



UNIVERSIDAD NACIONAL DE CÓRDOBA
Facultad de Ciencias Exactas, Físicas y Naturales
República Argentina

Programa de:

Ingeniería del Software

Código: 2629

Carrera: *Ingeniería en Computación*

Escuela: *Ingeniería Electrónica y Computación.*

Departamento: *Computación.*

Plan: 285-05

Puntos: 4

Carga Horaria: 96

Hs. Semanales: 6

Semestre: *Séptimo*

Año: *Cuarto*

Carácter: *Obligatoria*

Objetivos:

Al finalizar el curso se espera que el alumno sea capaz de:

- *Conocer las ventajas de la Ingeniería de Software*
- *Reconocer las ventajas de la orientación a objetos aplicadas a la Ingeniería de Software*
- *Aprender y experimentar un método orientado a objetos para el desarrollo de sistemas.*

Programa Sintético:

- 1. Ingeniería de Software*
- 2. Orientación a Objetos*
- 3. Desarrollo de Sistemas Orientados a Objetos*
- 4. Proceso de Análisis*
- 5. Proceso de Construcción*

Programa Analítico: de foja 2 a foja 2.

Programa Combinado de Examen (si corresponde): de foja a foja .

Bibliografía: de foja 3 a foja 3.

Correlativas Obligatorias: *Algoritmos y Estructuras de Datos*

Correlativas Aconsejadas:

Rige: 2005

Aprobado HCD, Res.:

Modificado / Anulado /Sust. HCD Res.:

Fecha:

Fecha:

El Secretario Académico de la Facultad de Ciencias Exactas, Físicas y Naturales (UNC) certifica que el programa está aprobado por el (los) número(s) y fecha(s) que anteceden. Córdoba, / / .

Carece de validez sin la certificación de la Decretaría Académica:

LINEAMIENTOS GENERALES

La presente asignatura es una actividad curricular que pertenece al séptimo semestre de la carrera de Ingeniería en Computación.

El desarrollo de Sistemas de Software experimentó una gran expansión en las dos últimas décadas, debido a la confluencia de diversos factores entre los que debemos destacar.

- La necesidad de desarrollo de nuevas aplicaciones crece, frente a la demanda de generar información a los negocios que se adecuen a los contextos cambiantes de los mismos.
- La reducción del costo de Arquitecturas computacionales de muy altas prestaciones que facilitan y promueven la programación en paralelo, que implican rediseño de software existente y nuevo software.
- Metodologías, herramientas, procesos y tecnologías que facilitan la construcción de software.
- La necesidad de industrializar la construcción de Software basada en Componentes con tecnología Orientada a Objetos y su Documentación en UML,.

Enfocados en estos factores en esta asignatura se desarrollan y se aplican conceptos dentro de las dos líneas de trabajo:

- **Desarrollo de Sistemas de Software:** Ser capaz, a partir de un problema de adquirir los requerimientos, de diseñar, de implantar y de probar un sistema Orientado a Objetos.
 - Aprender como se se obtienen requerimientos de software
 - Aprender las actividades mas importante del proceso de diseño y construcción
 - Aprender las actividades y procesos de prueba
 - Hacer uso de los distintos modelos, su representación y documentación
 - Identificar, evaluar y resolver los problemas de diseño
- **Diseñar el proceso de Desarrollo Software:** Ser capaz de elegir la metodología a aplicar y planificar las tareas del proceso de desarrollo.
 - Aprender la Metodologías de desarrollo
 - Aprender a seleccionar la metodología mas adecuada a la naturaleza de la aplicación.
 - Aprender a realizar un plan de desarrollo de la aplicación y Diseño de la documentación del proyecto.
 - Estrategias para generar casos de pruebas

El dictado se orienta a capacitar al alumno para **Diseñar y desarrollar modelos de sistemas donde se hace uso y explotan las ventajas de los objetos**; para lograr sistemas con alta **efectividad donde los requerimientos son volátiles y flexibles**. Todo expresado documentalente en UML y realizado en un lenguaje de objetos.

METODOLOGIA DE ENSEÑANZA

Las clases impartidas son teóricas, prácticas y de laboratorio.

Las actividades teóricas se realizan a través de exposiciones del docente orientadas a desarrollar en los alumnos la capacidad de Saber sobre metodologías de desarrollo de software y su ámbito de aplicación.

Durante el desarrollo de los Trabajos Prácticos se realizan actividades que le permiten al estudiante poner en práctica las habilidades y verificar los criterios y técnicas de modelado, diseño y prueba.

Por otra parte en las clases de Laboratorio el alumno verifica, a través de la implementación, el funcionamiento de sistemas y realizara los casos de pruebas

EVALUACION

Condiciones para la promoción de la materia

1. Tener aprobadas las materias correlativas.-
2. Asistir al 80% de las clases teóricas y prácticas.-
3. Rendir y aprobar los dos parciales con 50% , el alumno podrá recuperar una vez cada parcial
4. Presentar y aprobar las implementaciones que se exijan durante el desarrollo de los trabajos prácticos.-
5. Aprobar un coloquio integrador

Los alumnos que cumplan las exigencias referidas en los puntos 1 al 5 serán considerados promocionados.

Los alumnos que cumplan las exigencias referidas en los puntos 1 al 4 serán considerados regulares.

El resto será considerado libre.

CONTENIDOS TEMATICOS

Unidad I. Ingeniería de Software (12 hs)

Objetivo: manejar el contexto de la Ingeniería del Software, el lenguaje y la terminología asociada a la disciplina.

Modulo 1. Introducción

Contenido: Introducción a al Ingeniería de Software. Preguntas frecuentes: ¿Que es software?, ¿Que es la Ingeniería del software?, ¿Cuál es la diferencia entre Ingeniería de Software y Ciencias de la Computación o Ingeniería de Sistemas?, ¿Cuales son los atributos de un buen software?.

Modulo 2. Sistemas Socio Técnicos

Contenido: Ingeniería de Sistemas, Definición de Requerimientos del Sistema, Diseño de Sistemas, Modelado de Sistemas.

Modulo 3. Sistemas Críticos

Contenido: Confiabilidad de un Sistema, Disponibilidad y fiabilidad de un Sistema, Seguridad, Protección.

Modulo 4. Procesos de Software

Contenido: Modelos de proceso de software: Modelo en cascada, Modelo Evolutivo, Modelo basado en Componentes. Iteración de procesos: Entrega incremental, Desarrollo en Espiral. Actividades del Proceso: Especificación del proceso, Diseño e implementación del software, Validación del software, evolución del software. Herramientas Case.

Unidad II. Requerimientos (18 hs)

Objetivo: manejar el contexto básico de la Ingeniería de Requerimientos, el lenguaje y la terminología asociada a esta disciplina.

Modulo 5. Requerimientos del Software

Contenido: Requerimientos funcionales, no funcionales y de dominio. Requerimientos de Usuario. Requerimientos de Sistema. Especificaciones de la Interfaz.

Modulo 6. Procesos de la Ingeniería de Requerimientos

Contenido: Estudios de Viabilidad. Adquisición y análisis de requerimientos. Validación de requerimientos. Gestión de requerimientos.

Unidad III. Modelado y Especificación (16 hs)

Objetivo: desarrollar habilidades y manejar las técnicas de modelado y especificación.

Modulo 7. Modelos del Sistema Software

Contenido: Modelo de Contexto. Modelo de Comportamiento. Modelo de datos. Modelo de Objetos. Métodos Estructurados.

Modulo 8. Especificación de Sistemas Críticos

Contenido: Especificación dirigida por riesgos. Especificación de la seguridad. Especificación de la protección. Especificación de la fiabilidad.

Unidad IV. Diseño y Construcción y Prueba (16 hs)

Objetivo: desarrollar habilidades y manejar las técnicas y herramientas de diseño y construcción.

Modulo 9. Diseño arquitectónico del Software

Contenido: Decisiones del diseño arquitectónico. Organización del sistema. Estilos de descomposición modular. Estilos de control. Arquitecturas de referencia.

Modulo 10. Diseño orientado a objetos

Contenido: Objetos y clases. Proceso de diseño orientado a objetos. Evolución del diseño. UML. Vistas multidimensionales del sistema.

Modulo 11. Diseño de interfaces de Usuario

Contenido: Asuntos de diseño. El proceso de diseño de la interfaz de usuario. Análisis del usuario. Prototipado de la interfaz de usuario. Evaluación de la Interfaz.

Modulo 12. Pruebas del Software

Contenido: Pruebas del sistema. Pruebas de Componentes. Diseño de casos de prueba. Unidades de Testing.

Unidad V. Calidad, Auditoria y Peritaje (10 hs)

Objetivo: desarrollar habilidades y manejar las técnicas y herramientas de testing.

Modulo 13. Introducción a la calidad

Contenido: Conceptos de la calidad. Normas IEEE, ISO y CMMI. Los niveles y Areas de proceso clave de CMMI.

Modulo 14. Introducción a la Auditoria

Contenido: Conceptos de la Auditoria. Funciones del control interno y auditoría informática. Metodología de control interno, seguridad y auditoría informática. Principales áreas de la auditoría informática. Seguridad Física y Seguridad Lógica.

Modulo 15. Introducción al Peritaje

Contenido: Conceptos del Peritaje. Pericia y Peritos. Clases de Peritos. Metodología. Quién puede ser Perito. Fuentes de Información. Pasos a seguir. Métodos y Técnicas utilizadas en peritaje informático.

Distribución de la carga horaria

en clase en el ambito de la facultad

Actividad	Horas
Teórica	42
Formacion Practica	
Formación experimental	9
Resolución de problemas	18
Actividades de proyecto y diseño	27
Total de la Carga horaria	96

Dedicadas por el alumno fuera de clase

Actividad	Horas
Teórica	42
Formacion Practica	
Formación experimental	12
Resolución de problemas	24
Actividades de proyecto y diseño	33
Total de la Carga horaria	111

BIBLIOGRAFIA Principal

- **Sommerville I.**, Ingeniería del Software, 2005, Prentice Hall (ISBN 8478290745).
- **Fowler, Martin.** UML gota a gota. — 1ra en español ed. — México,MX : Pearson Educación, 2000 . — 203 p. (ISBN: 968-444-364-1).
- **Gonzalo Cuevas Agustín.** Una Guía del CMM. Para Comprender el Modelo de Madurez de Capacidad del Software. Traducción del Ingles “A Guide to the CMM” de Kenneth M. Dymond. 1998.

BIBLIOGRAFIA Complementaria

- **Weitzenfeld, A.**, 2004, Ingeniería de Software Orientada a Objetos con UML, Java e Internet, Cengage (Thomson) Learning (ISBN 970-686-190-4).
- **Fowler, Martin.** UML Distilled: A Brief Guide to the Standard Object Modeling Language, © 2004 | Addison-Wesley | ISBN-10: 0321193687 (ISBN-13: 9780321193681).
- **Booch, Rumbaugh & Jacobson.** The Unified Modeling Language User Guide, 2/E © 2005 | Addison-Wesley | (ISBN-10: 0321267974 | ISBN-13: 9780321267979)
- **BERTRAND MEYER,** CONSTRUCCION DE SOFTWARE, Editorial: PRENTICE HALL (ISBN: 84-8322-040-7)