



Especialización en Ingeniería de Software – Modalidad a distancia

Administración de Proyectos

Año 2021

Duración: 108 hs. Totales.

Cantidad de horas presenciales/VC: 50hs.
Cantidad de horas de actividades en línea y de trabajo final: 58hs.

Docente Responsable: Dr. Antonelli, Ruben Leandro

Docentes Tutores: Dr. Colla, Pedro
(Académico/Tecnológico)

Mg. Thomas, Pablo
(Académico/Tecnológico)

Tutores: A.C. Alem Joaquin
(Académico/Tecnológico)

Natalia Otero
(Administrativo)

OBJETIVOS GENERALES:

Identificar, analizar y establecer el alcance de aplicación de los conceptos principales de la Administración de Proyectos de Software.
Conocer los fundamentos de la Ingeniería de Software aplicables en forma directa a la Gestión de Proyectos de Software.
Subrayar los aspectos de la Ingeniería de Software y la Gestión de Proyectos, relacionados con la medición de productos y procesos y los enfoques experimentales.
Identificar y analizar el contexto de aplicación de los procesos de calidad y mejora de los procesos de las organizaciones

COMPETENCIAS A DESARROLLAR EN RELACION CON EL OBJETIVO DE LA CARRERA

- C.1- Manejar y aplicar tecnologías actuales para el desarrollo de sistemas de software, incluyendo métodos, lenguajes, arquitecturas, frameworks y herramientas.
- C.2- Tener capacidad para analizar diferentes modelos de proceso de desarrollo de software y evaluar su calidad tanto en aspectos del producto resultante como en la gestión de los individuos involucrados y sus interacciones.
- C.3- Gestionar, planificar y controlar proyectos de software de distinta envergadura.
- C.4- Definir parámetros de calidad tanto interna como externa de un producto software, y



establecer procesos de evaluación y mejora que atiendan la satisfacción de todos los involucrados (el cliente, los usuarios y su experiencia, y el equipo de desarrollo).

C.6- Tener capacidad de analizar el estado del arte en los distintos aspectos de la ingeniería de software.

CONTENIDOS MINIMOS:

- Conceptos principales de la Ingeniería de Software.
- Conceptos principales de la Administración de Proyectos de Software.
- Características de enfoques de desarrollo clásicos y ágiles
- Medición, calidad y mejora del proceso.

PROGRAMA

Software e Ingeniería de Software

Concepto de software. Naturaleza y cualidades del software. Objetivo de la producción de software. Productos de software. Ingeniería de Software (IS). Concepto y panorama de la Ingeniería de Software.

Introducción a los Proyectos de Software

Experiencia en la Gestión de Proyectos de Software. Características de los proyectos de Ingeniería. Razones de los fracasos y los éxitos en los proyectos de software. Unidad conceptual y complejidad dinámica de los proyectos. Importancia de la comunicación en un proyecto. Ley de Brooks.

Gestión de los Proyectos de Software

Introducción al Cuerpo de Conocimiento de la Gestión de Proyectos (PMBOK) del Project Management Institute (PMI). La Oficina de Proyectos (Project Management Office-PMO). Concepto de Proyecto y de Gestión de Proyectos. Procesos de la Gestión de Proyectos, su interacción y evolución a lo largo del proyecto. Áreas de conocimiento del PMBOK. Componentes y restricciones de un proyecto. Los procesos de calidad, riesgos y compras. Actividades del Gerente de Proyecto. Herramientas de gestión.

Ciclos de vida

Definición Modelo, proceso, técnica y herramienta. Ciclos de vida. Ciclo de vida del producto. Ciclo de vida del proyecto. Ciclo de vida de la gestión del proyecto. Integración de los distintos ciclos de vida.



Plan de Gestión de un Proyecto

Gestión de integración de un proyecto en el PMBOK. Desarrollo del plan de proyecto. Fundamentación de un plan de proyecto. Componentes de un plan de proyecto. Tareas, hitos y procesos. Formato y contenido de un Plan de Gerenciamiento de un Proyecto de Software.

