de respuestas ítem 3

Las actividades han sido claras y significativas para mi aprendizaje

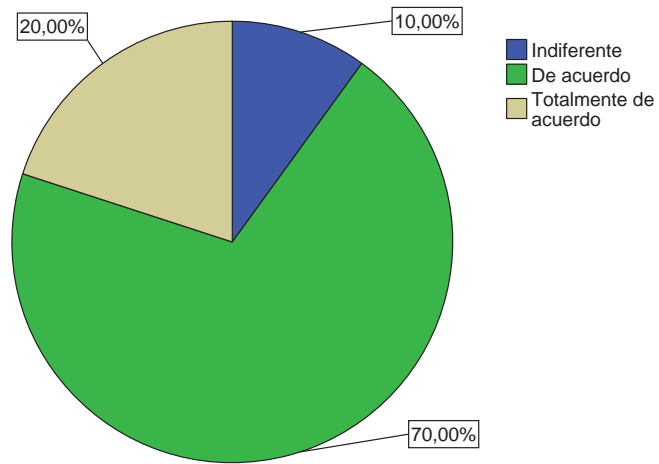


Gráfico D.4. Porcentaje de respuestas ítem 4

El sistema informa sobre mi progreso

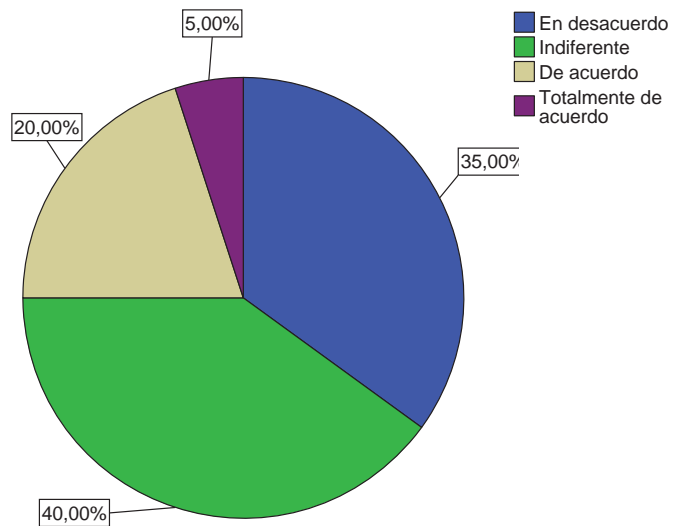


Gráfico D.5. Porcentaje de respuestas ítem 5

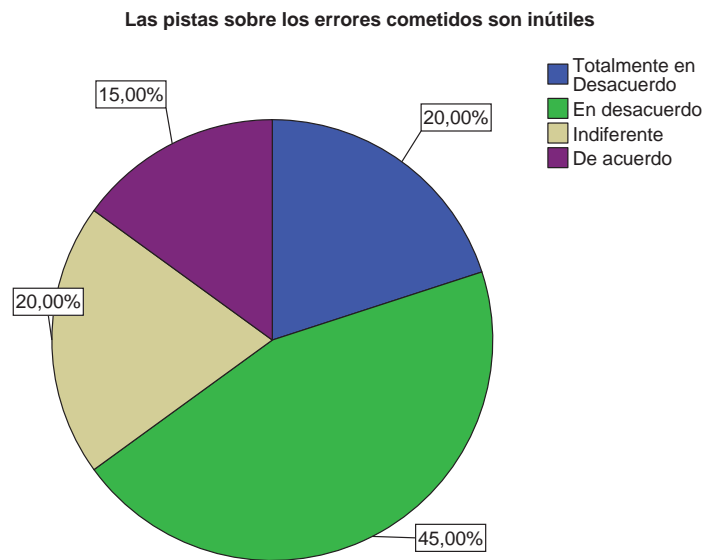


Gráfico D.6. Porcentaje de respuestas ítem 6

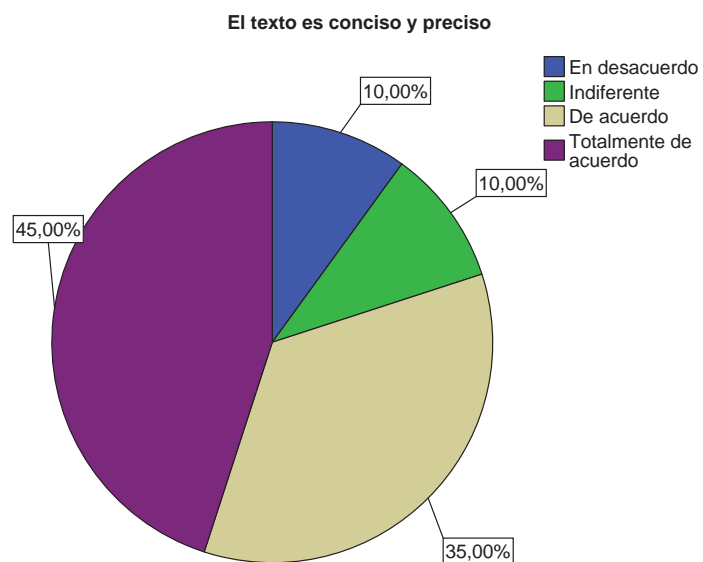


Gráfico D.7. Porcentaje de respuestas ítem 7

Los títulos son inadecuados , no se sabe cual es la acción que se debe realizar

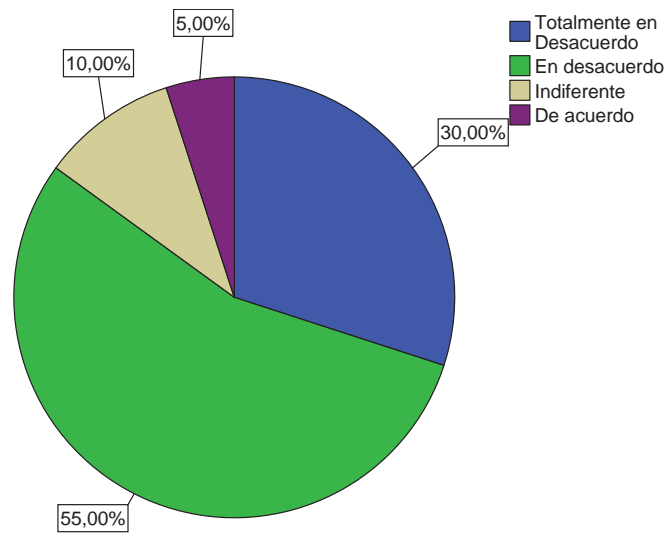


Gráfico D.8 Porcentaje de respuestas ítem 8

Las imágenes empleadas me ayudaron a aclarar los contenidos

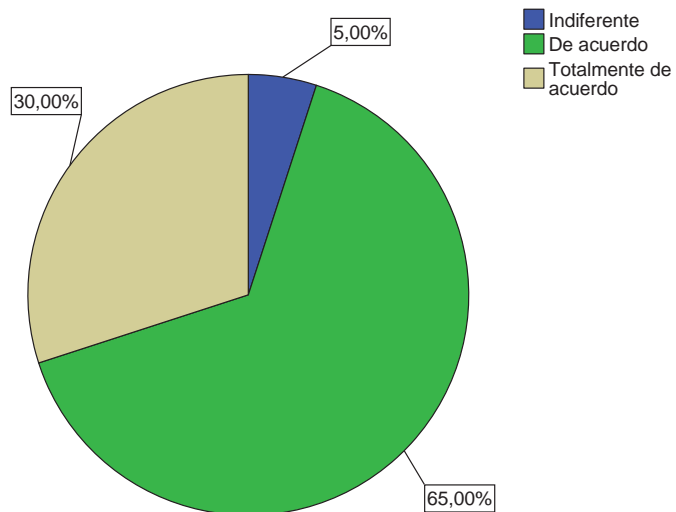


Gráfico D.9. Porcentaje de respuestas ítem 9

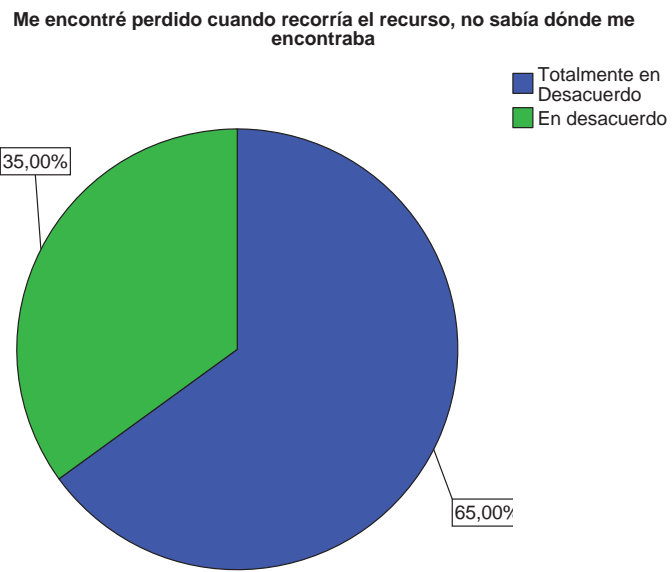


Gráfico D.10. Porcentaje de respuestas ítem 10

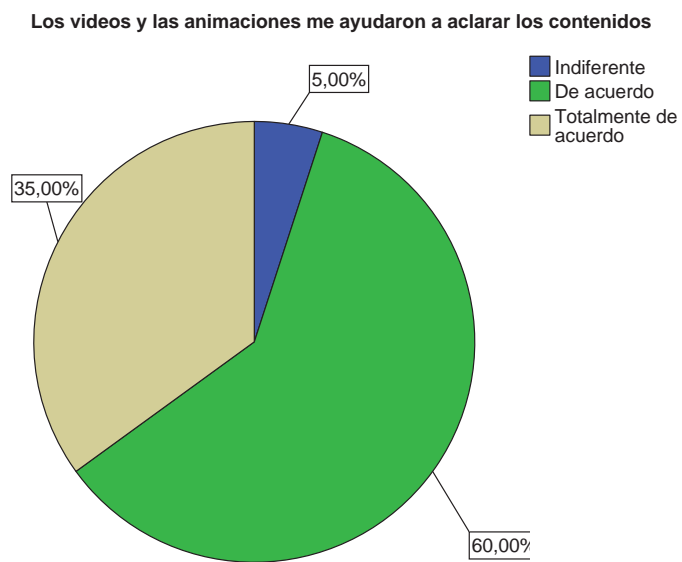


Gráfico D.11. Porcentaje de respuestas ítem 11

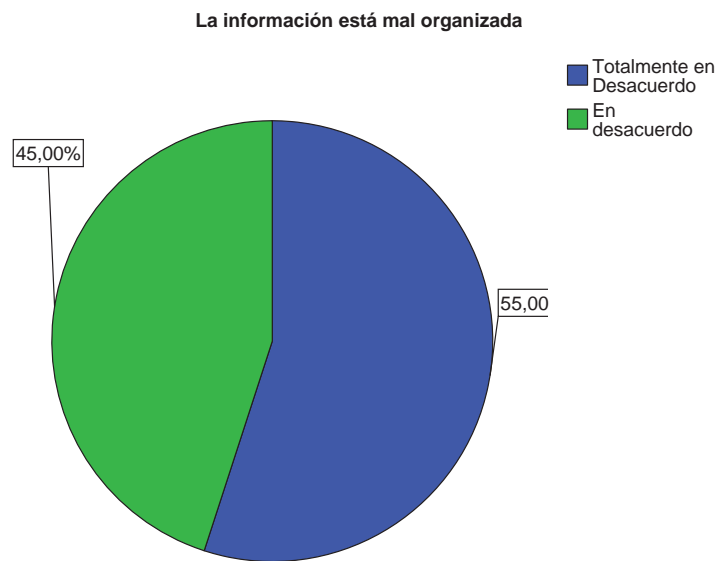


Gráfico D.12. Porcentaje de respuestas ítem 12

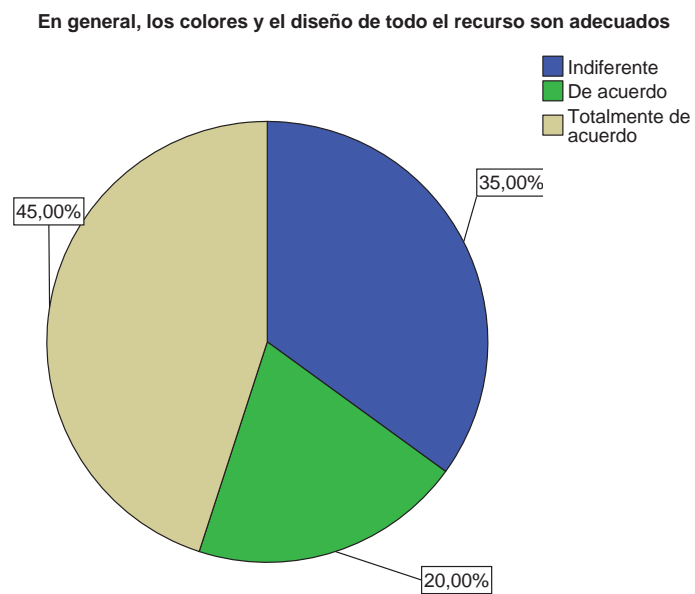


Gráfico D.13. Porcentaje de respuestas ítem 13

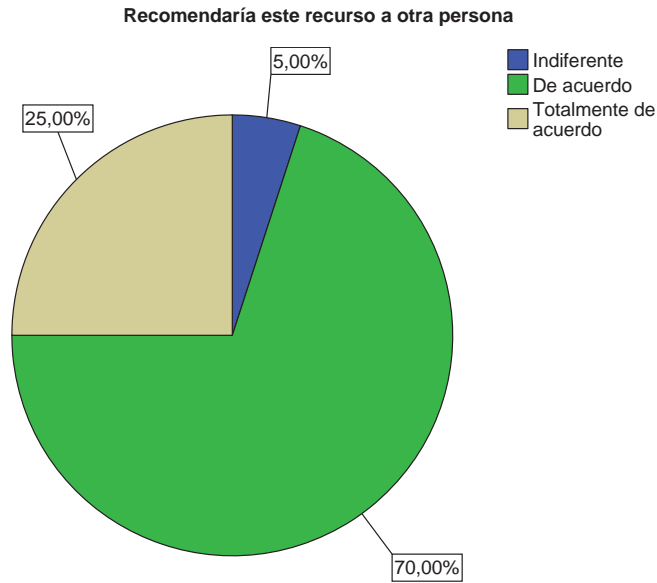


Gráfico D.14. Porcentaje de respuestas ítem 14

D.6. Registros de LOGS

ID	AC	OBJ	CO N	PP	PPV	V1	V2	PPR	V3	V4	PR	OAC	AG1	AG2	GRU PO
1	0	0	0	0	0	279,79	128,9	0	147	109	5,55	3,91	0	0	B
2	8,49	0,89	0,86	0,93	0,91	0,18	0,26	0,88	0,24	0,22	1,86	1,83	0,89	0,98	B
3	2,71	2,86	1,04	1,11	1,17	1,37	1,12	1,14	1,19	1,28	3,74	1,16	1,03	1,17	B
4	0,14		8,8	0,07	42,06	749,94	294,6	0,73	0	0,63	0	5,73	0	0,07	B
5	7,5	0,08	7,31	24,05	23,45	332,45	248,6	27,73	140,9	248,3	549,4	1,84	13,35		B
6	0,08	4,55	1,79	1,64	2,66	138,85	3,83	1,82	46,88	5,12	139	1,24	4,1	325,6	B
7	53,34	78,65	12,9	360,03	61,93	30,74	146,1	4,88	168,3	168,9	23,86	4,34	16,73	4,17	B
8	11,49	0,69	0	16,19	62,12	254,74	0,52	39,17	120,1	276,9	0,72	0	0,67	0	B
9	5,23	17,11	6,21	0,97	1,04	0,83	0,06	1,01	0,38	0,32	1,14	0,92	11,34	2,27	B
10	5,04	7,09	4,35	23,63	38,55	324,02	114,2	7,22	82,03	135	0	2,03	15,67	264,7	B
11	0,09	0,84	0,86	0,79	0,09	3,25	2,31	1,01	1,61	2,04	1,42	1,04	2,85	1,54	B
12	11,06	0,82	0,76	0,76	0,77	0,37	0,52	1,14	1,16	1,57	2,83	1,02	1,19	1,22	B
13	0	0	5,35	46,1	77,76	299,32	328,8	251,29	195,4	260,6	3,85	0	0	0	B
14	5,8	1,25	1,29	1,13	1,59	0,76	0,74	1,08	0,78	0,54	1,09	1,12	5,15	34,49	A
15	1,36	1,28	1,42	1,07	1,55	1,26	0	2,28	2,28	2,44	4,7	7,55	1065,1	3,51	A
16	2,06	2	4,09	3,44	2,34	1,24	1,38	1,37	1,41	1,99	13,89	22,1	37,52	5,06	A
17	1350,2	134,7	14,2	24,64	17,32	1146,5	141,3	61,74	135	289,1	846,6	4,37	65,45	155,5	B
18	5,29	4,09	4,76	2,61	2,1	3,49	15,11	15,37	0,74	0,07	1,02	1,11	1,04	1,08	0

19	1,76	4,62	2,15	30,16	43,11	260,07	271	57,98	206,8	204,2	43,18	7,38	114,09	85,69	B
20	14,06	5,88	2,67	17,92	4,99	3,85	8,58	1199,6	0	0	10,17	4,13	0,99	20,69	A
21	3,49	2,94	1,84	22,42	7,97	0,65	106	0,92	62,15	0,41	38,24	0	9,38	17,17	A
22	8,68	21,21	2,84	21,32	5,84	249,69	110,3	5,47	5,33	7,02	134	4,43	23,31	19,34	A
23	6,02	10,1	1,94	1,07	1,91	3,59	1,14	1,3	4,39	0,28	1,95	1,93	12,82	4,94	A
24	4,64	1,18	2,04	3,18	2,22	0,81	0,83	1,11	1,26	1,03	0,42	0	17,39	11,19	A
25	12,78	56,9	0,89	0,92	1,89	1,12	0,64	0,86	0,19	0,37	60,37	0,97	0,85	0,85	A
26	0	0	0	12,6	125,5	79,69	640,3	0,62	0	559,8	649,3	1,06	1,07	15,23	B
27	6,35	19,38	52,4	5,19	4,76	2,93	0,51	1,15	0,31	2,04	47,71	1,11	122,56	36,29	A
28	0	0	0	695,1	292,2	348,67	130,5	367	71,86	1,71	2,62	10	0	167,2	B
29	16,63	6,28	5,8	0,95	0,92	286,05	0,63	0,99	0,49	1,11	0,94	0,99	0,96	2,04	B
30	300,48	11,43	3,5	1970,3	1,42	19,54	687,2	5,79	15,52	0	0,48	10,1	19,38	5,54	B
31	3,83	13,87	7,93	3,15	3,03	0,65	0,08	0,73	1,59	1,47	12,11	0	296,78	0	A
32	5,96	27,91	8,76	36,47	19,96	167,35	197,5	27,04	116	1,84	4,18	1,85	10,86	15,21	B
33	0,98	0,09	0,92	0,91	0,92	24,56	0,18	0,92	0,02	0,36	2,02	1,03	0,95	0	A

Tabla D.19. Registros de Logs de las categorías

D.7. Histogramas de las variables en cada categoría

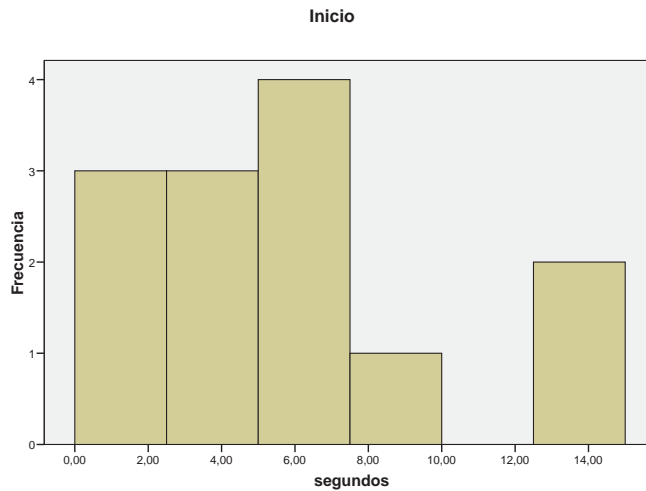


Gráfico D.15. Distribución de Logs AC – GRUPO A N=13

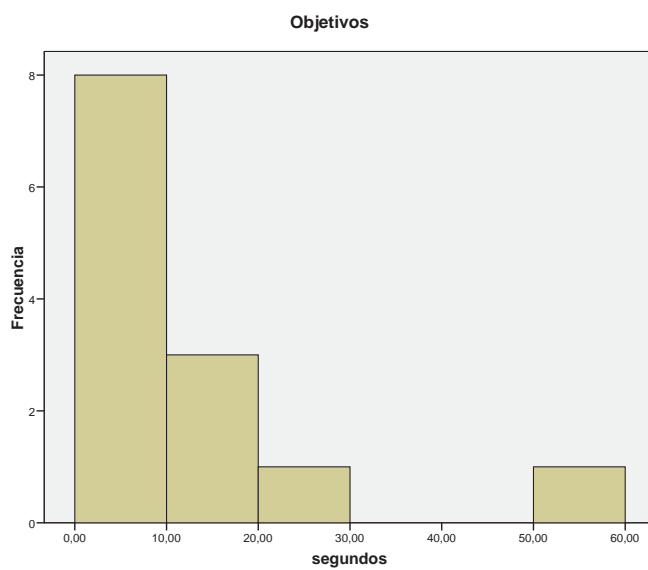


Gráfico D.16. Distribución de Logos OBJ – GRUPO A N=13

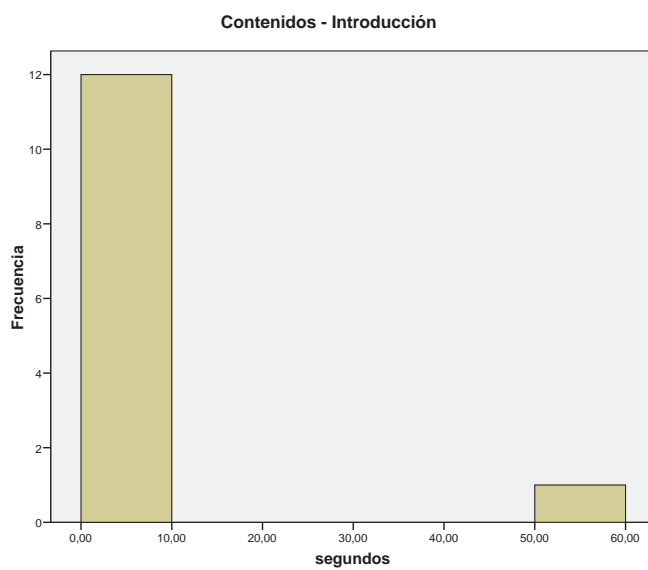


Gráfico D.17. Distribución de Logos CON – GRUPO A N=13

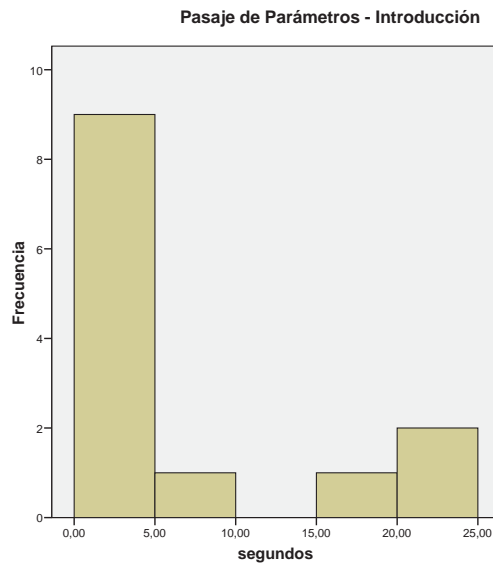


Gráfico D.18. Distribución de Logs PP – GRUPO A N=13

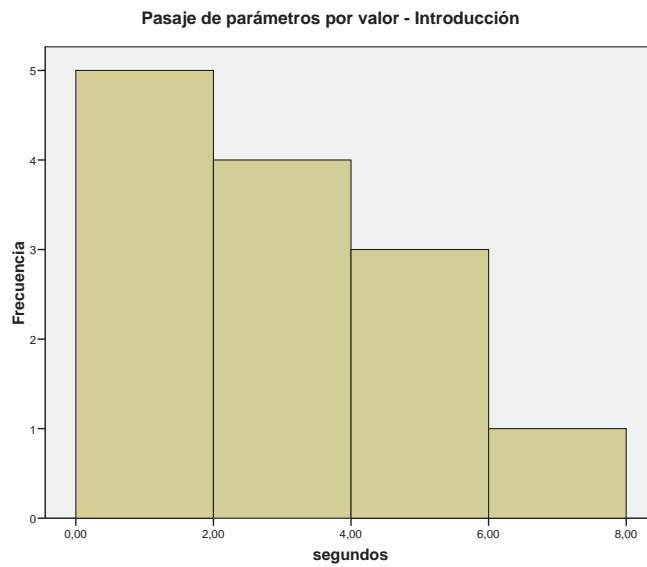


Gráfico D.19. Distribución de Logs PPV – GRUPO A N=13

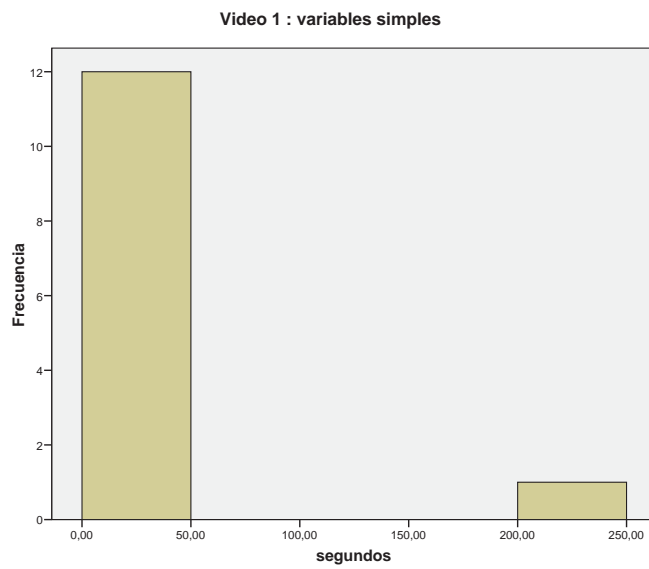


Gráfico D.20. Distribución de Logs V1 – GRUPO A N=13

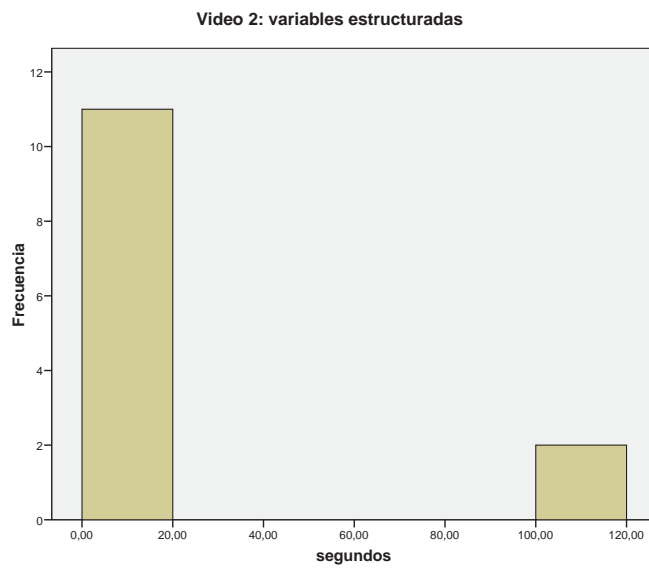


Gráfico D.21. Distribución de Logs – V2 – GRUPO A N=13

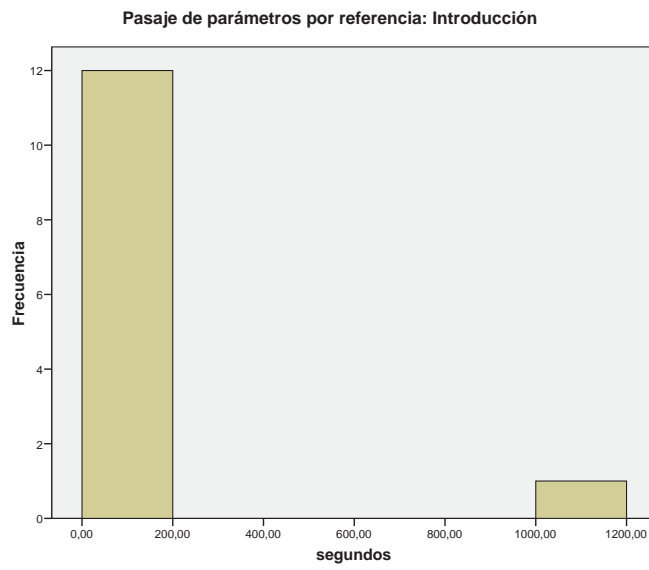


Gráfico D.22. Distribución de Logs PPR – GRUPO A N=13

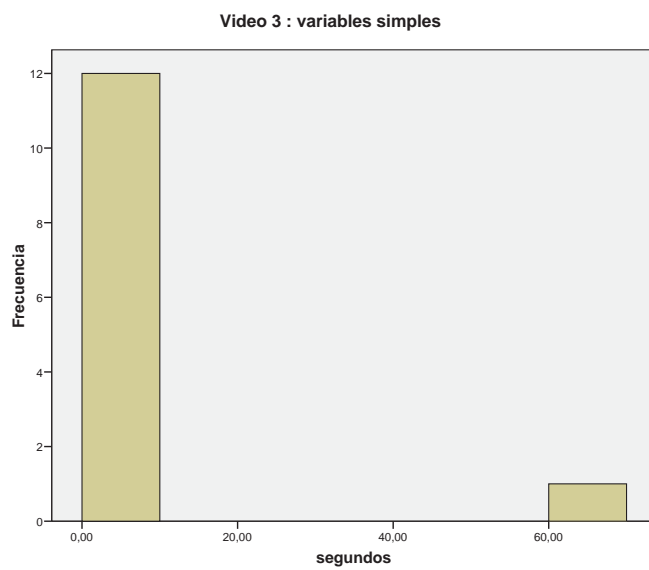


