



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

(Programa del año 2010)
(Programa en trámite de aprobación)
(Presentado el 21/08/2010 18:28:24)

I - Oferta Académica

Materia	Carrera	Plan	Año	Período
INGENIERIA DE SOFTWARE I	PROF.EN CS.DE LA COMPUTACIÓN	06/08	2010	1° cuatrimestre
INGENIERIA DE SOFTWARE I	LIC.EN CS.DE LA COMPUTACION	006/05	2010	1° cuatrimestre

II - Equipo Docente

Docente	Función	Cargo	Dedicación
FUNES, ANA MARIA	Prof. Responsable	P.Adj Exc	40 Hs
GARIS, ANA GABRIELA	Responsable de Práctico	JTP Semi	20 Hs
ABDELAHAD, CORINA NATALIA	Auxiliar de Práctico	A.1ra Semi	20 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	3 Hs	2 Hs	8 Hs

Tipificación	Periodo
B - Teoria con prácticas de aula y laboratorio	1° Cuatrimestre

Duración			
Desde	Hasta	Cantidad de Semanas	Cantidad de Horas
15/03/2010	25/06/2010	15	120

IV - Fundamentación

Se introduce al alumno en el desarrollo de software como una actividad ingenieril y al trabajo en equipo. Teniendo en cuenta que es la primer asignatura relacionada a dicho tema, se pretende que el alumno aprenda los fundamentos básicos de cada una de las etapas que comprende el proceso de desarrollo de software así como las distintas técnicas y metodologías aplicables, con especial énfasis en el desarrollo de especificaciones tanto formales como semi-formales, conocimientos de los principios de diseño así como de las principales técnicas de validación y verificación del software.

V - Objetivos / Resultados de Aprendizaje

Al finalizar la asignatura el alumno debe ser capaz de haber asimilado los conceptos de naturaleza y cualidades del software, principios de la Ingeniería del Software, el proceso de desarrollo de software, especificaciones, diseño, validación y verificación, y métricas del software, así como haberlo introducido en el uso de tecnologías de CASE.

VI - Contenidos

Bolilla 1: La Ingeniería de software.
Aspectos históricos. Crisis del software. El modelo tradicional de ciclo de vida de la Ingeniería de Software. El paradigma orientado a objetos. Fortalezas y debilidades de ambos paradigmas.

Bolilla 2: Modelos de ciclo de vida del software.

Modelo teórico. Modelo incremental e iterativo. Modelo en cascada. Modelo de desarrollo de prototipo. Modelos evolutivos. Modelo Incremental. Modelo en espiral. Otros modelos.

Bolilla 3: Especificaciones informales.

Especificaciones de software. Modelos del análisis clásico. Modelo de datos: diagramas de Entidad-Relación. Modelo de Comportamiento: diagramas de transición de estados. Modelo de procesos: diagramas de flujos de datos. Diccionario de datos.

Bolilla 4: Métodos Formales.

Los Métodos Formales en la Ingeniería de Software. Aplicabilidad. Estilos de desarrollo formal. Estilos de especificaciones. Grados de rigor. Pros y contras.

Bolilla 5: Especificaciones formales.

Especificaciones orientadas al comportamiento. Máquinas de estados finitos. Redes de Petri. Especificaciones orientadas a la propiedad y al modelo. Abstracción de datos y operaciones. Tipos y valores. Axiomas.

Bolilla 6: Diseño.

El proceso de Diseño. Abstracción. Refinamiento. Modularidad. Módulos. Cohesión y acoplamiento. Encapsulación de datos. Tipos de datos abstractos. Ocultamiento de la información. Objetos. Cohesión y acoplamiento de objetos.

Bolilla 7: Validación y verificación del software.

Conceptos de validación y verificación del software. El proceso de prueba (Testing). Estrategias: prueba de unidad, prueba de integración, prueba de validación, prueba de sistema. Prueba funcional y prueba estructural. JUnit.

Bolilla 8: Métricas del software.

