

Departamento de Ingeniería de Sistemas Informáticos y Telemáticos
Área de Lenguajes y Sistemas Informáticos

Asignatura: Análisis y Diseño de Sistemas.
Curso 2011-12

CARÁCTER: Materia troncal para I.T.I. Gestión,
materia obligatoria para I.T.I. Sistema

CRÉDITOS: 6 (4,5 teóricos + 1,5 prácticos).

HORARIO: Teoría: Lunes 12:30 I.T.I. Gestión
Miércoles 8:30 a 10:30 I.T.I. Sistemas, Miércoles 10:30 I.T.I. Gestión
Jueves 11:30 I.T.I. Sistemas, Jueves 12:30 I.T.I. Gestión
Prácticas: Lunes de 16 a 19 horas (pendiente de confirmar).

PROFESOR: Miguel Ángel Pérez Toledano

TUTORIAS: Lunes de 10:30 a 12:30 horas (Escuela Politécnica),
Martes de 9 a 11 horas (Facultad de Ciencias del Deporte) y
Jueves de 9:30 a 11:30 horas (Escuela Politécnica)

E-MAIL: toledano@unex.es

OBJETIVO GENERAL:

Se consideran objetivos generales a todos aquellos referidos a la Universidad, titulación o especialidad, que permiten fijar el marco general para la elaboración de los Planes de Estudio. El objetivo general que se plantea para la asignatura de *Análisis y Diseño de Sistemas* dentro del contexto de los Planes de Estudios de las Ingenierías Técnicas en Informática de Gestión y de Sistemas, se podría resumir como:

“Conocimiento y aplicación, por parte del alumno, de los principios del análisis y diseño de sistemas software, que les permitan adquirir los hábitos y conocimientos necesarios para su integración en equipos de trabajo que cooperen en la creación de grandes sistemas software”

Este objetivo primordial, puede matizarse resaltando aspectos concretos a cubrir, tales como:

- Conocimiento del uso de un ciclo de vida como sistema de organización de proyectos software.
- Introducción al desarrollo de grandes sistemas, y la necesidad del trabajo en equipo.
- Enseñar los conceptos básicos y generales que permitirán al estudiante comprender cómo aplicar una metodología de desarrollo de software.
- Familiarizar al estudiante con los entornos y herramientas de uso más general, que puedan serle de utilidad en su desarrollo profesional.
- Conocimientos que permitan al estudiante discriminar sobre el paradigma de desarrollo y la metodología más adecuada a cada situación.
- Servir de base al resto de asignaturas troncales y optativas del Plan de Estudios, que desarrollarán conceptos relacionados con los propuestos en la asignatura.

Estos aspectos servirán de referencia para la planificación de la acción docente. Sabiendo qué se pretende, hay que establecer dónde, cómo y con qué contenidos se va a llevar a cabo. Por ello se han de tomar

decisiones sobre los principios y procedimientos a seguir, dar coherencia al material, organizar el tiempo disponible, diseñar las actividades y valorar los factores que pueden condicionar en esta tarea; en definitiva, programar la asignatura en el contexto en que se va a impartir.

OBJETIVOS ESPECÍFICOS:

Los objetivos específicos son los contemplados para la asignatura, y permiten fijar el nivel de aprendizaje de la asignatura completa. Como objetivos específicos de la asignatura de *Análisis y Diseño de Sistemas* se proponen los siguientes:

1. Introducir la terminología y los conceptos básicos relacionados con la Ingeniería de Software.
2. Conocimiento del concepto de sistema de información.
3. Aplicación del concepto de ciclo de vida, así como las fases en la que se descompone.
4. Comprensión por parte del alumno de la necesidad de utilizar metodologías de trabajo como forma de desarrollo de software.
5. Comprensión por parte del alumno de la posibilidad de utilizar diferentes metodologías para el desarrollo de software, así como las diferencias conceptuales entre ellas.
6. Aprendizaje del ciclo de vida estructurado orientado a procesos así como las diferencias con el ciclo de vida estructurado orientado a los datos.
7. Aprender las propiedades que debe cumplir un buen diseño software.
8. Comparar y contrastar el análisis orientado a objetos con el análisis y diseño estructurado.
9. Aprender a evaluar un diseño software desde la perspectiva de la reutilización.
10. Conocimiento del desarrollo de sistemas mediante la utilización de la metodología orientada a objetos UML.
11. Realización de casos prácticos donde se pongan a prueba los conocimientos adquiridos.
12. Introducir al alumno en los problemas que plantea el mantenimiento de sistemas, centrándose en el conocimiento de los diferentes mantenimientos, y en el estudio de las auditorías a los sistemas.

Estos son los objetivos generales básicos que se intentarán enseñar en el desarrollo del temario de la asignatura; a ellos hay que sumar todos aquellos pequeños objetivos específicos de cada punto a tratar, así como de otros objetivos transversales que sean necesarios para alcanzar las metas propuestas.

NORMAS GENERALES:

- Todos alumnos deberán entregar una ficha al profesor de la asignatura.
- Durante la primera semana de clase cada alumno elegirá entre sus compañeros a dos para formar grupos de tres para realizar las prácticas. La lista provisional de grupos de prácticas se publicará y basándose en ella se asignarán las sesiones de práctica.

CRITERIOS DE EVALUACIÓN:

- En cada grupo de prácticas, que coincidirá con la hora en la que asiste el equipo a clase, se entregará un enunciado concreto con las características básicas de la práctica que deben desarrollar. En el transcurso de las clases se irán matizando las consideraciones que se estimen oportunas para el desarrollo del análisis.
- Con suficiente antelación, se fijará un día de entrega de la documentación de la práctica desarrollada.
- Con suficiente antelación, a los equipos que entreguen la documentación y no hayan tenido un seguimiento continuo en las horas de práctica, se fijará un día de defensa de la documentación

entregada. Para aprobar la defensa, deben presentarse los componentes del equipo requeridos por el profesor.

- Las personas que suspendan las prácticas en la convocatoria de Febrero deberán presentar una práctica individual en las convocatorias de Junio o Septiembre. El enunciado y contenido de la misma se podrá obtener en reprografías, en la página web de la asignatura, en el campus virtual, o en el despacho del profesor después de la publicación definitiva de notas de la convocatoria de Febrero.
- Si por alguna causa algún compañero se desentiende del cometido de la práctica, deberá informar inmediatamente al profesor.
- Cualquier sospecha de copia de la documentación entregada, implicará inexorablemente suspender todas las entregas similares efectuadas.
- Las prácticas serán evaluadas con un porcentaje del 30% de la nota total de la asignatura.
- Para poder hacer la nota media entre las notas de teoría y práctica debe alcanzarse como mínimo 5 puntos en cada una de las partes.
- Tanto la nota del examen de teoría como la nota de la práctica podrá ser guardada hasta la convocatoria de Septiembre de ese curso académico, siempre que sea superior a 5.
- El examen tiene un porcentaje del 70% del total de la nota, y contendrá preguntas sobre la parte de teoría, desarrollo de casos prácticos y/o problemas.
- Con carácter excepcional, cuando la asignatura deje de impartirse por extinción de los títulos de Ingeniería Técnica en Informática de Gestión y de Sistemas, y dado que no es posible la realización de las prácticas, la evaluación se realizará únicamente con el examen de la asignatura.

PRÁCTICAS

- Los casos prácticos se asignaran a cada grupo en clase de prácticas y opcionalmente se aceptaran trabajos propuestos por los alumnos, o por el profesor. En dichos casos, se realizarán todas las fases de una metodología estructurada en el desarrollo de un sistema informático hasta llegar a la programación.
- Utilización de una metodología Orientada a Objetos (UML) para describir los casos prácticos resueltos usando metodología estructurada. El modelo obtenido debe contener la descripción de los requisitos, el diagrama de clases de todo el sistema (el cual puede ser precisado usando diagramas de paquetes y/o componentes), una descripción precisa del funcionamiento de los requisitos (usando diagramas de secuencias), así como todos los diagramas accesorios que el alumno necesite para documentar la solución.
- Para aprobar las prácticas es necesario tener aprobadas las dos partes, es decir, aprobado el modelado estructurado y el orientado a objetos.