Gráfico D.23. Distribución de Logs V3 – GRUPO A N=13

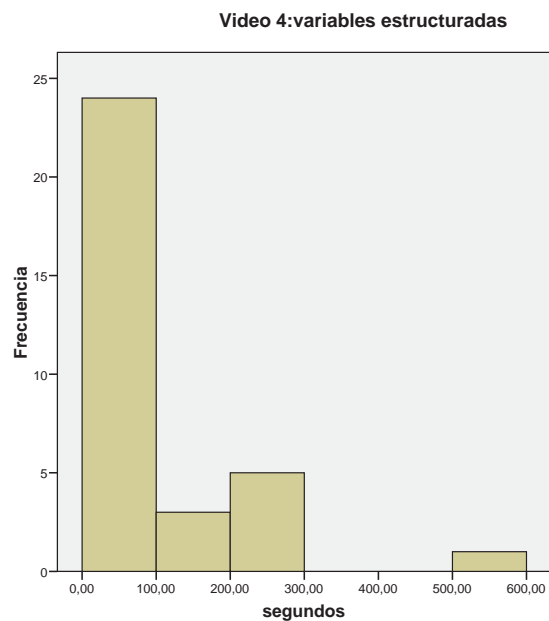


Gráfico D.24. Distribución de Logs V4 – GRUPO A N=13

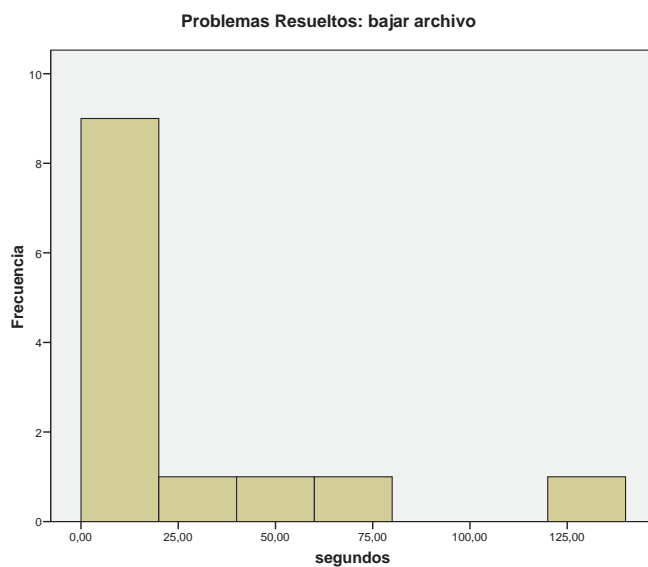


Gráfico D.25. Distribución de Logs PR – GRUPO A N=13

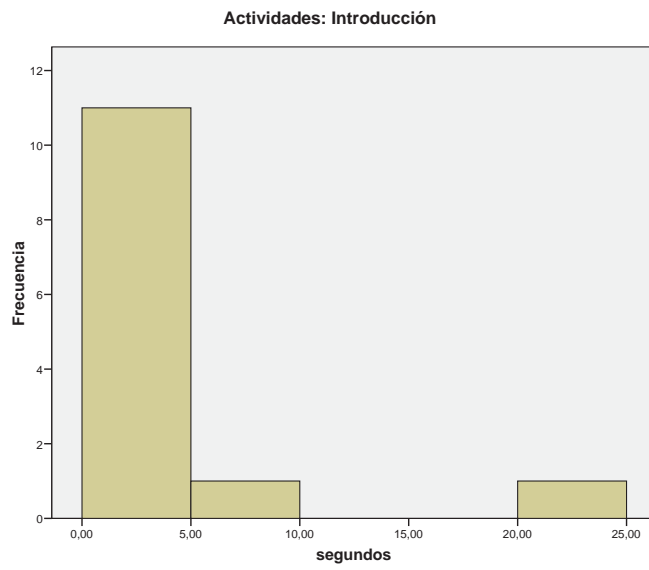


Gráfico D.26. Distribución de Logs – OAC– GRUPO A N=13

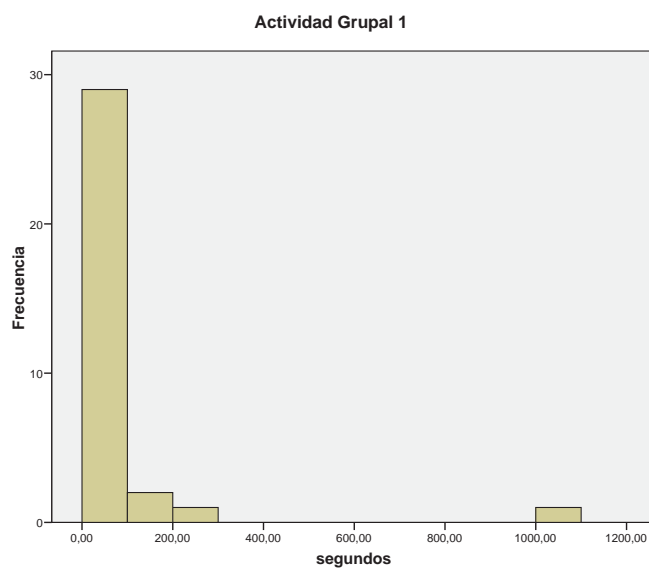


Gráfico D.28. Distribución de Logs AG1 – GRUPO A N=13

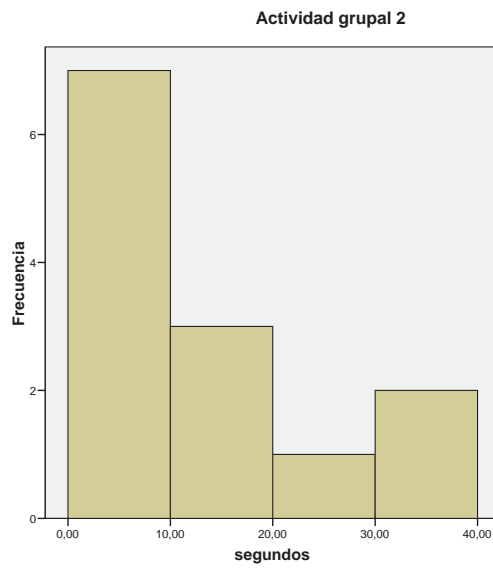


Gráfico D.29. Distribución de Logs AG2 - GRUPO A N=13

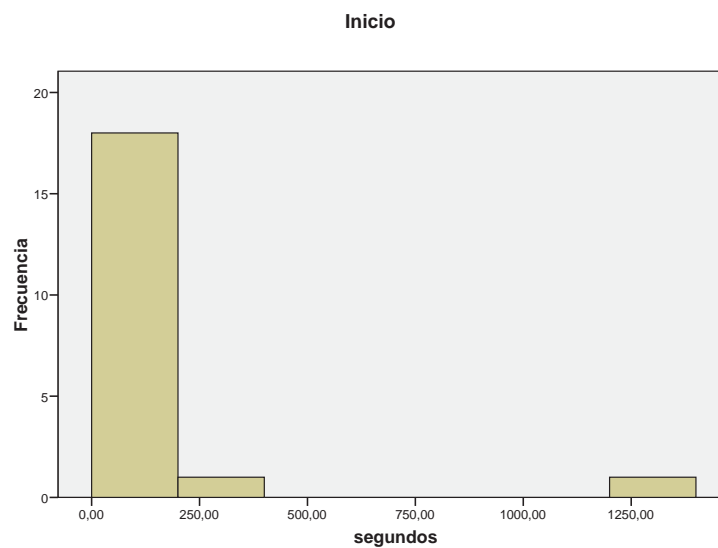


Gráfico D.30. Distribución de Logs OAC- GRUPO B N=20

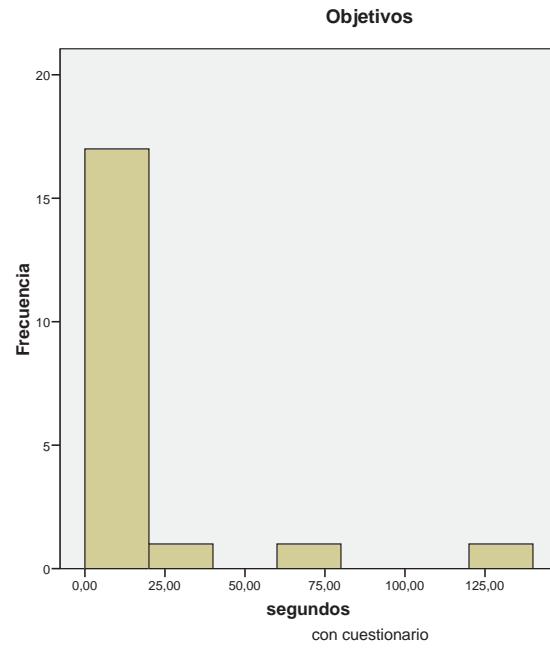


Gráfico D.31. Distribución de Logs OBJ – GRUPO B N=20

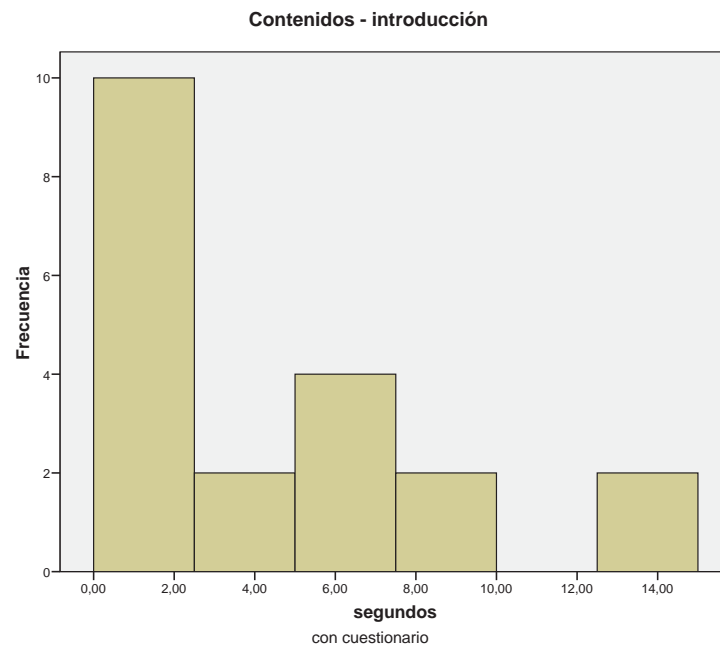


Gráfico D.32. Distribución de Logs CON – GRUPO B N=20

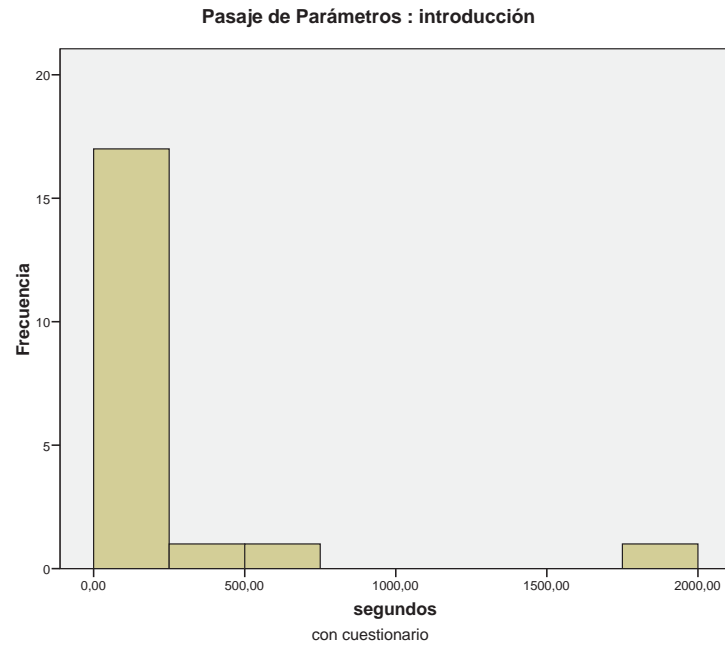


Gráfico D.33. Distribución de Logs PP – GRUPO B N=20

