



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 2018)  
 (Programa en trámite de aprobación)  
 (Presentado el 06/09/2018 12:49:43)

### I - Oferta Académica

Materia	Carrera	Plan	Año	Período
INGENIERIA DE SOFTWARE I	ING. INFORM.	026/1 2- 08/15	2018	2° cuatrimestre

### II - Equipo Docente

Docente	Función	Cargo	Dedicación
FUNES, ANA MARIA	Prof. Responsable	P.Asoc Exc	40 Hs
SANCHEZ, ALEJANDRO	Responsable de Práctico	JTP Semi	20 Hs
BERNARDIS, EDGARDO	Auxiliar de Práctico	A.1ra Simp	10 Hs
MIRANDA, ENRIQUE ALFREDO	Auxiliar de Práctico	JTP 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	1 Hs	2 Hs	5 Hs

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

Duración			
Desde	Hasta	Cantidad de Semanas	Cantidad de Horas
06/08/2018	16/11/2018	15	75

### 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 primera 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 la aplicación de un método de desarrollo particular, y de las principales técnicas de validación y verificación del software.

### V - Objetivos / Resultados de Aprendizaje

Introducir los conceptos y herramientas básicos correspondientes a la producción de software con enfoque industrial.

### VI - Contenidos

#### Unidad 1: La Ingeniería de Software

Aspectos históricos. Crisis del software. Software e Ingeniería del Software. Tendencias en la Ingeniería del Software. El paradigma clásico y orientado a objetos. Fortalezas y debilidades de ambos paradigmas.

#### Unidad 2: El proceso de Desarrollo de Software

Modelos de ciclo de vida del software. Modelo teórico. Modelo en cascada. Modelo de construcción de prototipos. Modelo incremental e iterativo. Modelos evolutivos. Modelo en espiral. Otros modelos. Actividades de Proceso. CASE: automatización.

### **Unidad 3: Modelado Básico del Comportamiento**

Casos de Uso. Términos y conceptos. Caso de uso: actores, flujos de eventos, escenarios. Organización de los casos de uso. Características. Diagramas de casos de uso. Términos y conceptos. Propiedades. Contenidos. Usos. Técnicas comunes de modelado: modelado del contexto del sistema, modelado de los requisitos de un sistema.

### **Unidad 4: Modelado del Dominio**

Modelo Conceptual o Modelo del Dominio. Elementos básicos de los Diagramas de Clases UML para la construcción de modelos conceptuales: clases, atributos, asociaciones, multiplicidades, clase asociación, asociaciones n-arias, generalizaciones.

### **Unidad 5: Interacción Hombre-Máquina**

Conceptos Básicos de Interfaces Hombre-Máquina (IHM). Importancia del diseño de IHM. La relación de la disciplina del diseño de interfaces a la ciencia de la Interacción Hombre- computadora. Sistemas interactivos y de diseño de interfaces. Tipos de diálogo. Técnicas de diálogo. Especificación del diálogo Hombre-Máquina. Diseño centrado en el usuario, diseño participativo. Análisis de la tarea. Prototipos y maquetas. Herramientas para crear prototipos. El concepto de usabilidad. Medidas de la calidad de la interfaz de usuario. Métodos para la observación y evaluación.

### **Unidad 6: Validación y Verificación del Software.**

Conceptos de validación y verificación del software. Inspecciones de software. Análisis estático automatizado. Introducción a la verificación formal. El proceso de prueba (Testing). Estrategias: prueba de unidad, prueba de integración, prueba de validación, prueba de sistema. Prueba funcional y prueba estructural. Automatización de las pruebas. JUnit.

### **Unidad 7: Métricas del Software.**

Introducción a la calidad del software. Métricas del modelo de Análisis. Métricas del modelo de Diseño. Métricas del código fuente. Métricas para pruebas. Métricas de mantenimiento. Estimación de Proyectos de Software. Puntos de Función. Puntos de Objetos.

## **VII - Plan de Trabajos Prácticos**

Práctico 1: La Ingeniería de Software y los modelos de ciclo de vida del software (aula).

Práctico 2: Modelado Básico del Comportamiento - Diagramas de Casos de Uso (aula y laboratorio).

Práctico 3: Modelado del Dominio con Diagramas de Clases (aula y laboratorio).

Práctico 4: Interacción Hombre - Máquina (aula y laboratorio).

Práctico 5: Prueba Estructural (aula y laboratorio).

Práctico 6: Prueba Funcional (aula y laboratorio).

Práctico 7: Prueba con JUnit (laboratorio).

Práctico 8: Métricas del Software (aula).

## **VIII - Regimen de Aprobación**

Condiciones para REGULARIZAR la asignatura:

- Haber asistido al menos al 70% de las clases teóricas y prácticas de la asignatura.
- Haber aprobado los trabajos prácticos solicitados por la cátedra con nota mayor o igual a 6.
- Haber aprobado un examen integrador o al menos una de sus dos recuperaciones con nota mayor o igual a 6.

Condiciones para PROMOCIONAR la asignatura:

- Haber asistido al menos al 80% de las clases teóricas y prácticas de la asignatura.
- Haber aprobado los trabajos prácticos solicitados por la cátedra con nota mayor o igual a 7.
- Haber aprobado un examen integrador o al menos una de sus dos recuperaciones con nota mayor o igual a 7.
- En caso de obtener nota de promoción, el alumno aprobará la materia con una nota que surgirá del promedio de las mejores

notas obtenidas de cada una de las evaluaciones.

Recuperaciones:

-Se otorgarán dos recuperaciones para el examen integrador.

Examen Final:

En caso de regularizar la materia, el alumno deberá rendir un examen final, el cual podrá ser oral o escrito, en cualquiera de los turnos de examen establecidos.

Exámenes Libres:

Dada las características prácticas de la asignatura, no se admitirán exámenes de alumnos libres.

## **IX - Bibliografía Básica**

- [1] Pressman, Roger; "Ingeniería del Software - Un enfoque práctico", Mc Graw Hill, 7ma Edición. 2010.
- [2] Booch, Rumbaugh, Jacobson. The Unified Modeling Language User Guide, 2nd Edition. Addison-Wesley, 2005.
- [3] Booch, Rumbaugh, Jacobson. The Unified Modeling Language Reference Manual, 2nd Edition. Addison-Wesley, 2005.
- [4] Jorgensen, Paul C.; "Software Testing, a Craftsman's Approach", CRC Press, 1995.
- [5] Daniel Bolaños y otros; "Pruebas de software y JUnit", Prentice-Hall, 2008.
- [6] Apuntes de la Cátedra

## **X - Bibliografía Complementaria**

- [1] Pressman, Roger, "Ingeniería del Software - Un enfoque práctico", Mc Graw Hill, 5ta Edición. 2002.
- [2] Sommerville, Ian; "Software Engineering", Addison-Wesley, 8va edición. 2007.
- [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] Norman E. Fenton & Shari Lawrence Pfleeger. "Software Metrics, A Rigorous and Practical Approach, 2nd edition", 1997.

## **XI - Resumen de Objetivos**

Introducir los conceptos y herramientas básicos correspondientes a la producción de software con enfoque industrial.

## **XII - Resumen del Programa**

La Ingeniería de software. Modelos de ciclo de vida del software. Modelado Básico del Comportamiento. Modelado del Dominio. Interacción Hombre - Máquina. Validación y verificación del software. Métricas del software. Estimaciones.

## **XIII - Imprevistos**

--

## **XIV - Otros**

--

**ELEVACIÓN y APROBACIÓN DE ESTE PROGRAMA**

**Profesor Responsable**

Firma:

Aclaración:

Fecha: