



Programa de la Asignatura:

Diseño y Arquitectura de Sistemas de Computación

Código: 28



Carrera: **Ingeniería en Computación** Plan: **2013** Carácter: **Obligatoria**
Unidad Académica: **Secretaría Académica** Curso: **Cuarto Año – Primer cuatrimestre**
Departamento: **Ingeniería** Carga horaria total: **128** hs. Carga horaria semanal: **8** hs.
Resolución de Problemas de Ingeniería: **10 %** Formación teórica: **50 %**
Actividades de Proyecto y Diseño: **30 %** Formación experimental: **10 %**
Práctica Supervisada: **00 %**

Materias Correlativas Obligatorias

- **20 Sistemas Operativos**
- **21 Fundamentos de Computación**
- **24 Laboratorio Microprocesadores**
- **26 Arquitectura de Computadoras II**

Cuerpo Docente

Mg. Ing. Diego Fontdevila

Índice

- Fundamentación pág. 2
- Encuadre y articulación de la asignatura pág. 2
 - Encuadre dentro del Plan de Estudios pág. 2
 - Articulación Horizontal pág. 3
 - Articulación Vertical pág. 3
- Objetivos pág. 3
 - Objetivo General pág. 3
 - Objetivos Específicos pág. 3
- Contenidos mínimos pág. 3
- Programa analítico pág. 3
- Bibliografía básica pág. 3
- Bibliografía de consulta pág. 4
- Metodología del aprendizaje pág. 4
 - Desarrollo de la asignatura pág. 4
 - Dinámica del dictado de las clases pág. 4
 - Trabajos prácticos pág. 4
- Metodología de evaluación pág. 5
- Planificación pág. 5
- Información de versiones pág. 6

AÑO ACADÉMICO 2013

ÚLTIMA REVISIÓN 03/07/2013

Firma Docente

Firma Coordinador

1. FUNDAMENTACION

Esta materia se propone desarrollar las capacidades de los alumnos para realizar el diseño y la implementación de un sistema de computación con foco en el software.

Las actividades de diseño y requerimientos son centrales al proceso de desarrollo de software, puesto que una vez que el diseño del sistema está completamente especificado, el producto está listo. Dicho de otra forma, no hay una etapa de manufactura posterior, si no que todo el trabajo de desarrollo, incluyendo la codificación como actividad de diseño detallado, corresponde a la etapa de diseño de producto.

En esa perspectiva, y difiriendo la introducción a la problemática de requerimientos para la materia Ingeniería de Requerimientos, esta materia se enfoca en el diseño y construcción de un sistema completo. El objetivo es mostrar múltiples herramientas pero también aportar un método de referencia que habilite a los alumnos a desarrollar su tarea profesional en forma eficaz.

2. ENCUADRE Y ARTICULACIÓN DE LA ASIGNATURA

Esta materia pertenece al grupo de materias de software de la carrera, y aporta dos aspectos fundamentales a la formación de los alumnos:

- Técnicas de diseño y arquitectura, sobre la base de la formación anterior.
- El cierre de un proceso de desarrollo completo, incluyendo la codificación, pruebas, puesta en producción, implementación sobre hardware, diseño, arquitectura y requerimientos.

Articulación Horizontal

Esta materia se cursa en el mismo cuatrimestre que Base de Datos, esta última puede aportar conocimientos de esos componentes si se decidiera integrarlos al proyecto de desarrollo de esta materia.

Articulación Vertical

En ese aspecto de cierre del proceso antes descrito, esta materia actúa como una experiencia integradora para todas las materias de software anteriores de la carrera, incluyendo las materias de programación.

Finalmente, esta materia presenta a los alumnos algunos aspectos esenciales del software como la complejidad, que serán retomados en materias posteriores, como Ingeniería de Software e Ingeniería de Requerimientos.

3. OBJETIVOS

Objetivo General

Que el alumno adquiera manejo fluido de diversas técnicas de diseño y arquitectura de sistemas, incluyendo métodos orientados a objetos, con vistas a desarrollar capacidades de selección y mirada crítica sobre las técnicas aprendidas.

Objetivos Específicos

De esa manera, construir sólidos profesionales con un conjunto de habilidades a su disposición y criterio para utilizarlas:

- Desarrollar en el alumno una experiencia práctica del proceso de desarrollo, incluyendo todas las actividades desde la codificación hasta las pruebas de aceptación.
- Exponer al alumno a las complejidades intelectuales, humanas y tecnológicas del diseño de software para permitirle una aprehensión inmediata de su actividad profesional.