Gestión del Alcance y detalle de tareas

La Gestión del Alcance de un proyecto como área de conocimiento del PMBOK. Planeamiento y control del alcance en un proyecto. Alcance del producto, de los procesos y del proyecto. El documento de alcance de un proyecto. La estructura de desglose tareas del proyecto (WBS). Guías prácticas para la construcción. Procesos de aseguramiento de calidad del WBS. Definición de workpackages.

Gestión del Tiempo

La Gestión del Tiempo como área de conocimiento del PMBOK. Procesos de planeamiento y control del tiempo en un proyecto. Actividades en un paquete de trabajo. Contenido de las actividades de gestión del tiempo. Organización de las tareas de gestión de tiempo. Relación entre duración y esfuerzo. Datos para estimar el tiempo disponible. Recursos: definición y clasificaciones. Técnicas y herramientas para estimar tiempos y esfuerzos de recursos. Documentos para el planeamiento y control.

Estimaciones y tamaño del software

Incertidumbre de las estimaciones. Fuentes de errores en las estimaciones. Errores relacionados con la práctica de estimación. Manejo de los errores de estimación. Factores determinantes del esfuerzo, costo, cronograma. Tamaño del software. Medidas de proceso, producto y recursos. Medición de atributos externos e internos del software. Medidas de longitud, funcionalidad, complejidad y reuso.

Control de proyectos

Características generales de los controles. Control Gerencial. Informes de avance. Control de proyectos. Earned Value Analysis (EVA). La medición del avance de un proyecto. Componentes principales del Earned Value Analysis. Variaciones y ratios. Implementación de EVA.

Calidad y Mejora del proceso software

Calidad del software. Procesos de calidad del PMI. Introducción al Modelo de Maduración de Capacidad. Objetivos del modelo clásico de Capability Maturity Model (CMM). Niveles de madurez. Evolución del proceso. Estructura del CMM. Áreas clave del proceso por nivel. Capability Maturity Model Integration (CMMI). Otros modelos de calidad.

Gestión de riesgos.

Concepto de gestión de riesgo: amenazas y oportunidades. Definiciones: incertidumbre, probabilidad, impacto. Actividades: identificación, análisis cualitativo, análisis cuantitativo, planificar respuesta.

Capital humano



Recursos y recursos humanos. Modelos de producción industrial. Canales de comunicación. Comunicación no verbal. Sesgos en la comunicación. Desafíos de la interculturalidad. Equipos: tipología y desarrollo. Personalidades: tipología y motivación. Conducción: estilos, liderazgo y poder. Negociación.

Gestión de proyectos ágiles

Metodologías ágiles. Manifiesto ágil. Principios y valores. Equipos auto-organizados. Liderazgo. XP. SCRUM. Kanban. Kaizen.

MODALIDAD DE EVALUACIÓN Y ACREDITACIÓN

La materia se desarrolla completamente en modalidad virtual a través de encuentros sincrónicos con actividades mediante el entorno virtual de enseñanza y aprendizaje (EVEA). Se requiere un 80% de asistencia a dichos encuentros, incluyendo el encuentro inicial de presentación de la materia, y el encuentro final de integración, ambos de asistencia obligatoria.

La evaluación se realiza en dos etapas. Durante el desarrollo del curso se realizan actividades prácticas que constituyen evaluaciones parciales. Y al final del curso se realiza una evaluación final integradora.

Las actividades prácticas durante el curso tienen distintas mecánicas: trabajos individuales y trabajos grupales. Los trabajos individuales apuntan principalmente a poner en práctica algún método, técnica o herramienta. Mientras que los trabajos grupales (ya sean de 2 o más integrantes) tienen por objetivo desarrollar una opinión crítica sobre algún método, técnica o herramienta. Es allí en donde el discutir con un par es de mucha utilidad. La disciplina de gestión de proyectos es una disciplina cuya aplicación depende del contexto, por lo cual, el intercambio de opiniones es una parte muy valiosa del curso.

