Ministerio de Cultura y Educación **(Programa del año 2011)**
Universidad Nacional de San Luis **(Programa en trámite de aprobación)**
Facultad de Ciencias Físico Matemáticas y Naturales **(Presentado el 18/08/2011 09:53:29)**
Departamento: Informatica
Area: Area IV: Pr. y Met. de Des. del Soft.

I - Oferta Académica

Materia	Carrera	Plan	Año	Período
INGENIERIA DE SOFTWARE II	LIC.EN CS.DE LA COMPUTACION	006/05	2011	2° cuatrimestre

II - Equipo Docente

Docente	Función	Cargo	Dedicación
RIESCO, DANIEL EDGARDO	Prof. Responsable	P.Adj Exc	40 Hs
ABDELAHAD, CORINA NATALIA	Auxiliar de Práctico	A.1ra Semi	20 Hs
BERNARDIS, EDGARDO	Auxiliar de Práctico	A.2da Simp	10 Hs

III - Características del Curso

Credito Horario Semanal				
Teórico/Práctico	Teóricas	Prácticas de Aula	Práct. de lab/ camp/ Resid/ PIP, etc.	Total
Hs	2 Hs	4 Hs	2 Hs	8 Hs

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

Duración			
Desde	Hasta	Cantidad de Semanas	Cantidad de Horas
08/08/2011	18/11/2011	15	120

IV - Fundamentación

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

V - Objetivos / Resultados de Aprendizaje

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 tanto formales como semi-formales. Introducir a los alumnos a las Arquitecturas Orientadas a Servicios.

VI - Contenidos

Unidad 1: El Lenguaje UML

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

Unidad 2: Modelos en UML

Introducción. Modelos. Importancia de los modelos. Modelos estáticos. Clases: atributos, operaciones y responsabilidades. Relaciones: dependencia, generalización y asociación. Modelos estáticos y dinámicos. Diagrama de Interacción: Secuencia y Colaboración. Modelo de Casos de Usos.

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. Modelo de pruebas. Casos de prueba. Procedimiento de prueba. Plan de prueba.

Unidad 8: Especificacion usando OCL
Introducción. Componentes. Self. Invariantes. Propiedades. Clase Asociación. Colección y sus operaciones. Bolsa y sus operaciones. Conjunto. Secuencia. Pre y Post orientada a objeto. Relaciones derivadas.

Unidad 9: Procesos de Negocios.
Introducción. Modelado de Procesos de Negocios (BPM). BPMN. Objetos de Flujo. Actividades. Procesos. Subprocesos. Gateways. Eventos. Tipos. Artefactos. Objetos de Datos. Objetos de Conexión. Flujo de Secuencia. Flujos de Mensajes. Swinlanes. Pools. Líneas. Procesos Colaborativos B2B. BPEL

Unidad 10: Arquitectura Orientada a Servicios
Conceptos básicos relacionados con SOA y con sistemas orientados a servicios en general. SOA y BPM. XML. Esquemas XML. Estructura de Datos XML. Atributos. Subelementos. Espacio de Nombres. XPath. XQuery. Introducción al Protocolo de Acceso a Objetos Simple. Escenarios Básicos. Mensajes SOAP. Intercambio de Mensajes. Modelo de Procesamiento SOAP. Arquitectura Web y SOAP. Escenarios Avanzados. Introducción a WSDL (Web Service Definition Language). Servicios Web. Programación en la Nube. Arquitectura SOA.

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. Especificación usando OCL.
7. Desarrollo de un sistema utilizando el Proceso.

8. XML. XPath. XQuery.
9. BPMN.
10. Programación en la Nube.

VIII - Regimen de Aprobación

La materia se desarrolla con la modalidad de promoción sin examen final. Existen dos niveles:

a) Regularización solamente: Para regularizar la materia se deberá:
1.- Tener como mínimo un 80% de asistencia a clases prácticas.
2.- Tener los prácticos, solicitados por la cátedra, aprobados, como método aplicado por la cátedra para la evaluación continua del alumno.
3.- Presentación y aprobación del proyecto integrador de laboratorio con nota mayor o igual a 7 (siete).
4.- Aprobar dos parciales o sus respectivas recuperaciones con una nota mayor o igual a 6 (seis).

b) Promoción sin examen final: Para regularizar y aprobar la materia se deberá:
1.- Cumplir con los requisitos a.1, a.2 y a.3.
2.- Aprobar dos parciales o sus respectivas recuperaciones con una nota mayor o igual a 7 (siete).
3.- Aprobar una prueba final integradora con una nota mayor o igual a 7 (siete).
La nota final será la que surja de aplicar la siguiente fórmula:
NotaFinal = (max(p1, r1) + max(p2, r2) + pfi + pil) / 4
max : función maximo
p1 : primer parcial,
p2 : segundo parcial
r1 : recuperatorio del primer parcial
r2: recuperatorio del segundo parcial
pfi: prueba final integradora
pil : proyecto integrador de laboratorio.

Aquellos alumnos que sólo regularicen la materia deberán rendir un examen final, en los turnos establecidos.

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

Alumnos que Trabajan: Los alumnos que trabajan tienen una recuperación adicional, según las Ordenanzas CS Nº 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] The Object Constraint Language: Precise Modeling With UML. Addison-Wesley. Jos B. Warmer and Anneke G. Kleppe. 2004.
[7] Ingeniería del Software. Pressman, Roger S.Ed. Mc-Graw Hill. 2004.
[8] OpenUP/Basic, http://epf.eclipse.org/wikis/openupsp/, 2008.
[9] http://www.w3.org/standards/xml/
[10] http://www.omg.org/spec/BPMN/

X - Bibliografia Complementaria

[1] UML Semantics. http://www.omg.org
[2] Object Constraint Language Specification. http://www.omg.org
[3] UML Notation Guide. 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 Illustred 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 tanto formales como semi-formales. Introducir a los alumnos a las Arquitecturas Orientadas a Servicios.

XII - Resumen del Programa

1.- El Lenguaje UML
2.- Modelos en UML
3.- Proceso Unificado: Requerimientos.
4.- UML Avanzado.
5.- Patrones de Diseño.
6.- Proceso Unificado: Análisis y Diseño.
7.- Proceso Unificado: Framework
8.- Especificacion usando OCL
9.- Procesos de Negocios.
10.- Arquitectura Orientada a Servicios

XIII - Imprevistos

.

XIV - Otros

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