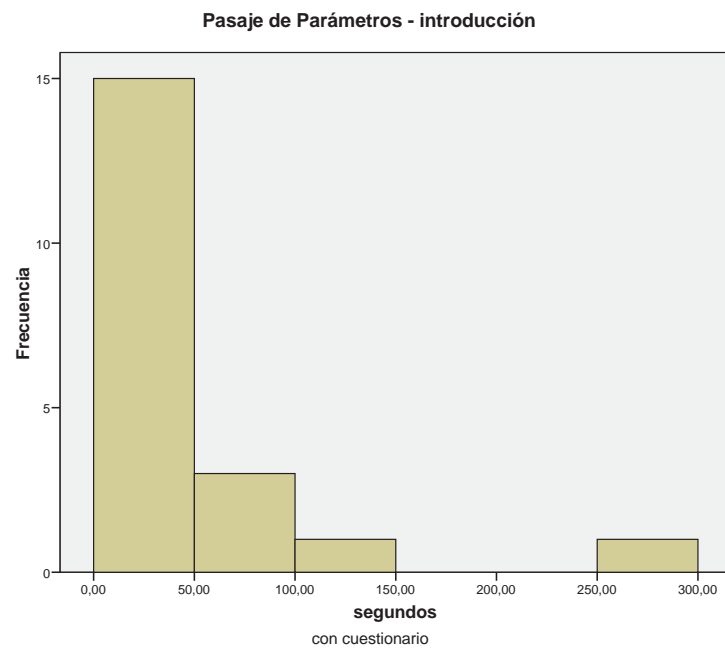


Gráfico D.34. Distribución de Logs PPV – GRUPO B N=20

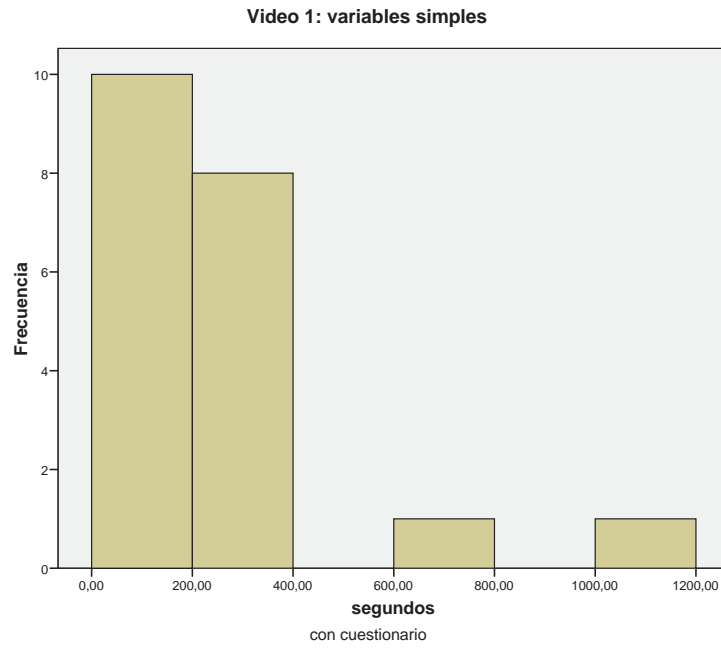


Gráfico D.35. Distribución de Logs V1 – GRUPO B N=20

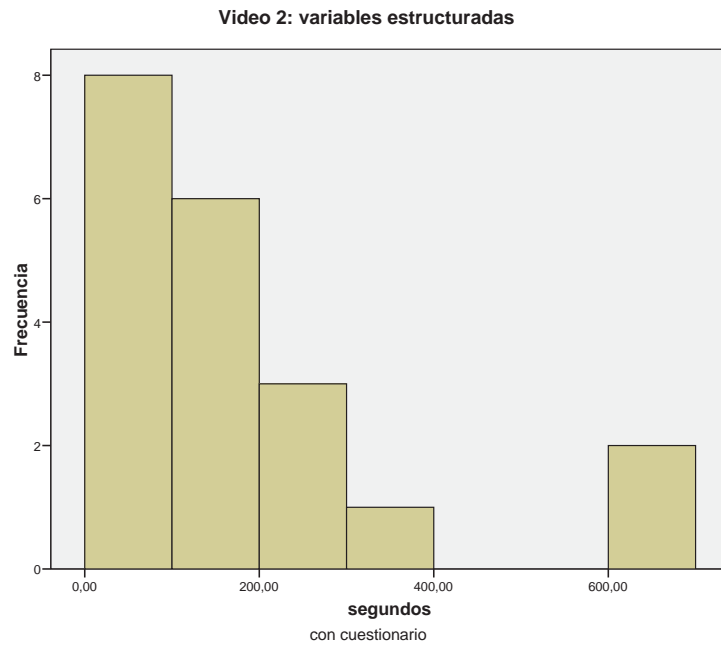


Gráfico D.37. Distribución de Logs V2 – GRUPO B N=20

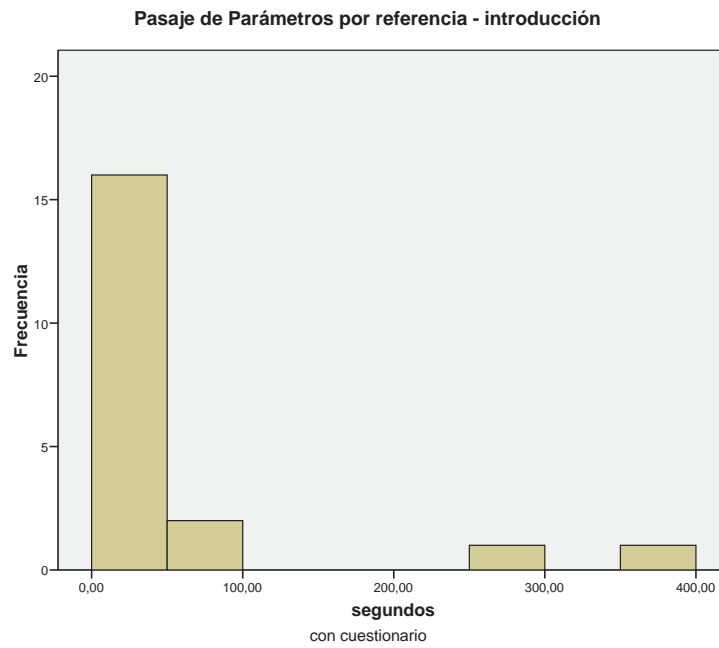


Gráfico D.38. Distribución de Logs PPR – GRUPO B N=20

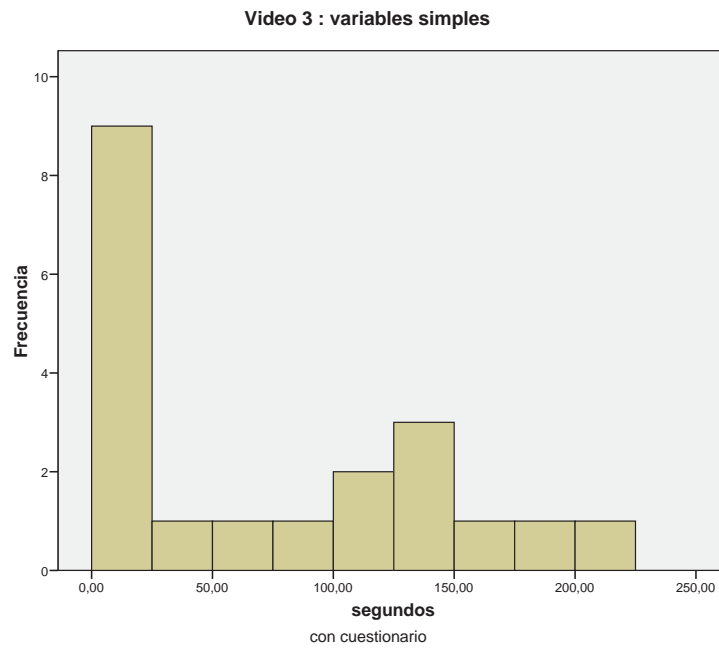


Gráfico D.39. Distribución de Logs V3 – GRUPO B N=20

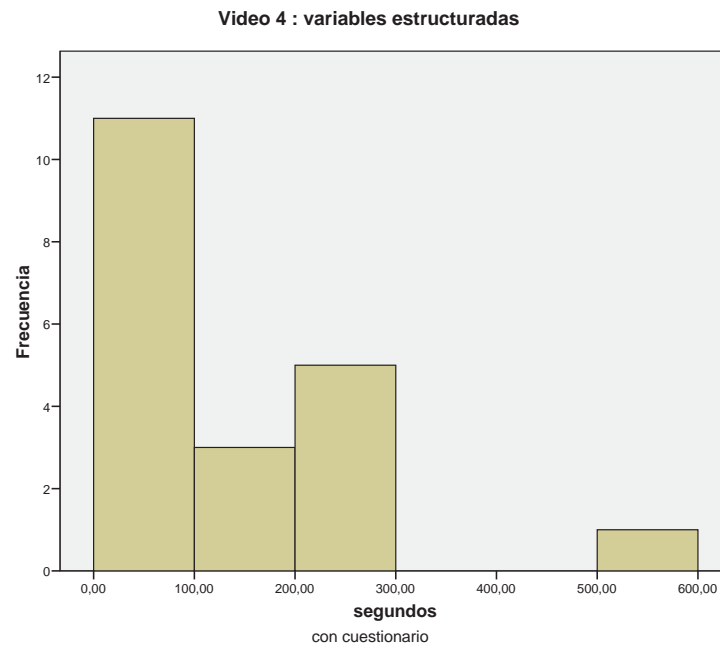


Gráfico D.40. Distribución de Logs – V4 – GRUPO B N=20

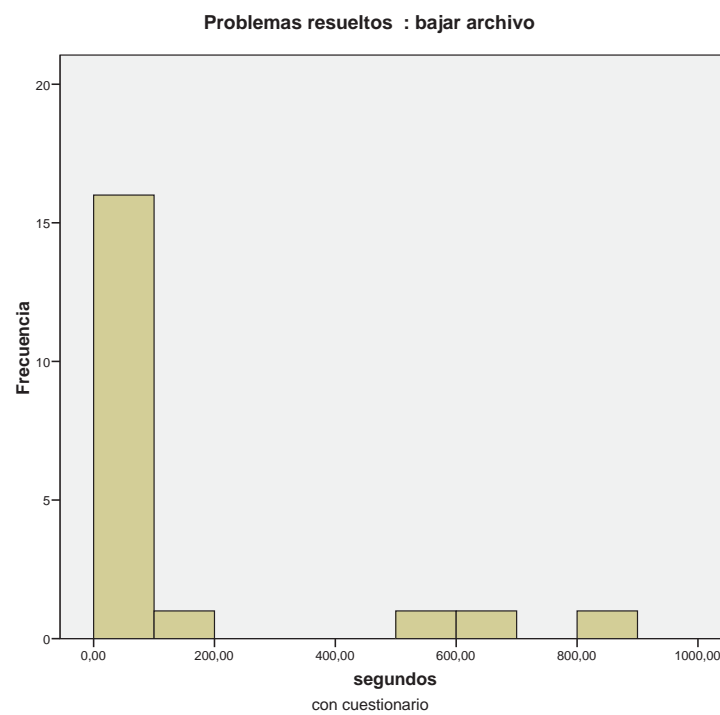


Gráfico D.41. Distribución de Logs PR – GRUPO B N=20

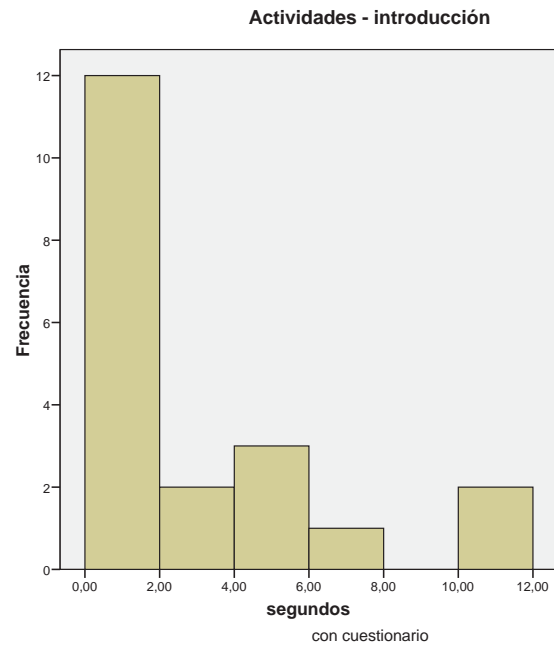


Gráfico D.42. Distribución de Logs OAC– GRUPO B N=20

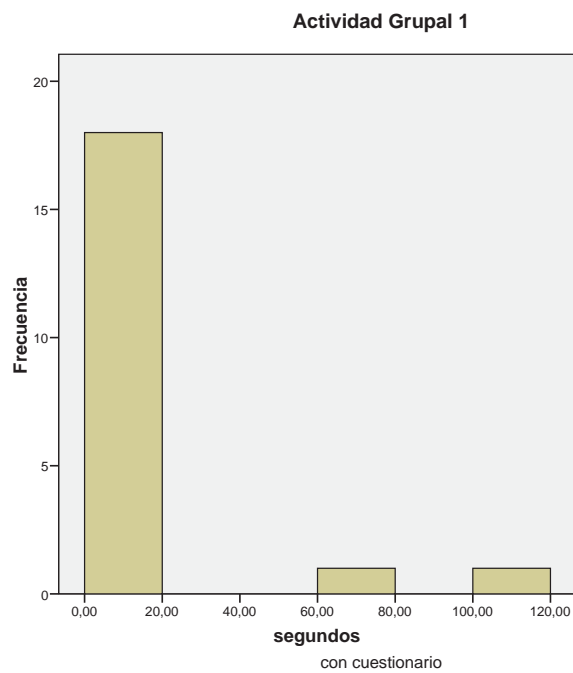


Gráfico D.43. Distribución de Logs AG1 – GRUPO B N=20

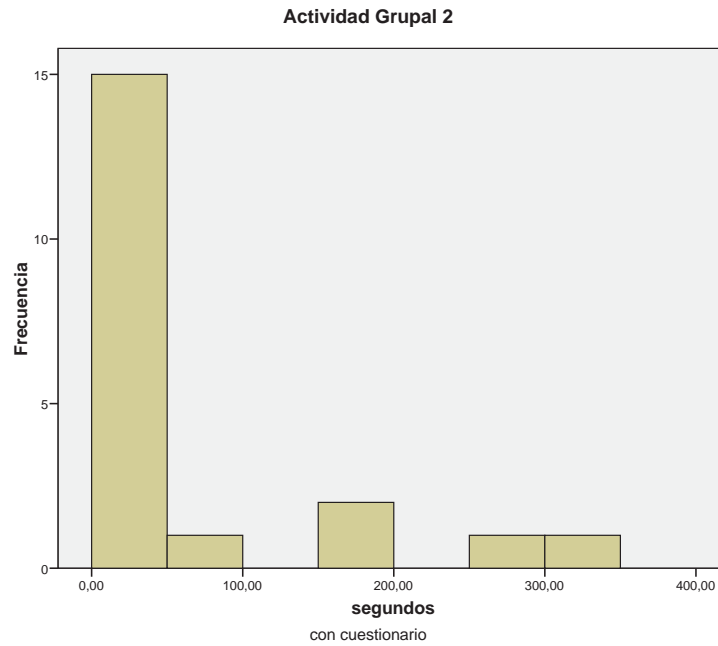


Gráfico D.44. Distribución de Logs AG2 - GRUPO B N=20

D.8. Pruebas de normalidad

		ESTRUCTURA DEL OA	OBJETIVOS			CONTENIDOS CONCEPTUALES		
		AC	OBJ	CON	OAC	PP	PPV	PPR
Diferencias más extremas	Absoluta	0,35	0,42	0,35	0,27	0,45	0,55	0,30
	Positiva	0,35	0,42	0,35	0,08	0,20	0,32	0,10
	Negativa	-0,20	-0,10	-0,25	-0,27	-0,45	-0,55	-0,30
Z de Kolmogorov-Smirnov		0,98	1,19	0,98	0,74	1,26	1,54	0,83
Sig. asintót. (bilateral)		0,29	0,12	0,29	0,64	0,08	0,02	0,49
a. Variable de agrupación: cuest								

Tabla D.20 a. Prueba de normalidad

		VIDEOS			
		V1	V2	V3	V4
Diferencias más extremas	Absoluta	0,62	0,60	0,52	0,45
	Positiva	0,10	0,00	0,02	0,00
	Negativa	-0,62	-0,60	-0,52	-0,45
Z de Kolmogorov-Smirnov		1,75	1,68	1,47	1,26
Sig. asintót. (bilateral)		0,00	0,01	0,03	0,08
a. Variable de agrupación: cuest					

Tabla D.20 b. Prueba de normalidad

D.9. Pruebas de diferencias de medias

