
Formulario de aprobación de curso de posgrado/educación permanente

Asignatura: Evolución de Software

Modalidad:

(posgrado, educación permanente o ambas)

Posgrado

Educación permanente

Profesor de la asignatura ¹: Ing. Jorge Triñanes, grado 3, Instituto de Computación

Profesor Responsable Local ¹:

Otros docentes de la Facultad: Msc. Ing. Sebastián Pizard, grado 2, Instituto de Computación, Msc. Ing. Cecilia Apa, grado 2, Instituto de Computación,

Docentes fuera de Facultad: Msc. Ing. Elisa Facio, ANDA

Programa(s) de posgrado: Diploma Especialización en Ingeniería de Software
Maestría en Ingeniería de Software

Instituto o unidad: Instituto de Computación

Departamento o área: Ingeniería de Software

Horas Presenciales: 45

Nº de Créditos: 6

Público objetivo: Profesionales en informática, en particular integrantes de equipos de desarrollo y mantenimiento de software. Líderes de equipos de desarrollo y mantenimiento de software. Desarrolladores de software interesados en el mantenimiento y la evolución del software. Estudiantes de posgrado y áreas afines en el desarrollo, el mantenimiento y la evolución del software.

Cupos: Sin cupo.

Objetivos:

El objetivo de este curso es introducir al estudiante en los conceptos de la Evolución y el Mantenimiento de Software. Se verán aspectos teóricos, ejemplos prácticos y reales de aplicación, buenas prácticas de la industria, introducción a los requisitos relacionados a esta disciplina en estándares y normas internacionales. El estudiante aprenderá a analizar los problemas relacionados con la evolución y el mantenimiento de software desde las perspectivas técnica y del negocio, teniendo en cuenta las necesidades de corto y de largo plazo y considerando distintos aspectos, tales como: calidad, costo, esfuerzo y satisfacción de los interesados. Como parte del curso el alumno también conocerá los principales conceptos y prácticas relacionadas a la Gestión de la Configuración del Software.

Conocimientos previos exigidos: Conocimientos generales de ingeniería de software

Conocimientos previos recomendados: ---

Metodología de enseñanza:

Descripción de la metodología:

Las clases presenciales serán expositivas y con resolución de ejercicios durante las mismas. Se expondrán y discutirán temas entre todos los participantes. A lo largo del curso los estudiantes deberán completar tareas de laboratorio iniciadas en clase.

Detalle de horas:

- Horas de clase (teórico): 30
- Horas de clase (práctico): 7
- Horas de clase (laboratorio): 4
- Horas de consulta: N/A
- Horas de evaluación: 4
 - Subtotal de horas presenciales: 45
- Horas de estudio: 25
- Horas de resolución de ejercicios/prácticos: N/A
- Horas proyecto final/monografía: 20
 - Total de horas de dedicación del estudiante: 90

Forma de evaluación:

El curso se evaluará de la siguiente forma:

- Evaluación escrita individual que consiste en preguntas abiertas y/o múltiple opción.
- Presentación de un trabajo grupal del estilo de laboratorio sobre temas del curso.

Presentación en clase de un artículo científico reciente publicado en alguna revista especializada. La presentación debe mostrar un análisis crítico por parte del estudiante.

Temario:

1. Conceptos básicos de la evolución y mantenimiento de software
2. Marco general y fundamentos del cambio del software
3. Limitaciones e impacto económico del cambio del software
4. Comprensión de los programas
5. Refactoring y reestructuración
6. Ingeniería reversa
7. Gestión de la Configuración
8. Testing y su relación con Gestión de la Configuración
9. Gestión y organización
10. Documentación y conocimiento
11. Sistemas legados
12. Estrategias de liberación
13. Deuda técnica
14. Arquitectura y evolución del software
15. Integración, liberación y despliegue continuos, DevOps
16. Herramientas
17. Mediciones y uso de la información disponible
18. Modelos de evaluación y de mejora del mantenimiento
19. Procesos de evolución y mantenimiento

Bibliografía:

Software Evolution and Maintenance: A Practitioner's Approach, Priyadarshi Tripathy, Kshirasagar Naik, Wiley, ISBN: 978-0-470-60341-3, 2014

Software Maintenance: Concepts and Practice (2nd edition), Penny A. Grubb, Armstrong A. Takang, World Scientific, ISBN 981-238-425-1, 2003

Design – Who needs an architect?, Martin Fowler, IEEE Software, pp. 11-13, IEEE, 15 Sep. 2003

Software Maintenance Maturity Model (Smmm): the software maintenance process model, Alain April, Jane Huffman Hayes, Alain Abran, Reiner Dumke, Journal of Software Maintenance and Evolution: Research and Practice, 2005, 17, pp. 197-223

SMmm Model to Evaluate and Improve the Quality of Software Maintenance Process, Alain A. April, Reiner R. Dumke, Alain Abran, disponible en:
https://www.researchgate.net/profile/Alain_Abran/publication/221569924_SMCMM_Model_to_Evaluate_and_Improve_the_Quality_of_the_Software_Maintenance_Process/links/0fcfd511c98ba62b8f000000/SMCMM-Model-to-Evaluate-and-Improve-the-Quality-of-the-Software-Maintenance-Process.pdf?origin=publication_list

Configuration Management Principles and Practice - Anne Mette Jonassen Hass - Addison-Wesley Professional - ISBN-10: 0321117662 – 2003

The DevOps Handbook: How to Create World-Class Agility, Reliability, & Security in Technology Organizations (2nd edition), Gene Kim, Jez Humble, Patrick Debois, and John Willis, IT Revolution Press, ISBN-10: 1950508404, 2021

Datos del curso

Fecha de inicio y finalización: desde 14 de noviembre al 16 de diciembre de 2022.

Horario y Salón: Lunes, miércoles y viernes de 18 a 21 h.
Salón a confirmar.

Arancel: \$25.200

[Si la modalidad no corresponde indique "no corresponde". Si el curso contempla otorgar becas, indíquelo]

Arancel para estudiantes inscriptos en la modalidad posgrado: \$25.200

Arancel para estudiantes inscriptos en la modalidad educación permanente: \$25.200