HORAS DE ESTUDIO RECOMENDADAS

Se recomienda seguir la asignatura al día para poder avanzar de manera coordinada los contenidos teóricos de la asignatura con los de la práctica. Debido a la heterogeneidad del alumnado no existe un número de fijo de horas de trabajo, no obstante la dedicación de unas tres horas o cuatro horas de trabajo semanales debe ser suficiente.

RECOMENDACIONES

Para un correcto seguimiento de la asignatura se recomienda la asistencia a todas las clases prácticas, en las cuales se realizará un estudio individualizado de las evoluciones de cada uno de los trabajos. Se controlará la asistencia a las prácticas.

PROGRAMA DE LA ASIGNATURA:

CONTENIDO :

Tema 1.- Introducción. Conceptos Generales.

- 1.1.- Introducción.
- 1.2.- Concepto de sistema y subsistema.
- 1.3.- Conceptos de mensaje, dato, información y conocimiento.
- 1.4.- Sistema de información. Sistema informático. Elementos.
- 1.5.- Papel del analista.
- 1.6.- Necesidad del análisis.
- 1.7.- Problemas generales del análisis.

Tema 2.- Ciclo de Vida de un Sistema.

- 2.1.- Concepto de Ciclo de Vida: Proceso de Ingeniería. Ciclo Básico.
- 2.2.- Ciclo de Vida Clásico
 - 2.2.1.- Etapas: modelo, objetivo, salidas.
 - 2.2.2.- Problemas de la aproximación clásica.
- 2.3.- Aproximación Estructurada.
 - 2.3.1.- Introducción.
 - 2.3.2.- Etapas: objetivos, productos, hitos, revisiones.
- 2.4.- Otros Modelos de Desarrollo de Software.
 - 2.4.1.- Modelo de desarrollo lineal secuencial.
 - 2.4.2.- Modelo de desarrollo de construcción de prototipos.
 - 2.4.3.- Modelo de desarrollo rápido de aplicaciones.
 - 2.4.4.- Modelo de desarrollo incremental.
 - 2.4.5.- Modelo de desarrollo en espiral.
 - 2.4.6.- Modelo de ensamblaje de componentes.
 - 2.4.7.- Otros modelos de desarrollo de software.

Tema 3.- Metodologías de desarrollo de software.

- 3.1.- Introducción.
- 3.2.- Características principales.
- 3.3.- Clasificación de las metodologías.
- 3.4.- Principales metodologías de desarrollo.

Tema 4.- Análisis estructurado de sistemas orientado a procesos.

- 4.1.- Introducción a la fase de Análisis del sistema.
- 4.2.- Fases previas en el ciclo de vida estructurado orientado a procesos.
 - 4.2.1.- Fase de Definición del Problema.
 - 4.2.2.- Fase de Estudio de Viabilidad.
- 4.3.- Sintaxis de los Diagramas de Flujos de Datos (DFD's).
- 4.4.- Análisis del sistema usando DFD's.
 - 4.4.1.- Propiedades de los DFD's.
 - 4.4.2.- Técnicas de modelados de DFD's.
 - 4.4.3.- Ejemplos.
- 4.5.- Salidas de la fase de Análisis del sistema.

Tema 5.- Fases de Diseño y Diseño Detallado Estructurado de Sistemas Orientado a Procesos.