- Trabajar en conjunto múltiples fuentes de bibliografía para desarrollar en el alumno la capacidad de lectura crítica en el contexto de su profesión.

4. **CONTENIDOS MÍNIMOS**

Conceptos de Diseño. Características de un buen diseño. Diseño como proceso de decisión. Arquitectura como diseño estratégico del sistema. Diseño a distintos niveles. Tipos de Estructura de un sistema: Código, Tiempo de Ejecución, Física. Métodos, Procesos y Modelos de Ciclo de vida. Diseño Orientado a Objetos. Introducción al análisis y diseño estructurado. Diseño guiado por atributos de calidad. Estilos y patrones de arquitectura. Patrones y tácticas de diseño. Patrones de diseño orientados a objetos. Patrones y tácticas para desempeño (performance): concurrencia, replicación, cache, ajustes de uso de recursos, etc. Patrones y tácticas para disponibilidad: excepciones, pulso (heart-beat), replicación, etc. Patrones y tácticas para confiabilidad: votación, redundancia, excepciones, etc. Desarrollo guiado por pruebas. Reglas de buena documentación de arquitectura. Documentación basada en vistas. Reuso. Condiciones de reuso. Economía del reuso. Aspectos organizacionales. Reuso de componentes comerciales y de software libre. Frameworks.

5. **PROGRAMA ANALÍTICO**

Unidad 1: Diseño de sistemas

Concepto de diseño. Criterios, reglas y principios de diseño y especificación. Técnicas de documentación. Arquitectura y diseño de sistemas. Sistemas hardware y software.

Unidad 2: Diseño y Arquitectura

Diseño y Arquitectura en el ciclo de vida de desarrollo de software. Modelos de proceso de desarrollo y ciclos de vida. Estructuras de un sistema. Perspectiva histórica.

Unidad 3: El proceso unificado como método de desarrollo orientado a objetos

Orientación a objetos en el proceso de desarrollo y su vínculo con la programación orientada a objetos. Actividades, roles y artefactos del proceso unificado.

Unidad 3: Atributos de calidad

Soluciones de compromiso involucrando múltiples atributos de calidad. Relaciones entre atributos de calidad. Tácticas de diseño por atributo de calidad. Método de diseño guiado por atributos de calidad. Patrones de diseño como tácticas. Flexibilidad del software.

Unidad 4: Concurrencia

Nociones de concurrencia. Aplicaciones de concurrencia en lenguajes.

Unidad 5: Desarrollo guiado por pruebas

Desarrollo guiado por pruebas. Teoría y práctica.

Unidad 6: Reuso

Conceptos de Reuso. Condiciones de reuso. El caso de negocio del reuso. Frameworks. Experiencias de reuso. Familias de productos, Arquitecturas Orientadas a Servicios, reuso de librería. El caso del mundo de software libre.

6. **BIBLIOGRAFÍA BÁSICA**

Meyer, Bertrand, Construcción de Software Orientada a Objetos, Prentice-Hall, 1985, 2da. Edición 1997.

Gamma, Erich, Johnson, Ralph, Helm, Richard, Vlissides, John, Design Patterns, Elements of Reusable Object-Oriented Software, Addison-Wesley, 1994.

Bass, Len, Clements, Paul, Kazman, Rick, Software Architecture in Practice, SEI Series in Software Engineering, Addison-Wesley, 1997.

Ambler, Scott, Open UP, <http://epf.eclipse.org/wikis/openup/index.htm>, accedido en Octubre de 2012
Larman, Craig, UML y Patrones, Prentice-Hall, 2004.

7. BIBLIOGRAFÍA DE CONSULTA

Garlan, David, Shaw, Mary, An Introduction to Software Architecture, Carnegie Mellon University, 1994. Also published as "An Introduction to Software Architecture," Advances in Software Engineering and Knowledge Engineering, Volume I, edited by V.Ambriola and G.Tortora, World Scientific Publishing Company, New Jersey, 1993.

Fowler, Martin, Who Needs an Architect, IEEE Software, 3, 2003.

8. METODOLOGÍA DEL APRENDIZAJE

8.a DESARROLLO DE LA ASIGNATURA

La materia tiene un claro sesgo técnico, su foco está en el desarrollo de habilidades de diseño, en particular del método orientado a objetos tomado como referencia.

Con respecto a las técnicas de análisis y diseño, se muestran las técnicas del análisis y diseño estructurado y otras técnicas apropiadas para fomentar una mentalidad de "herramientas válidas de acuerdo al contexto" como opuesta a "si su única herramienta es un martillo, considere todo tornillo como un clavo". Con ese objetivo, se fomenta la constante comparación entre técnicas y la clara comprensión de su contexto de aplicabilidad.