Aprobadas las actividades prácticas, los estudiantes deberán rendir un examen final integrador en el cual deberán mostrar conocimiento sobre los temas dictados, como así también su opinión y punto de vista.

RECURSOS Y MATERIALES DE ESTUDIO

El curso dispone de los siguientes recursos y materiales:

- Slides powerpoint para presentar cada uno de los diferentes temas.
- Ejemplos de aplicación.
- Ejercicios prácticos para desarrollar en clase.
- Píldoras formativas para explicar temas puntuales que lo requieran.
- Material de lectura complementaria (libros, artículos científicos o artículos de divulgación), tanto para profundizar los conceptos vistos en clase, como así también para realizar actividades prácticas.



ACTIVIDADES EXPERIMENTALES PLANIFICADAS PARA LA APROPIACIÓN DE LOS SABERES Y LA EVALUACIÓN

Los alumnos deben realizar Trabajos Prácticos (individuales y grupales) en cada eje temático. Estas actividades pueden tomar distintas formas: (i) poner en práctica cierto método, técnica o herramienta, (ii) sintetizar una opinión sobre cierto método, técnica o herramienta (iii) comunicar tanto las características del método, técnica o herramienta, como la opinión sintetizada.

(i) Las actividades prácticas de cierto método, técnica o herramienta serán ejercicios que comenzarán en clase y podrían finalizar en la misma clase o la siguiente. Estas actividades tendrán una consigna que el docente explicará y luego, a partir de los conceptos previamente vistos, los alumnos tendrán que llevarlo a la práctica. Las actividades podrán ser individuales o grupales. Para esto último se configurará el entorno virtual para que los alumnos del mismo grupo se encuentren en un espacio virtual diferente del resto. Durante el desarrollo de la actividad, el docente estará conectado respondiendo dudas y consultas.

(ii) Ya sea en forma explícita (a través un ítem dentro de las consignas) o en forma implícita, los alumnos deberán sintetizar una opinión como conclusión de cada actividad. Los ejercicios grupales permiten que la opinión sea discutida entre los participantes del grupo y así poder tener mejores argumentos. Con el fin de enriquecer esta actividad, los alumnos deberán recurrir al análisis del material bibliográfico complementario.

(iii) Finalizada la actividad, se realizará una sesión de discusión conjunta donde los participantes comunicarán sus opiniones e intercambiarán los distintos puntos de vista.

Específicamente, se realizarán las siguientes actividades con la mecánica descripta a continuación.

Actividad 1: identificación de las características del software. Trabajo individual donde cada alumno selecciona 2 características de un conjunto de características potenciales ofrecidas por el docente. El alumno ingresa las características a una herramienta que construye una nube de palabras con las contribuciones de todos los alumnos. Este trabajo se realiza en forma sincrónica, de manera que los alumnos puedan ver cómo se construye la nube a partir de los aportes y cuales son las características en las que la mayoría coinciden. La actividad finaliza con una presentación de la justificación de las características que han tenido menor consenso (no mayor, en las de mayor consenso es evidente el acuerdo).

Actividad 2: comparación del software con otros productos de otras Ingenierías (por ejemplo, la ingeniería civil). El docente plantea un conjunto de características del software como producto (intangibles, complejo, maleable, etc...). El objetivo de la actividad es evaluar si otros productos de la ingeniería presentan las mismas características que presenta el software. Es una actividad grupal de más de 2 alumnos a realizar durante un encuentro sincrónico. Para tal propósito se configuran salas individuales en el ambiente enseñanza virtual, de forma de



que los alumnos puedan intercambiar opiniones aislados del resto de los alumnos, aunque el docente podrá ingresar para monitorear el trabajo y responder dudas. Luego de la sesión de discusión privada, todos los grupos regresan al espacio común, para expresar las opiniones y registrarlas en un documento colaborativo (como google doc por ejemplo).

