



DISEÑO DE SISTEMAS

PROGRAMA ANALÍTICO

PLAN DE ESTUDIOS	2008
ORDENANZA CSU. N°	1150
HORAS/AÑO:	192
OBLIGATORIA	X
ELECTIVA	
ANUAL	X
PRIMER CUATRIMESTRE	
SEGUNDO CUATRIMESTRE	
NIVEL / AÑO	3°
HORAS CÁTEDRA SEMANALES	6

OBJETIVOS

OBJETIVO GENERAL

1. Objetivo General de la Materia

Formar en el dominio de la metodología de Diseño de Sistemas y su aplicación profesional, permitiendo integrar los conocimientos de otras áreas de forma tal de dar significación a los mismos y desarrollar criterios tendientes a definir la idiosincrasia del Ingeniero en Sistemas de Información.

2. Objetivos expresados en Capacidades a lograr en los Alumnos:

- Conocer las metodologías, modelos, técnicas y lenguajes del proceso de Diseño.
- Elaborar modelos para el diseño de sistemas de información.
- Aplicar patrones al diseño de sistemas de información.
- Diseñar y construir productos de software asociado a los sistemas de información aplicando herramientas de soporte de diseño.

OBJETIVOS ESPECÍFICOS

OBJETIVOS DE LA UT N° 1. Introducción al Diseño de Sistemas

Conocer el propósito del Diseño de Sistemas, sus incumbencias y el rol del diseñador.

Comprender al Diseño de sistemas como proceso de resolución de problemas, construcción de alternativas de solución con calidad.

Estudiar las diferentes fases del Diseño y su participación en el Ciclo de Vida de Desarrollo de Software.

Conocer las distintas Metodologías para el Diseño de Software.





OBJETIVOS DE LA UT N° 2. Prototipado de Software y Ensamblaje de Componentes

Conocer el objetivo del prototipado, estudiar los diferentes tipos de prototipos.
Identificar los lineamientos para su desarrollo, ventajas y desventajas de su utilización.
Analizar las etapas del ciclo de vida con prototipado.
Remarcar el rol del Cliente / Usuario en el desarrollo de Prototipos.
Conocer el uso de componentes comerciales y no comerciales.

OBJETIVOS DE LA UT N° 3. Diseño Orientado a Objetos

Conocer diferentes procesos asociados al Desarrollo de Software.
Comprender los conceptos de Clases y Objetos y propiedades avanzadas, como así también las técnicas y heurísticas para identificar Clases de Dominio.
Remarcar el proceso de refinamiento y evolución de los modelos de clases en el avance del proceso de producción.
Modelar diagramas de Clases de Dominio y de Diseño, establecer diferencias del enfoque.
Elaborar Diagrama de Objetos.
Identificar el aporte de precisión y calidad al modelo de Diseño OO a través de la implementación del lenguaje OCL.
Entender el concepto de Patrón de Diseño, contribuciones de su uso en el Desarrollo de Software.
Analizar y comprender el uso de tipos específicos de patrones.
Demarcar la importancia de los conceptos de cohesión, acoplamiento, complejidad y reusabilidad aplicados al modelado OO.

OBJETIVOS DE LA UT N° 4. Diseño de la Interfaz externa

Conocer y diseñar los distintos tipos de salida del sistema software.
Establecer las características de salidas dirigidas a diferentes roles de usuarios.
Entender el diseño de la salida impresa, buenas prácticas para su construcción.
Denotar la importancia de la correcta captura de datos, estudiar técnicas de validación y codificación de los datos de entrada.
Comprender los lineamientos para un buen diseño de formularios de captura de información.
Conocer otras formas de captura de datos.
Entender los criterios definidos en las buenas prácticas de diseño de interfaz gráfica, diseño de ventanas y modelos de navegación.