- 5.1.- Introducción a la fase de Diseño del sistema.
- 5.2.- Diseño de las salidas.
- 5.3.- Diseño de las entradas.
- 5.4.- Interfaces de usuario.
- 5.5.- Salidas de la fase de Diseño del sistema.
- 5.6.- Introducción a la fase de Diseño Detallado del sistema.
- 5.7.- Mapas de estructuras.
 - 5.7.1.- Propiedades de los mapas de estructuras.
- 5.8.- Transformación de DFD's en mapas de estructuras.
- 5.9.- Especificación de programas usando tarjetas I.P.O.
- 5.10.- Ejemplos de resolución de problemas.
- 5.11.- Salidas de la fase de Diseño Detallado del sistema.

Tema 6.- Implementación, Implantación y mantenimiento.

- 6.1.- Tareas a realizar en la fase de implementación de sistemas software.
- 6.2.- La implantación del sistema construido.
- 6.3.- Mantenimiento de sistemas.

Tema 7. Introducción al Desarrollo de Sistemas Orientados a Objetos.

- 7.1.- Introducción de conceptos generales.
- 7.2.- Evolución histórica y fundamentos del desarrollo de sistemas Orientado a Objetos.
- 7.3.- Ventajas del desarrollo Orientado a Objetos.
- 7.4.- Metodologías de desarrollo Orientado a Objetos y UML.
- 7.5.- Proceso de modelado de sistemas software usando UML.
 - 7.5.1.- Proceso Unificado. Fases y Workflows.
- 7.6.- Orientación a Objetos y reutilización.

Tema 8.- Modelado de sistemas utilizando UML.

- 8.1.- Introducción al modelado de sistemas usando UML.
- 8.2.- Descripción de requisitos usando diagramas de casos de uso.
- 8.3.- Modelado del sistema usando diagramas de clases.
- 8.4.- Utilización de paquetes.
- 8.5.- Modelado de interacciones usando diagramas de secuencia.
- 8.6.- Modelado de interacciones usando diagramas de colaboración.
- 8.7.- Modelado del comportamiento de los objetos usando diagramas de estados.
- 8.8.- Modelado del comportamiento del sistema usando diagramas de actividad.
- 8.9.- Diagramas de implementación usando componentes.
- 8.10.- Modelado de la implementación usando diagramas de despliegue.

BIBLIOGRAFIA

- Durán Toro A., Bernardez Jiménez B.. “*Metodología para la Elicitación de Requisitos de Sistemas Software, versión 2.1*”. Universidad de Sevilla, Informe Técnico LSI-2000-10. <http://www.lsi.us.es/~amador>.
- Arias J., Díaz M., Navasa A., “*Ciclo de vida estructurado orientado a procesos*”, Análisis I.P., S.L. 1993. ISBN: 84-932170-1-8.
- Bennet S, McRobb S., Farmer R. “*Análisis y Diseño orientado a objetos de sistemas*”. Editorial Mc Graw Hill, 2006. ISBN: 978-84-481-5640-4.
- Bruegge and Dutooit. “*Object-Oriented Software Engineering: Using UML, Patterns and Java, 3/E*”.

Editorial Prentice Hall, 2009. ISBN: 97-801-3606-1250.