Calidad del software. Métricas del modelo de Analisis. Métricas del modelo de Diseño. Métricas del código fuente. Métricas para pruebas. Métricas de mantenimiento.

VII - Plan de Trabajos Prácticos

Practico 1: La Ingeniería de Software y los modelos de ciclo de vida del software.

Práctico 2: Especificaciones de software informales- Modelos del Análisis Clásico.

Práctico 3: Especificaciones de software formales I: Maq. de Estado Finitas y Redes de Petri.

Práctico 4: Especificaciones de software formales II: RSL.

Práctico 5: Pruebas Estructural.

Práctico 6: Pruebas Funcional.

Práctico 7: Métricas del Software

Práctica de Máquina I: Modelos del Análisis Clásico

Práctico de Máquina II: Especificación formal de un caso de estudio haciendo uso del lenguaje de especificación formal RSL.

VIII - Regimen de Aprobación

Condiciones para regularizar la asignatura:

- Haber asistido al menos al 80% de las clases de la asignatura.

- Haber aprobado los prácticos de máquina con toda su documentación entregada en tiempo y forma, más dos exámenes parciales escritos o sus respectivas recuperaciones, con notas mayores o iguales a seis.
- En caso de solo regularizar el alumno deberá rendir un examen final para aprobar la asignatura.

Condiciones para promocionar la asignatura:

- Haber asistido al menos al 80% de las clases de la asignatura.
- Haber aprobado los prácticos de máquina con toda su documentación entregada en tiempo y forma, más dos exámenes parciales escritos o sus respectivas recuperaciones, con notas mayores o iguales a siete.
- Haber aprobado la evaluación final integradora escrita con nota mayor o igual a siete.
- En caso de promocionar la materia el alumno aprobará la misma con una nota que surgirá del promedio entre los parciales, prácticos de máquina y evaluación final integradora.

Exámenes Libres:

Se admitirán alumnos libres. El alumno deberá aprobar en una primera instancia un examen de la parte práctica de la asignatura con una nota mayor o igual a seis para así poder rendir un examen teórico de la misma, del cual surgirá la nota definitiva.

IX - Bibliografía Básica

- [1] Pressman, Roger, "Ingeniería del Software - Un enfoque práctico", Mc Graw Hill, 5ta Edición.2002.
- [2] Sommerville, Ian; "Software Engineering", Addison-Wesley, 5ta Edición. 1996.
- [3] Ghezzi, Carlo y otros; "Fundamentals of Software Engineering", Prentice Hall, 1991.
- [4] Jorgensen, Paul C.; "Software Testing, a Craftsman's Approach", CRC Press, 1995.
- [5] The RAISE Language Group; "The RAISE Specification Language", Prentice-Hall, 1992.

X - Bibliografía Complementaria

- [1] Yourdon, Edward; "Análisis Estructurado Moderno", Prentice-Hall Hispanoamericana, 1993.
- [2] Bolaños, Javier. "Pruebas de Software y JUnit".Pearson Education, 2008.
- [3] Pankaj Jalote. "An Integrated Approach to Software Engineering, 3rd ed.", Springer, 2005.
- [4] Stephen Schach. "Ingeniería de Software Clásica y Orientada a Objetos, 6ta ed.", Mc Graw Hill, 2005.
- [5] The RAISE Method Group; "The RAISE Development Method", Prentice-Hall, 1995.

XI - Resumen de Objetivos

Al finalizar la asignatura el alumno debe ser capaz de haber asimilado los conceptos de naturaleza y cualidades del software, principios de la Ingeniería del Software, el proceso de desarrollo de software, especificaciones, diseño, validación y verificación del software, así como haberlo introducido en el uso de tecnologías de CASE.

XII - Resumen del Programa

La Ingeniería de software. Modelos de ciclo de vida del software. Especificaciones de software informales. Modelos del análisis clásico. Métodos Formales de desarrollo de software. Especificaciones orientadas al comportamiento. Especificaciones orientadas a la propiedad y al modelo. El proceso de Diseño. Validación y verificación del software. Testing. Métricas.

XIII - Imprevistos

.

XIV - Otros

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