

Facultad de Ingeniería
Escuela de Informática y Telecomunicaciones

PROGRAMA DE ASIGNATURA
Arquitectura de sistemas

I. Identificación

Código	: CIT-2004
Créditos	: 6
Duración	: Semestral
Ubicación en plan de estudios:	Semestre 8
Requisitos	: Ingeniería de software (CIT-2005)
Sesiones Semanales	: 2 cátedras, 1 ayudantía

II. Objetivos Generales y Específicos

El objetivo general del curso corresponde a entender los enfoques, métodos y las técnicas que permiten la aplicación de un enfoque sistemático, disciplinado, cuantificable, para la especificación y evolución de arquitecturas de sistemas de software, consistentes con el desarrollo, operación, y mantenimiento de software.

En este contexto, al finalizar el curso el alumno será capaz de:

- Aplicar disciplinada y sistemáticamente métodos y técnicas de especificación y evolución de arquitecturas de sistemas de software.
- Aplicar métodos formales (cuantitativos, cualitativos) de apoyo a la especificación y evolución de arquitecturas de sistemas de software.
- Aplicar capacidades de modelamiento, análisis y síntesis como apoyo a la especificación y evolución de arquitecturas de sistemas de software.

III. Descripción de Contenidos

1. Introducción (fundamentos y desempeño, definición y conceptos básicos de arquitectura de software)
2. Diseño arquitectural de software
 - a. Estilos arquitecturales
 - b. Vistas arquitecturales
 - c. Componentes y conectores
3. Diseño de propiedades no funcionales
 - a. Atributos de calidad de software
 - b. Análisis de arquitecturas
4. Implementación
5. Pruebas, medición y validación de la arquitectura en base a atributos de

calidad.

IV. *Importancia del curso en el plan de estudios*

La matriz de competencias de la asignatura propone la capacidad para diseñar formalmente arquitecturas como una competencia primordial para el desarrollo de sistemas informáticos. En este caso, el diseño se sustenta en el pensamiento analítico, pero también en la abstracción y síntesis.

Se propone un perfil de la asignatura centrado en las capacidades técnicas y de pensamiento analítico-sintético. Adicionalmente, la capacidad de comunicación oral y escrita, como también el aprendizaje autónomo deben ser dos competencias presentes en cada uno de los cursos de la carrera.

Así, este curso contribuye al cumplimiento del perfil de egreso a través del desarrollo del siguiente conjunto de objetivos de aprendizaje (vistos como una serie de conocimientos, habilidades, actitudes y valores):

- Modelar el comportamiento de sistemas, empleando lenguaje matemático, conceptos de física, lenguaje computacional y simulación, entre otros métodos.
- Elaborar y adaptar diseños en el campo de las Tecnologías de la Información y de las Comunicaciones, que permitan satisfacer necesidades detectadas mediante el diagnóstico y la modelación.
- Conocer y aplicar los métodos de gestión de proyectos
- Planificar, analizar y diseñar sistemas informáticos y de telecomunicaciones con una visión de negocio.
- Contribuir al mejoramiento de la calidad de los procesos de producción de software y/o de diseño de sistemas de telecomunicaciones.
- Comunicar ideas en forma oral y escrita
- Capacidad de pensar en forma analítica y racional
- Habilidad de procesar datos generados experimentalmente
- Capacidad de abstracción y modelación
- Habilidad de identificar, formular y resolver problemas complejos de forma autónoma, con enfoque sistémico
- Capacidad de integrar conocimientos
- Capacidad de liderar, dirigir, organizar y planificar
- Capacidad de trabajar en equipos disciplinarios o multidisciplinarios
- Capacidad de aprender en forma autónoma y continua
- Capacidad de actuar con iniciativa y tomar decisiones
- Capacidad de crítica y autocrítica.
- Motivación al logro y a la calidad
- Ética profesional acorde con los valores de la Universidad

V. Metodología

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 para diseñar y evaluar arquitecturas de software. Para poner en práctica los conceptos aprendidos en problemas realistas, un proyecto es realizado por los alumnos.

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

VI. Evaluación

Se contempla la realización de evaluaciones parciales (controles y trabajos), un proyecto semestral, dos pruebas solemnes de igual valor 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. En caso contrario, la nota de presentación se calcula a través de la siguiente fórmula.

Si promedio solemnes es mayor o igual a 4.0:

Nota de Presentación = (25% Solemne 1 + 25% Solemne 2 + 15% Proyecto + 5% Parciales)/0.7

Si promedio solemnes es menor a 4.0:

Nota de Presentación = (32.5% Solemne 1 + 32.5% Solemne 2 + 5% Parciales)/0.7

Nota Final = 70% Nota de Presentación + 30% Examen.

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.

VII. Bibliografía básica de referencia

Bibliografía obligatoria

1. Gorton, Ian. *Essential Software Architecture*. Springer- Verlag, 2006.
2. Taylor, R. N., Medvidovic, N., Dashofy, E. *Software Architecture: Foundations, Theory, and Practice*. John Wiley and Sons, 2009.

Bibliografía complementaria

1. Larman, Craig, *UML y patrones una introducción al análisis y diseño orientado a objetos y al proceso unificado*, Madrid Pearson 2003.
2. Fowler, Martin. *Patterns of enterprise application architecture*. Addison Wesley, 2003.

PAUTAS ETICAS BASICAS

El plagio es el uso de las ideas o trabajo de otra persona sin el adecuado consentimiento. El plagio puede ser intencional o no. El plagio intencional es el claro intento de hacer pasar el trabajo o ideas ajenas como el suyo propio para su beneficio. El plagio no intencional puede ocurrir si Ud. no conoce el mecanismo adecuado de referenciar la fuente de sus ideas e información. Si no está seguro de los métodos aceptados para referenciar, debería consultar con su profesor, tutor o personal de biblioteca.

El plagio comprobado es una actitud que puede resultar en severas sanciones disciplinarias y/o en la exclusión de la Universidad (Artículo 44, Reglamento del Estudiante de Pregrado).

Elaborado por: Beatriz Marín, Andrea Nieto, Jonathan Frez

Fecha revisión: Diciembre 2015

Fecha vigencia: Marzo 2016