Finalmente, se realizan múltiples actividades participativas en todas las clases para transmitir la naturaleza social y colaborativa de las actividades de análisis y diseño. Los alumnos realizan además la implementación de un sistema hardware-software, un ascensor, basado en la tecnología Lego Mindstorms, que les permite construir y probar el software especificado.

8.b DINÁMICA DEL DICTADO DE LAS CLASES

Clases teóricas y prácticas en aula con Pcs.

Las partes teóricas de las clases incluyen:

- Presentaciones teóricas basadas en diapositivas.
- Círculos de discusión
- Lectura compartida en voz alta

Las actividades prácticas incluyen:

- Actividades de modelado.
- Actividades colaborativas. Por ejemplo, proponer tácticas para resolver un problema.
- Presentaciones orales de trabajos realizados.
- Resolución colaborativa de problemas en el pizarrón.
- Trabajo práctico en grupo.

8.c TRABAJOS PRÁCTICOS

8.c.i ASPECTOS GENERALES.

Se efectuarán dos tipos diferentes de trabajos prácticos, resolución colaborativa de problemas en clase y un trabajo integrador grupal.

8.c.ii ASPECTOS PARTICULARES.

La materia incluye un trabajo práctico integrador que se desarrolla en entregas parciales durante toda la materia.

Consiste en la implementación de punta a punta de un sistema hardware-software, un ascensor construido con el sistema Lego Mindstorms.

9. METODOLOGÍA DE EVALUACIÓN

9.a NORMAS DE EVALUACIÓN.

La materia consta de dos parciales y un final. Los alumnos deben además desarrollar un ensayo individual sobre algunos de los temas de la materia.

9.b RÉGIMEN DE APROBACIÓN DE LA MATERIA.

Los alumnos deben aprobar ambos parciales con 4. La materia se promociona con un promedio de 7 en ambos parciales. La nota del ensayo se promedia con las notas de los parciales para determinar la nota de cursada. Ambos parciales cuentan con un recuperatorio, más un recuperatorio general de ambos.

10. PLANIFICACIÓN

CALENDARIO DE CLASES Y EVALUACIONES	
Semana 1	Introducción. Conceptos de Diseño. Análisis y Diseño. Diseño Orientado a Objetos. Actividad #1: analizar cómo el paradigma de objetos cumple (o no) con los puntos vistos.
Semana 2	Diseño y arquitectura (diseño a distintos niveles). - Estructuras de un sistema: Código, Tiempo de Ejecución, Física. - Métodos, Procesos Modelos de Ciclo de vida. - Análisis y Diseño orientado a objetos. - Historia del análisis y diseño oo Actividad #2: Discusión ¿Cómo se vinculan el AyDOO con la Programación OO?
Semana 3	Un Método de análisis y diseño oo: RUP Ejercicio de sistema de información: Modelo de Dominio de Twitter y Editor de Textos
Semana 4	UP (Proceso Unificado): Roles, artefactos y flujos de trabajo
Semana 5	UP (Proceso Unificado): Analisis de robustez Revisión del modelo de dominio del Ascensor
Semana 6	Atributos de Calidad Trade Off Study. Case Representation.
Semana 7	Patrones y tácticas de diseño Role play de patrones
Semana 8	Patrones y tácticas de diseño orientadas a objetos Role play de patrones
Semana 9	Estilos y patrones de arquitectura (1) y modelo de robustez

Semana 10	Estilos y patrones de arquitectura (2) Kata
Semana 11	Cierre del proceso, gap entre el análisis de robustez-arquitectura-diseño de clases
Semana 12	Concurrencia
Semana 13	TDD, mocking y xUnit test patterns
Semana 14	Conceptos de Reuso. Condiciones de reuso. El caso de negocio del reuso.
Semana 15	Experiencias de reuso. Familias de productos, Arquitecturas Orientas a Servicios, reuso de librería. El caso del mundo de software libre.
Semana 16	Cierre: Espacio abierto, perspectivas y evaluación conjunta de la materia
Del al de	FINAL

Información de Versiones	
Nombre del Documento:	Ficha Académica de la asignatura Diseño y Arquitectura de Sistemas de Computación
Nombre del Archivo	Diseño y Arquitectura de Sistemas de Computación – Plan 2013.docx
Documento origen:	-
Elaborado por:	Diego Fontdevila
Revisado por:	Alejandro Oliveros
Aprobado por:	Alejandro Oliveros
Fecha de Elaboración:	03-07-2013
Fecha de Revisión:	04-07-2013
Fecha de aprobación	04-07-2013
Versión:	1.0