



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 2021)
(Programa en trámite de aprobación)
(Presentado el 03/09/2021 17:39:58)

I - Oferta Académica

Materia	Carrera	Plan	Año	Período
INGENIERIA DE SOFTWARE I	ING. INFORM.	026/1	2- 2021	2° cuatrimestre
		08/15		

II - Equipo Docente

Docente	Función	Cargo	Dedicación
FUNES, ANA MARIA	Prof. Responsable	P.Asoc Exc	40 Hs
SANCHEZ, ALEJANDRO	Prof. Colaborador	P.Adj Exc	40 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	3 Hs	1 Hs	6 Hs

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

Duración			
Desde	Hasta	Cantidad de Semanas	Cantidad de Horas
23/08/2021	26/11/2021	14	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 con la temática, se pretende que el alumno aprenda los fundamentos básicos de cada una de las etapas que comprenden 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 al alumno en los conceptos fundamentales de la Ingeniería de Software con hincapié en los conceptos y herramientas correspondientes a la producción de software con un enfoque de calidad. En particular, se busca profundizar en las primeras etapas del ciclo de vida (captura y análisis de requisitos), además de introducir al alumno en conceptos de validación y verificación, y de métricas del software.

VI - Contenidos

Unidad 1: Conceptos de Software e Ingeniería de Software

Aspectos históricos. Crisis del software. Software e Ingeniería del Software. Tendencias en la Ingeniería del Software.

Cuestionamientos e intereses de la Ingeniería del Software. Aspectos claves de la Ingeniería del Software. Características del software. Dominios de aplicación.

Unidad 2: El proceso de Desarrollo de Software

Modelos de ciclo de vida del software. Modelo teórico. Actividades del Proceso. Especificación. Diseño. Implementación. Validación. Evolución. Modelo en cascada. Modelo de construcción de prototipos. Modelo incremental e iterativo. Modelos evolutivos. Modelo en espiral. Metodologías ágiles. Desarrollo basado en componentes. Métodos formales. Proceso Unificado. Herramientas y técnicas para modelado de procesos. Ingeniería de Software Asistida por computadora (CASE).

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, modelado de los procesos del negocio.

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: Modelo de Presentación o Interfaces

Sistemas interactivos y diseño de interfaces. La Interfaz de Usuario (IU). Tipos de diálogo. Técnicas de diálogo. Especificación del diálogo Hombre-Máquina. Descripción estática y dinámica. Diagrama de navegación de ventanas. El concepto de Usabilidad. La importancia y los beneficios del diseño de una IU usable. Heurísticas de usabilidad para el diseño de la IU.

Unidad 6: Validación y Verificación del Software

Conceptos de validación y verificación del software. Inspecciones de software. Análisis estático automatizado. Concepto de verificación formal. El proceso de prueba (Testing). Estrategias: prueba de unidad, prueba de integración, prueba de validación, prueba de sistema. Técnicas de Prueba: 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. Puntos de Casos de Uso.

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).

Práctico Integrador (laboratorio): Proyecto grupal de estimación del sistema y desarrollo de modelo de requisitos (modelo de casos de uso, modelo del dominio, modelo de interfaces) para un dominio dado.

VIII - Regimen de Aprobación

Condiciones para REGULARIZAR la asignatura:

- Haber asistido, de manera virtual o presencial (de acuerdo a lo determinado oportunamente), al menos al 70% de las clases

teóricas y prácticas de la asignatura.

- Haber aprobado el trabajo práctico integrador solicitado por la cátedra con nota mayor o igual a 6.
- Haber aprobado dos exámenes parciales o una de sus respectivas dos recuperaciones con nota mayor o igual a 6.

Condiciones para PROMOCIONAR la asignatura:

- Haber asistido, de manera virtual o presencial (de acuerdo a lo determinado oportunamente), al menos al 80% de las clases teóricas y prácticas de la asignatura.
- Haber aprobado el trabajo práctico integrador solicitado por la cátedra con nota mayor o igual a 7.
- Haber aprobado dos exámenes parciales o una de sus respectivas dos recuperaciones con nota mayor o igual a 7.
- Haber aprobado un examen global integrador 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 notas obtenidas de cada una de las evaluaciones.

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] Bolaños, Daniel et al. "Pruebas de software y JUnit", Prentice-Hall, 2008.
- [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] Jacobson, Ivar et al. Object Oriented Software Engineering. A Use Case Driven Approach. Addison Wesley, 1992.
- [5] Jorgensen, Paul C. "Software Testing, a Craftsman's Approach", 4ta edición, CRC Press, 2014.
- [6] Pressman, Roger. "Ingeniería del Software - Un enfoque práctico", Mc Graw Hill, 8va Edición. 2015.
- [7] Material producido por la Cátedra

X - Bibliografía Complementaria

- [1] Alianza Ágil, <http://www.agilealliance.org>
- [2] Beck, Kent, "Extreme Programming Explained". Addison-Wesley. 2004.
- [3] Brooks, Frederik P. - No Silver Bullet. Essence and Accidents in Software Engineering. IEEE Computer. Abril 1987.
- [4] Fenton, Norman E. & Fleeger, Shari Lawrence. "Software Metrics, A Rigorous and Practical Approach, 2nd edition", 1997.
- [5] Pankaj Jalote. "An Integrated Approach to Software Engineering, 3rd ed.", Springer, 2005.
- [6] Schach, Stephen. "Ingeniería de Software Clásica y Orientada a Objetos, 6ta ed.", Mc Graw Hill, 2005.
- [7] Sommerville, Ian; "Software Engineering", Addison-Wesley, 9na edición. 2011.

XI - Resumen de Objetivos

Introducir los conceptos y herramientas básicos correspondientes a la producción de software con un 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

El presente programa puede presentar ajustes dada la situación epidemiológica por COVID919. Toda modificación será acordada y comunicada con el estudiantado e informada a Secretaría Académica.

XIV - Otros

--

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