



Ministerio de Cultura y Educación
 Universidad Nacional de San Luis
 Facultad de Ciencias Físico-Matemáticas y Naturales
 Departamento: Informática
 Area: Area IV: Pr. y Met. de Des. del Soft.

(Programa del año 2007)
 (Programa en trámite de aprobación)
 (Presentado el 08/10/2007 11:49:39)

I - Oferta Académica

Materia	Carrera	Plan	Año	Período
INGENIERIA DE SOFTWARE	TCO.UNIV.EN WEB	008/05	2	2c

II - Equipo Docente

Docente	Función	Cargo	Dedicación
RIESCO, DANIEL EDGARDO	Prof. Responsable	P.ADJ EXC	40 Hs
BAIGORRIA FERNANDEZ, LORENA S.	Prof. Co-Responsable	P.ADJ EXC	40 Hs
PERALTA, MARIO GABRIEL	Responsable de Práctico	JTP EXC	40 Hs
ABDELAHAD, CORINA NATALIA	Auxiliar de Práctico	A.1RA SEM	20 Hs
BRITOS MANRIQUE, LUIS EDUARDO	Auxiliar de Práctico	A.2DA SIM	10 Hs
LUZZA BONILLA, MARIANO GABRIEL	Auxiliar de Práctico	A.2DA SIM	10 Hs

III - Características del Curso

Credito Horario Semanal				
Teórico/Práctico	Teóricas	Prácticas de Aula	Práct. de lab/ camp/ Resid/ PIP, etc.	Total
Hs	3 Hs	4 Hs	2 Hs	9 Hs

Tipificación	Periodo
B - Teoria con prácticas de aula y laboratorio	2 Cuatrimestre

Duración			
Desde	Hasta	Cantidad de Semanas	Cantidad de Horas
06/08/2007	09/11/2007	14	126

IV - Fundamentación

Dar las bases teóricas y prácticas que permiten al Ingeniero de Software aplicar un método de desarrollo orientado a objetos utilizando herramientas graficas capaces de automatizar las actividades que se realizan durante el proceso de desarrollo del software.

V - Objetivos

Introducir al alumno en el desarrollo de sistemas aplicando métodos de desarrollo que permiten producir software de manera fiable, de calidad y que funcione en máquinas reales, cubriendo desde la especificación de requisitos hasta la obtención del producto, construyendo artefactos semi-formales.

VI - Contenidos

Unidad 1: El Modelo de Objetos

Abstracción. Encapsulamiento. Modularidad. Jerarquía. Herencia. Agregación. Tipificación. Estricta. Débil. Polimorfismo. Concurrencia. Persistencia. Estado. Comportamiento. Identidad. Visibilidad. Objeto. Interfaz e Implementación. Relaciones.

Unidad 2: El Lenguaje UML (Unified Modeling Language)

Introducción. Historia. Diagramas. Modelos. Importancia de los modelos. Modelos estáticos y dinámicos. Diagramas de

Clases. Clases: atributos, operaciones y responsabilidades. Relaciones: dependencia, generalización y asociación. Diagrama de Interacción: Secuencia y Colaboración. Modelo de Casos de Usos.

Uso de herramientas que soportan UML.

Unidad 3: Proceso Unificado: Requerimientos.

Introducción. Dirigido por Casos de Usos. Centrado en la Arquitectura. Iterativo e Incremental. Modelo de Casos de Usos. Captura de requisitos. Contexto del Sistema. Modelo del Dominio.

Unidad 4: UML Avanzado.

Mecanismos comunes. Estereotipos. Valores etiquetados. Restricciones. Diagrama de Objetos. Diagrama de Actividades. Máquinas de Estado. Modelo Arquitectónico. Componentes. Despliegue.

Unidad 5: Patrones de Diseño.

Introducción. Conceptos. Descripción. Selección de un patrón de Diseño. Utilización. Problema. Solución. Consecuencia. Catálogo de Patrones de Diseño. Patrones Creacionales. Patrones Estructurales. Patrones de Comportamiento.

Unidad 6: Proceso Unificado: Análisis y Diseño.

Introducción. Propósito. Diferencias. Artefactos. Modelo del Análisis. Clases del Análisis. Realización de Casos de Uso del Análisis. Paquetes del análisis. Arquitectura. Flujo de Trabajo. Rol del diseño. Artefactos. Modelo del Diseño. Clases del Diseño. Realización de Casos de Uso del Diseño. Subsistemas. Interfaz. Arquitectura. Modelo de Desarrollo. Flujo de Trabajo. Aplicación de Patrones en el Diseño.

Unidad 7: Proceso Unificado: Framework

Introducción. Distintas Instanciaciones del Proceso. Modelo de Negocio. Relación con los requisitos. Análisis. Modelo de análisis. Clases de análisis. Realización de casos de uso-análisis. Análisis de la arquitectura. Relación con el Diseño. Pasos a la implementación. Modelo de implementación. Componentes. Subsistemas de Implementación. Pruebas. Modelo de pruebas. Casos de prueba. Procedimiento de prueba. Plan de prueba. Utilización de herramientas como soporte de automatización del Proceso.

VII - Plan de Trabajos Prácticos

1. Modelado Estático con UML. Ingeniería Directa. Ingeniería Inversa con Java.
2. Modelado Dinámico con UML. Ingeniería Directa. Ingeniería Inversa con Java.
3. Patrones de Diseño.
4. Modelado Avanzado con UML.
5. Modelo de Casos de Usos.
6. Desarrollo de un sistema utilizando el Proceso con herramientas.

VIII - Regimen de Aprobación

Condiciones de regularización:

- Presentación y aprobación de los proyectos de práctico de máquina.
- Aprobación de dos exámenes parciales teórico-prácticos con nota mayor o igual a 6 (seis).

Condiciones de aprobación:

1. Por promoción, para lo cual el alumno debe contar con las condiciones de regularización y con una nota de exámenes parciales mayor o igual a 8 (ocho), ó,
2. Por examen final.

Alumnos Libres: Por las características propias del proyecto de práctico de máquina a desarrollarse durante todo el cuatrimestre, no se aceptan alumnos libres.

Alumnos que Trabajan: La cátedra se rige según las Ordenanzas CS N° 26/97, 15/00 y 13/03.

IX - Bibliografía Básica

- [1] El Proceso de Desarrollo de Software Unificado. Booch, Rumbaugh, Jacobson. Addison-Wesley, 1999.
- [2] The Unified Modeling Language User Guide, 2nd Edition. Booch, Rumbaugh, Jacobson. Addison-Wesley, 2005.
- [3] The Unified Modeling Language Reference Manual, 2nd Edition. Booch, Rumbaugh, Jacobson. Addison-Wesley, 2005.
- [4] Design Patterns: Elements of Reusable Object-Oriented Software. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides. Addison-Wesley. 1995.
- [5] Object-Oriented analysis and design with applications. Booch, Grady. The Benjamin/Cummings Publishing Company Inc. 1994.
- [6] Ingeniería del Software. Pressman, Roger S.Ed. Mc-Graw Hill. 2004.

X - Bibliografía Complementaria

- [1] UML Semantics. Booch, Rumbaugh, Jacobson, et al. <http://www.omg.org>
- [2] Object Constraint Language Specification. Booch, Rumbaugh, Jacobson. <http://www.omg.org>
- [3] UML Notation Guide. Booch, Rumbaugh, Jacobson. <http://www.omg.org>
- [4] UML y Patrones: Introducción al análisis y diseño orientado a objetos. Craig Larman, Prentice Hall, 1999.
- [5] Patterns in Java. Volume 1. A Catalog of Reusable Design Patterns Illustrated with UML. Mark Grand. John Wiley & Sons Inc. 1998.

XI - Resumen de Objetivos

Introducir al alumno en el desarrollo de sistemas aplicando métodos de desarrollo que permiten producir software de manera fiable, de calidad y que funcione en máquinas reales, cubriendo desde la especificación de requisitos hasta la obtención del producto, construyendo artefactos semi-formales.

XII - Resumen del Programa

- 1.- El Modelo de Objetos
- 2.- El Lenguaje UML (Unified Modeling Language)
- 3.- Patrones de Diseño.
- 4.- El Proceso Unificado

XIII - Imprevistos

.

ELEVACIÓN y APROBACIÓN DE ESTE PROGRAMA

ELEVACIÓN y APROBACIÓN DE ESTE PROGRAMA	
Profesor Responsable	
Firma:	
Aclaración:	
Fecha:	