		Prueba de Levene para la igualdad de varianzas		Prueba T para la igualdad de medias		
		F	Sig.	t	gl	Sig. (bilateral)
		Inferior	Superior	Inferior	Superior	Inferior
AC	varianzas iguales	3,84	0,06	-0,98	31,00	0,33
	varianzas desiguales			-1,22	19,01	0,24
OBJ	varianzas iguales	1,45	0,24	-0,42	31,00	0,68
	varianzas desiguales			-0,48	28,75	0,63
CON	varianzas iguales	1,92	0,18	0,70	31,00	0,49
	varianzas desiguales			0,58	13,55	0,57
PP	varianzas iguales	5,56	0,02	-1,22	31,00	0,23
	varianzas desiguales			-1,52	19,02	0,14
PPR	varianzas iguales	2,98	0,09	0,66	31,00	0,51
	varianzas desiguales			0,55	13,29	0,59

Tabla D.21. Prueba T-STUDENT

	PPV	V1	V2	V3	V4
U de Mann-Whitney	105,50	52,00	63,00	73,00	65,00
W de Wilcoxon	196,50	143,00	154,00	164,00	156,00
Z	-0,90	-2,87	-2,47	-2,10	-2,40
Sig. asintót. (bilateral)	0,37	0,00	0,01	0,04	0,02
Sig. exacta [2*(Sig. unilateral)]	0,37(a)	0,00(a)	0,01(a)	0,04(a)	0,02(a)

Tabla D.22. Prueba U de Mann-Whitney