

Facultad de Ingeniería y Ciencias
Escuela de Informática y Telecomunicaciones

Descriptor de asignaturas

Fundamentos de Ingeniería y Arquitectura de Software

1. Identificación de la asignatura:

Nombre de la Asignatura: Fundamentos de Ingeniería y Arquitectura de Software	
Códigos: CDI-2009	Créditos: 5
Duración: Semestral	Ubicación en el plan de estudios: Semestre 7
Requisitos: CDI-1104 – Bases de datos, CDI-2002 – Proyecto de integración I	
Sesiones cátedras semanales: 2 cátedras	
Sesiones de Ayudantía: 1	

2. Descripción de la asignatura:

Este curso entrega los fundamentos necesarios para el diseño, evaluación y optimización de soluciones de software a través de metodologías de ingeniería de software y principios de arquitectura moderna. Se abordan técnicas de recolección de requerimientos, diseño modular, patrones arquitectónicos y tecnologías como microservicios y contenedores. El curso culmina con un proyecto colaborativo aplicado, reforzando las competencias necesarias para enfrentar desafíos reales en la industria del software.

3. Resultados de Aprendizaje:

1. Aplica metodologías modernas para el análisis, diseño y evaluación de software.
2. Diseña arquitecturas de software escalables y seguras, basadas en principios de calidad.
3. Formula y analiza requerimientos, utilizando herramientas actuales.
4. Despliega soluciones de software siguiendo buenas prácticas, a partir de una correcta documentación e implementación.
5. Participa en proyectos colaborativos, aplicando principios de arquitectura y diseño.

4. Unidades Temáticas:

- **Unidad 1: *Introducción a la Ingeniería y Arquitectura de Software***
 - *Importancia de la arquitectura en el ciclo de vida del software*
 - *Rol del/la ingeniero/a de software en equipos modernos*
- **Unidad 2: *Ingeniería de Requerimientos***
 - *Requerimientos funcionales y no funcionales*
 - *Historias de usuario y criterios de aceptación*
 - *Técnicas de análisis y priorización*
- **Unidad 3: *Estilos y Principios de Arquitectura de Software***
 - *Arquitecturas en capas, cliente/servidor, orientada a servicios (SOA)*
 - *Atributos de calidad: escalabilidad, seguridad, rendimiento, mantenibilidad*
 - *Principios de diseño arquitectónico*
- **Unidad 4: *Diseño Modular y APIs***
 - *Descomposición de sistemas en módulos cohesivos*

- *Definición de interfaces claras y uso de APIs*
- **Unidad 5: Patrones de Diseño y Arquitectura**
 - *MVC, Microkernel, Pipeline*
 - *Patrones comunes y antipatrones*
- **Unidad 6: Introducción a Microservicios y Contenedores**
 - *Comparación con arquitectura monolítica*
 - *Contenedores y orquestación*
- **Unidad 7: Evaluación, Planificación y Análisis de Brechas**
 - *Evaluación integral del proyecto: revisión de objetivos, procesos, requerimientos y soluciones actuales.*
 - *Identificación de brechas entre el estado actual y el estado deseado del sistema y del proyecto.*
 - *Planificación de mejoras: arquitectura, procesos, tecnología, organización*

5. Descripción general del método de enseñanza:

Las clases se imparten en modalidad teórico-práctica. Se espera que el alumno adquiera los conocimientos técnicos y metodológicos necesarios mediante clases expositivas, y su aplicación a problemas realistas. Las clases presentan los principios, métodos y técnicas utilizadas en Ingeniería de Software. Para poner en práctica los conceptos aprendidos, se realizará un proyecto semestral en contextos reales de manera colaborativa con equipos formados por los alumnos.

A partir de las actividades antes mencionadas se desarrollará la capacidad para modelar formalmente sistemas informáticos. La disciplina de Ingeniería de Software tiene una relación estrecha con la habilidad de un ingeniero para reducir los riesgos del software. Planificar, analizar y diseñar son actividades orientadas a esos aspectos y serán reforzadas/evaluadas durante la ejecución del proyecto antes descrito y en las evaluaciones formales del curso.

6. Descripción general de la modalidad de evaluación:

Se contempla la realización de evaluaciones parciales (controles y trabajos), un proyecto semestral, dos pruebas solemnes y un examen final. Si la nota asociada a la ejecución del proyecto semestral es inferior a 4.0, el alumno reprobará la asignatura con nota final igual a la nota obtenida en el proyecto.

Podrán eximirse aquellos alumnos que cumplan con los siguientes requisitos:

1. Nota de presentación mayor o igual a 5.0.
2. Solemne 1, Solemne 2 y Proyecto mayor o igual 4.0.
3. Todas las evaluaciones rendidas, incluyendo las dos solemnes, las 3 entregas del proyecto y todas las evaluaciones parciales.

7. Bibliografía Básica Obligatoria:

1. Sommerville, Ian. *Ingeniería de Software*, 10ª ed., Prentice-Hall, 2015.
2. Pressman, Roger. *Ingeniería del Software: Enfoque práctico*, 8ª ed., McGraw Hill, 2014.
3. Gamma, E. et al. *Patrones de diseño*, Addison-Wesley.
4. Craig Larman. *UML y Patrones*, 3ª ed.
5. Apuntes y recursos del curso disponibles en línea.