OBJETIVOS DE LA UT N° 5. Arquitectura de Software

Comprender que es un modelo arquitectónico, su importancia y contenidos.
Conocer distintos estilos arquitectónicos y modelos que los representan.
Abordar los modelos más conocidos asociados a diferentes paradigmas:
Modelo de Eventos, descomposición modular, Flujo de Datos, OO y (SOA) Arquitectura orientada a servicios (SOA).
Estudiar patrones arquitectónicos.

OBJETIVOS DE LA UT N° 6. Pruebas de Sistemas

Establecer la importancia de las pruebas en el ciclo de vida del software como parte del plan de calidad.
Entender el alcance de las pruebas de Unidad y las pruebas de Integración.
Abordar diferentes tipos de pruebas y su aplicación.
Generar Casos de Prueba desde las especificaciones de los Casos de Uso en RUP.





Identificar la necesidad de actividades Validación y Verificación de Software como parte del plan de calidad: Normas Estándar (IEEE, ISO, CMMI)

OBJETIVOS DE LA UT N° 7. Implantación de Sistemas

Conocer las fases y actividades del proceso de implantación de un sistema en producción.
Comprender la gestión del plan de Capacitación de Usuarios.
Evaluar y rever el proceso Implantación.
Conocer la administración del mantenimiento, sus diferentes tipos e impactos.

OBJETIVOS DE LA UT N° 8. Arquitectura dirigida por modelos (MDA)

Conocer el paradigma MDA: Arquitectura dirigida por modelos, su contribución a la calidad de artefactos intermedios y del software, impacto en el proceso de producción y posterior del producto. Identificar a este modelo como eje fundamental del proceso de construcción del software desde un enfoque evolutivo.

OBJETIVOS DE LA UT N° 9. Diseño Web

Conocer las características de las Aplicaciones Web, establecer aspectos específicos del Proceso de Desarrollo de software Web.
Analizar el desarrollo de aplicaciones Web con metodologías ágiles

CONTENIDOS

CONTENIDOS SINTÉTICOS

- Actividades de Diseño.
- Patrones de Diseño.
- Diseño de Arquitectura.
- Verificación y Validación del Diseño.
- Documentación de las Diferentes Etapas del Diseño.
- Diseño de Interfaces. Diseño de Procedimientos.
- Estrategias de Prototipado y de Ensamblaje de Componentes.

CONTENIDOS ANALÍTICOS

UNIDAD TEMÁTICA N° 1. Introducción al Diseño de Sistemas

- i. Propósito del Diseño de Sistemas.
- ii. Rol y Perfil del Diseñador de Sistemas.
- iii. El Diseño como proceso de resolución de problemas.
- iv. El Diseño y la calidad de Software.
- v. Fases del Diseño: Diseño de datos, Desarrollo de Procesos, Diseño de Interfaz
- vi. Diseño lógico y Diseño físico.
- vii. Significado dentro del Ciclo de Vida de Desarrollo de Software.
- viii. Metodologías para el Diseño de Software.

TIEMPO ASIGNADO: 18 HORAS

UNIDAD TEMÁTICA N° 2. Prototipado de Software y Ensamblaje de Componentes

- i. Objetivos del Prototipado.
- ii. Tipos de Prototipos.
- iii. Lineamientos para el desarrollo de Prototipos
- iv. Etapas del ciclo de vida con prototipado.
- v. Ventajas y Desventajas de los Prototipos.





vi. El rol del Cliente / Usuario en el desarrollo de Prototipos.
TIEMPO ASIGNADO: 18 HORAS

UNIDAD TEMÁTICA N° 3. Diseño Orientado a Objetos

- i. Procesos de Desarrollo de Software.
- ii. Clases y Objetos. Propiedades avanzadas.
- iii. Identificación de Clases. Instancias.
- iv. Diagrama de Clases de Diseño
- v. Diagrama de Objetos
- vi. Especificación de la Interfaz de los Objetos
- vii. Restricciones OCL sobre el modelo de Diseño OO.
- viii. Introducción a los Patrones de Diseño: Historia. Ventajas y Desventajas del uso de Patrones en el Desarrollo de Software.
- ix. Tipos de Patrones: Patrones creacionales, Patrones Estructurales, Patrones de Comportamiento.
- x. Grasp: Patrones de Principios Generales para asignar responsabilidades: Experto – Creador – Bajo Acoplamiento – Alta cohesión – Controlador.
- xi. Medidas de cohesión, acoplamiento, complejidad y reusabilidad de módulos.
- xii. Frameworks.
- xiii. Refactoring.
- xiv. El Modelo de Clases y el Modelo de Datos.
- xv. Implementación del Modelo de Objetos sobre Bases de Datos Relacionales. Patrones de persistencia.
- xvi. El modelo de Ensamblaje de Componentes.

TIEMPO ASIGNADO: 50 HORAS

UNIDAD TEMÁTICA N° 4. Diseño de la Interfaz externa

- i. Diseño de Salida.
 1. Gráficos, Informes, Datos tabulados, Preimpresos.
 2. Otros tipos de salida
 3. Salida para Información Gerencial.
- ii. Captura de datos. Importancia.
 1. Lineamientos para la Codificación de Datos de Entrada. Tipos de Códigos.
 2. Validación de Datos de Entrada y de Transacciones de Entrada.
 3. Diseño de Formularios de Entrada.
 4. Otras formas de Captura
- iii. Diseño de la Interfaz Externa. Criterios.
 1. Diseño y Especificación de Ventanas.
 2. Diagrama de Navegación de Ventanas
 3. Evaluación de la Interfaz.

TIEMPO ASIGNADO: 24 HORAS

UNIDAD TEMÁTICA N° 5. Arquitectura de Software

- i. Diseño arquitectónico. Definición. El uso de modelos.
- ii. Estilos arquitectónicos
 1. Modelos de descomposición de sistemas.

- Almacén central.
- Cliente-Servidor.





- Máquinas abstractas.
- 2. Modelos de control.
 - Centralizado.
 - Modelo de Eventos.
- 3. Modelos de descomposición modular.
 - Modelo de Flujo de Datos.
 - Orientado a Objetos.
- 4. Modelos de dominio específico.
- iii. Arquitectura orientada a servicios (SOA).
- iv. Patrones arquitectónicos

TIEMPO ASIGNADO: 20 HORAS

UNIDAD TEMÁTICA N° 6. Pruebas de Sistemas

- i. La prueba en el ciclo de vida del software. Workflow de prueba.
- ii. Pruebas de Unidad. Pruebas de Integración.
- iii. Prueba Alfa. Prueba Beta.
- iv. Pruebas de Sistemas: Carga Máxima, Almacenamiento, Tiempo de Ejecución, Recuperación, Procedimientos, Factores Humanos
- v. Pruebas de regresión.
- vi. Generación de Casos de Prueba desde las especificaciones de los Casos de Uso.
- vii. Prueba de Clases de Objetos.
- viii. Integración de Objetos.
 - 1. Pruebas de Casos de Usos.
 - 2. Prueba de Cadena de Eventos.
 - 3. Prueba de interacción de objetos.
- ix. Validación y Verificación de Software: Normas Estándar (IEEE, ISO, CMMI)

TIEMPO ASIGNADO: 24 HORAS

UNIDAD TEMÁTICA N° 7. Implantación de Sistemas

- i. La Organización y el Plan de Implantación.
- ii. Conversión: Directa, en Paralelo, Enfoque piloto, por Etapas.
- iii. Capacitación de Usuarios.
- iv. Administración de Recursos
- v. Documentación del sistema.
- vi. Manuales de Procedimiento. Manuales del usuario. Ayuda on-line.
- vii. Carga de Datos. Conversión de Datos. Migración de Bases de Datos.
- viii. Proceso de Revisión y Evaluación de la Implantación.
- ix. Mantenimiento del software.
 - 1. Tipos de mantenimiento: Correctivo, Perfectivo, Adaptativo, Preventivo
 - 2. Costos del mantenimiento.
 - 3. Reingeniería de Procesos y de Software.