- Gane, C. y Sarson, T., “*Análisis estructurado de sistemas*”, El Ateneo, 1988 . ISBN: 950-02-5261-2.
- Hawryszkiewicz, I.T., “*Introducción al análisis y diseño de sistemas con ejemplos prácticos*”, Anaya 1990. ISBN: 84-7614-259-5.
- Jacobson, Booch, Rumbaugh. “*El lenguaje unificado de modelado*”. Segunda edición. Editorial Addison Wesley, 2006. ISBN: 84-78290-76-1.
- Jacobson, Booch, Rumbaugh. “*El proceso unificado de desarrollo de software*”. Editorial Pearson Education, 2000. ISBN: 978-84-782-9036-9.
- Jacobson, Griss, Jonson. “*Software Reuse. Architecture, Process and Organization for Business Success*”. Editorial Addison Wesley, 1997. ISBN: 0-201-92476-5.
- John R. Cameron. “*Tutorial JSP&JSD: The Jackson approach to software development*”. Editorial IEEE Computer Society ,1983. ISBN: 0-8186-8516-6.
- Larman G. “*UML y patrones. Introducción al análisis y diseño orientado a objetos*”. Editorial Pearson Alhambra. 2003. ISBN: .978-84-2053-4381
- Priestley M. “*Practical Object-Oriented Design with UML*”. Segunda edición. Editorial Mc Graw Hill, 2003. ISBN: 0-07-709599-5.
- Martin y Kendall. “*UML Distilled: applying the standard Object Modeling Language*”. Segunda edición. Editorial Addison Wesley, 2000. ISBN: 0-201-65783-X.
- Matheron P. “*Merise. Metodología de desarrollo de sistemas*”. Editorial Paraninfo, 1990. ISBN: 84-283-1768-2.
- Merle P. Martin. “*Analysis and Design of Business Information Systems*”. Editorial Mc Millan, 1991. ISBN: 0-675-20852-1.
- MÉTRICA.VERSIÓN 3. “*Metodología de Planificación, Desarrollo y Mantenimiento de sistemas de información*”. <http://www.csae.map.es/csi/metrica3/>
- Meyer B.. “*Construcción de Software Orientado a Objetos*”. 2ª edición. Editorial Prentice, 1999. ISBN: 84-8322-040-7.
- Navasa Martínez Amparo, Pérez Toledano Miguel Ángel, Sánchez Alonso Marisol. “*Aplicación de Uml al Desarrollo de Sistemas Orientados a Objetos*”. ISBN: 84-605-9632-x.
- Perdita S.y Pooley R. “*Utilización de UML en Ingeniería de Software con Objetos y Componentes*”. Addison Wesley, 2007. ISBN: 97-884-78290-86-4.
- Peter Coad and Edward Yourdon. “*Object-Oriented Analysis*”. 2th Edición. Editorial Prentice Hall, 1991. ISBN: 0-13-629981-4.
- Peter Coad and Edward Yourdon. “*Object-Oriented Design*”. 2th Edición. Editorial Prentice Hall, 1991. ISBN: 0-13-630070-7.
- Piattini M, Del Peso E, “*Auditoria Informática: Un enfoque práctico*”. 2ª edición. Editorial Ra-Ma, 2000. ISBN: 84-7897444-x.
- Piattini M, et all. “*Mantenimiento del Software: Modelos, técnicas y métodos para la gestión del cambio*”. Editorial Ra-Ma, 2000. ISBN: 84-7897448-2.
- Piattini Mario, Calvo-Manzano José, Cervera Joaquín, Fernández Luís. “*Análisis y diseño detallado de Aplicaciones Informáticas de Gestión*”. Ed. Ra-ma, 1996. ISBN:84-7897232-1.
- Pressman, R.S., “*Ingeniería del Software. Un enfoque práctico*” Séptima Edición, McGraw-Hill, 2010. ISBN:84-481-3214-9.
- Rumbaugh J., Blaha M., Premerlani, W., Eddy F., Lorensen.W. “*Modelado y diseño orientado a objetos. Metodología OMT*”. Editorial Prentice Hall 1994. ISBN: 0-13-240698-5.
- Schach S.R. “*Análisis y Diseño Orientado a Objetos con UML y el Proceso Unificado*”. Editorial Mc Graw Hill, 2005. ISBN: 97-010-4982-9
- Sommerville I. “*Ingeniería del Software*”. Séptima edición. Editorial Addison Wesley, 2005. ISBN. 84-782-90745.

- Tayi, Ballou. “*Examining data quality*”. ACM, 41(2), Febrero 1998.
- Texel y Williams. “*Use Cases Combined with Booch, Omt, Uml. Process and products*”. Prentice Hall, 1997.
- Warner J., Kleppe A.. “*The Object Constraint Language. Precise Modeling with UML*”. Editorial Addison-Wesley 1998. ISBN: 0-201-37940-6.