Actividad 3: caracterización de los proyectos de software. Actividad grupal de a dos alumnos, a realizar durante la clase y que tiene por objetivo discutir la problemática de los proyectos. Para tal propósito los alumnos ingresarán a salas privadas para que puedan intercambiar opiniones aislados del resto de los alumnos, aunque el docente podrá ingresar para monitorear el trabajo y responder dudas. Luego de la sesión de discusión privada, todos los grupos regresan al espacio de clase común, para expresar sus opiniones. El docente del curso consolida las opiniones en los slides de la clase que luego pondrá a disposición de los alumnos.

Actividad 4: especificación del alcance a través de un WBS. Actividad grupal, en donde todos los alumnos del curso participan simultáneamente en la construcción de un único WBS. El objetivo es el de brindar una situación lo más realista posible para el desarrollo del desglose de tareas (en inglés work breakdown structure), ya que en el mismo se debe involucrar a todo el equipo de desarrollo. El docente del curso oficia de moderador de la actividad y los alumnos proponen y justifican actividades para incluir en el WBS que se realizará en una herramienta colaborativa web. Al final de la clase, el WBS construido se compartirá a todos los alumnos.

Actividad 5: estimación de módulos de software. Actividad individual, que los alumnos realizan luego del encuentro sincrónico y deben entregar un par de días antes del comienzo del siguiente encuentro. Los alumnos deben estimar ciertos módulos especificados por el docente. La estimación consiste en una medida como así también una justificación de la misma. El docente analiza las respuestas de cada alumno, e identifica estimaciones bien contradictorias para discutir al inicio del siguiente encuentro. Lo más rico de la actividad es que distintas personas con distintas experiencias tendrán distintas opiniones, por lo cual, el objetivo es presentar los distintos puntos de vista de forma de enriquecer al conjunto.

Actividad 6: gestión de riesgos. Actividad individual a realizar durante el encuentro sincrónico, en la cual los distintos alumnos contribuyen en un espacio colaborativo (del estilo de google spreadsheet) con la identificación de un riesgo y sus características (probabilidad de ocurrencia, impacto, desencadenante, plan con el tratamiento a realizar cuando el riesgo ocurre, etc.). El objetivo es que en forma simultánea los alumnos puedan describir su riesgo a la vez de que consumen la información producida por sus compañeros, ya que esta información les servirá para enriquecer la descripción de su riesgo. El objetivo de la actividad es concluir que la actividad de riesgos (al igual que la de WBS) son actividades para realizar en forma colaborativa, ya que por más experiencia que tenga una persona, el desarrollo colaborativo permite construir artefactos más completos y detallados.

BIBLIOGRAFÍA BÁSICA



- Nicholas, John M., Steyn, Herman: Project Management for Engineering, Business and Technology, Taylor and Francis, 6th edition 2021
- Ruhe, Günther, Wohlin, Claes: Software Project Management in a Changing World, Springer, ISBN 978-3-642-55034-8, © 2014.
- Project Management Institute: A guide to the Project Management Body of Knowledge (PMBOK guide) 6th edition, ISBN 9781628251845, © 2018.
- Robert K. Wysocki: Effective Project Management: traditional, agile, extreme 7th edition, John Wiley & Sons © 2014
- Bourque, Pierre, Fairley, Richard: Guide to the software engineering body of knowledge, SWEBOK v3.0, IEEE, ISBN 0-7695-5166-1, © 2014.
- Jason, Jennifer: Agile project management: kanban, scrum, kaizen, ISBN 9781719949187, 2018.
- Boehm, B, Rombach, H. D, Zelkowitz, M. V. Foundations of Empirical Software Engineering, The Legacy of Victor R. Basili, Springer-Verlag Berlin Heidelberg, 978-3-540-24547-6, 2005.
- Brooks, F., The Mythical Man-month. Essays on Software Engineering, 1982, Addison Wesley, Reading, Massachusetts.

BIBLIOGRAFÍA COMPLEMENTARIA

- El siguiente material lo conforman papers clásicos de la disciplina y otros más actuales reflejando las tendencias modernas
- Bloch M, Blumberg S, Laartz J Delivering large-scale IT projects on time, on budget and on value. McKinsey Finance 45:28–35, 2013
- Dalcher D The nature of project management: a reflection on the anatomy of major projects by Morris and Hough. Int J Manag Proj Bus 5(4):643–660, 2012.
- Delater A, Narayan N, Paech B, Tracing requirements and source code during software development. In: ICSEA'12: 7th international conference of software engineering advances, pp 274–282, 2012.
- Dingsøyr T, Nerur S, Balijepally V, Moe NB, A decade of agile methodologies: towards explaining agile software development. J Syst Softw 85(6):1213–1221. doi:10.1016/j.jss.2012. 02.033, 2012.
- Eveleens JL, Verhoef C, The rise and fall of the chaos report figures. IEEE Softw 27 (1):30–36, 2010.
- Hayat, F., Rehman, A. U., Arif, K. S., Wahab, K. Abbas, M., "The Influence of Agile Methodology (Scrum) on Software Project Management," 2019 20th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), Toyama, Japan, pp. 145-149, 2019.
- Gibbs, W.W., "Software Chronicle Crisis", Scientific American, November 1994
- Gotel, O., Cleland-Huang, J., Huffman Hayes, J., Zisman, A., Egyed, A., Grunbacher, P., Dekhtyar, A., Antoniol, G., Maletic, J., The grand challenge of traceability (v1.0). In Software and Systems Traceability, pages 343–409. Springer, 2012.



- ISO/IEC/IEEE International Standard - Systems and software engineering - Life cycle processes - Project management," in ISO/IEC/IEEE 16326:2019(E) , vol., no., pp.1-42, 13 Dec. 2019
- Jalali S, Wohlin C, Global software engineering and agile practices: a systematic review. *J Softw Evol Proces* 24(6):643–659, 2012.
- Jørgensen M, Moløkken K, How large are software cost overruns? A review of the 1994 Chaos report. *Inform Softw Technol* 48(8):297–301, 2006.
- Kennedy D, Nur M, The rise of Taylorism in knowledge management. In: *Proceedings of PICMET'12: technology management for emerging technologies (PICMET)*, 2012.
- Kruchten P, The frog and the octopus – a model of software development. *CSI Commun* 35 (4):12–15, 2011.
- Lampasona C, Heidrich J, Basili V, Ocampo A, Software quality modeling experiences at an oil company. In: *Proceedings of the 6th international conference on empirical software engineering and measurement (ESEM)*, 20–21, pp 243–246, 2012.
- Larman C, Basili VR, Iterative and incremental development: a brief history. *Computer* 36 (6):47–56, 2003.
- Moe NB, Dingsøyr T, Dyba T, A teamwork model for understanding an agile team: a case study of a Scrum project. *Info Softw Technol* 52(5):480–491, 2010.
- Moe NB, Smite D, Hanssen GK, From offshore outsourcing to offshore insourcing: three stories. In: *Proceedings of the 7th international conference on Global Software Engineering (ICGSE)*, pp 1–10, 2012.
- Saleem, N., "Empirical Analysis of Critical Success Factors for Project Management in Global Software Development," 2019 ACM/IEEE 14th International Conference on Global Software Engineering (ICGSE), Montreal, QC, Canada, pp. 68-71, 2019.
- Novais RL, Torres A, Mendes TS, Mendonça M, Zazworka N, Software evolution visualization: a systematic mapping study. *Inf Softw Technol* 55:1860–1883, 2013
- Noll J, Beecham S, Richardson I. Global software development and collaboration: barriers and solutions. *ACM SIGCSE bulletin - special section on global intercultural collaboration*, September 2010.
- Steinberga L, Smite D (2011) Towards a contemporary understanding of motivation in distributed software projects: solution proposal, vol 770. University of Latvia, Computer Science and Information Technologies, pp 15–26, 2011.
- Wong, W. Y., Wen Yu, S., Too, C. W., "A Systematic Approach to Software Quality Assurance: The Relationship of Project Activities within Project Life Cycle and System Development Life Cycle," 2018 IEEE Conference on Systems, Process and Control (ICSPC), Melaka, Malaysia, pp. 123-128, 2018.