TIEMPO ASIGNADO: 18 HORAS

UNIDAD TEMÁTICA N° 8. Arquitectura dirigida por modelos (MDA)

- i. Introducción a MDA: Arquitectura dirigida por modelos. Visión integradora del proceso de producción de software y el carácter evolutivo del producto.

TIEMPO ASIGNADO: 10 HORAS





UNIDAD TEMÁTICA N° 9. Diseño Web

- i. Aplicaciones Web. Características.
- ii. Proceso de Desarrollo de software Web.
 1. Ingeniería de requisitos.
 2. Diseño conceptual.
 3. Diseño navegacional: Modelo de la estructura de navegación. Modelo del espacio navegacional.
 4. Diseño de presentación: Interfaz del usuario.
 5. Implementación.
 6. Testing.
- iii. Arquitectura de sistemas Web.
- iv. Desarrollo de aplicaciones Web con metodologías ágiles

TIEMPO ASIGNADO: 10 HORAS

BIBLIOGRAFÍA OBLIGATORIA

TITULO	AUTOR	EDITORIAL	AÑO DE EDICIÓN /ISBN	EJEMP. DISP.
INGENIERÍA DEL SOFTWARE	Ian Sommerville.	Pearson Education	9°Edición (2011)	-
INGENIERIA DEL SOFTWARE: Un enfoque práctico	Roger S. Pressman	MCGRAW-HILL	7°Edición (2010)	-
EL LENGUAJE UNIFICADO DE MODELADO (MANUAL DE REFERENCIA)	Booch, Rumbaugh, Jacobson	PEARSON EDUCACION	2°Edición (2007)	4
EL LENGUAJE UNIFICADO DE MODELADO	Booch, Rumbaugh, Jacobson	PEARSON EDUCACION	2°Edición (2006)	-
ANALISIS Y DISEÑO DE SISTEMAS	Kendall & Kendall	PEARSON ADDISON-WESLEY	6°Edición (2005)	12
UML Y PATRONES	Craig Larman	PEARSON ALHAMBRA	2°Edición (2003)	-
INGENIERÍA DE SOFTWARE: teoría y práctica	Shari Lawrence Pfleeger	Pearson Education	3°Edición (2002)	-

BIBLIOGRAFÍA COMPLEMENTARIA

- Arquitectura de Software Autor: Shaw-Garlan 1996
- Agile Software Development methods. Review and Analysis – P. Abrahamsson, O. Salo y otros - VTT Publications, 2002
- Agile Software Development - A. Cockbun – Addison-Wesley, 2001
- Agile & Iterative Development. A Manager's guide – C. Larman – Addison-Wesley, 2004.

TITULO	AUTOR	EDITORIAL	AÑO DE EDICIÓN /ISBN	EJEMP. DISP.
INGENIERIA DE SOFTWARE: Una perspectiva orientada a objetos	Braude Eric J.	Alfaomega Grupo Editor	1°Edición (2003)	-

DIRECCION ACADEMICA
ES COPIA FIEL DEL ORIGINAL



MARIA EUGENIA LAVORATTO
DIRECTORA
DIRECCION ACADEMICA
U.T.N. F.R.L.P.



INGENIERIA DE SOFTWARE ORIENTADO A OBJETOS	Bernd Bruegge, Allen H. Dutoit.	Pearson Education	1ª Edición (2002)	-
EL PROCESO UNIFICADO DE DESARROLLO DE SOFTWARE	Ivar Jacobson, Grady Booch, James Rumbaugh	Addison Wesley	1ª Edición (2000)	-

CARACTERÍSTICAS DE LA ACTIVIDAD CURRICULAR

DESCRIPCIÓN - MODALIDAD DE LA ENSEÑANZA

Los contenidos de la materia se presentan en forma teórica-práctica donde se abordan situaciones reales, con un criterio integrador, de revisión y asociación de los temas ya desarrollados en la asignatura o en otras materias vinculadas, de tal manera de crear una red de conocimientos significantes con un fuerte enfoque en lo pragmático.

Las actividades prácticas son elaboradas y consensuadas por los profesores de la Cátedra en forma conjunta. Son las mismas para todas las comisiones y el alumno dispone de ellas en papel a través de la fotocopidora o en formato digital en el sitio de la Cátedra.

Estas se realizan en gabinete de computación donde se desarrollan en forma grupal, se confrontan alternativas de solución, enfoques y se produce una solución que se analiza y se corrige en conjunto. Las herramientas que se utilizan son:

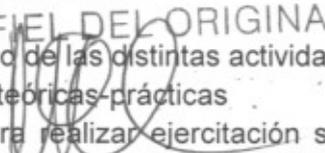
CASE Enterprise Architect para representar las diferentes visiones del modelado del software, trazabilidad entre modelos y seguimiento del proyecto, y un Generador de Reportes. Además de los trabajos prácticos propuestos, los alumnos realizan tres trabajos:

- a. Actividad de lecto-comprensión de Papers en castellano de autores de distintas nacionalidades, publicación reciente y relacionados al con el campo de la Ingeniería de Software, con el mismo deben elaborar una síntesis del trabajo en filminas y exposición verbal de mismo para compartir con sus compañeros los conceptos abordados y la propuesta del autor, el desarrollo, conclusiones y enlaces futuros. Este trabajo es individual y cuenta con un plan de gestión disponible en el sitio virtual de la cátedra. (1er. Cuatrimestre). Al final del proceso los alumnos llenan una encuesta sobre su apreciación del trabajo, detalles técnicos y tutoría del profesor para retroalimentar el proceso y mejorarlo en próximas ediciones.
- b. Trabajo grupal de integración con la Cátedra de Inglés referido a la lectura y traducción de un trabajo en inglés referido a un tema actual de la disciplina, que consiste en una guía de filminas en inglés que deberán ser traducidas y adecuadas para transmitir la idea correctamente. El producto será una guía de filminas en castellano describiendo sintéticamente el trabajo original.
- c. Trabajo grupal de investigación y desarrollo de diferentes Metodologías del Proceso de Desarrollo de sistemas con elaboración y entregas pautadas donde el profesor asigna los temas, monitorea el proceso de producción y los productos entregados. Luego la producción se difunde en el lugar de la cátedra donde todos tienen acceso a su bajada.

Los materiales didácticos disponibles para el desarrollo de las distintas actividades son:

- Aula con proyector para el dictado de clases teóricas-prácticas
- Gabinete de computación con proyector para realizar ejercitación sobre distintas tecnologías de diseño de sistemas
- Biblioteca con libros actualizados

DIRECCIÓN ACADÉMICA
ES COPIA FIEL DEL ORIGINAL



MARIA EUGENIA LAVORATTO
DIRECTORA
DIRECCIÓN ACADÉMICA
U.T.N. F.R.L.P.



- Sitio web y Foro oficial de la cátedra.

EVALUACIÓN

La evaluación de la cursada consta de dos parciales prácticos semestrales con sus respectivos recuperatorios. Es condición necesaria para poder rendir estos exámenes la aprobación de todas las actividades prácticas establecidas por la cátedra en el correspondiente semestre. En cada evaluación el profesor plantea los objetivos a alcanzar: conocimientos, habilidades y actitudes que el alumno deberá elaborar y expresar.

El examen final consta de dos fases: un Múltiple-Choice el cual es corregido in situ y cuya aprobación permite continuar con la segunda instancia a través de una prueba oral donde el alumno expone un tema de su elección entre las bolillas correspondientes al diseño programático de la materia. Los examinadores pueden preguntar sobre el tema desarrollado en virtud de puntos deficientes o demás bolillas del programa. Tanto el examen Múltiple Choice como la evaluación oral quedan documentados conjuntamente.



MARIA EUGENIA LAHORATTO
DIRECTORA
DIRECCIÓN ACADÉMICA
U.T.N. F.R.L.P.