



ASIGNATURA: Gestión y Explotación de Sistemas Informáticos.

Curso: 2003/2004

Carácter: Troncal **Temporalidad:** 2º Cuatrimestre **Créditos:** 6 (4,5T+1,5P)
Profesor: Andrés Santiago Martín **Despacho:** Despacho 10
Web: <http://cum.unex.es/profesores/asanmar/default.htm> **E-mail:** asanmar@unex.es

NORMAS GENERALES:

- Las convocatorias de los exámenes serán fijadas por la Subdirección Académica del Centro.
- Todo alumno deberá entregar obligatoriamente una ficha al profesor de la asignatura.
- La prácticas se realizarán en grupos de dos personas.
- La composición de los grupos de prácticas deben de entregarse al profesor no más tarde del día 15 de Octubre de 2003.
- En los trabajos prácticos de la asignatura el profesor establecerá, en su momento, la fecha límite de entrega de cada uno de dichos trabajos.

CRITERIOS DE EVALUACIÓN:

- **Parte de Teoría:**
 - Habrá un examen que constará de dos partes: una parte de teoría y otra parte de problemas. La nota de este examen representará el 65% de la nota final.
 - Para poder superar el examen, en cada parte del mismo (teoría y problemas) deberá obtener el alumno, al menos, el 30 % de la nota máxima que puede obtenerse en cada parte.
- **Parte Práctica:**
 - Su entrega será indispensable para aprobar la parte práctica. Si no se entrega la asignatura quedará suspensa aunque la teoría se encuentre aprobada.
 - Para aprobar las prácticas estas deben tener una nota mayor o igual a 5.
- **Trabajos voluntarios presentados.**
 - Dichos trabajos versarán sobre temas relacionados directamente con la asignatura, y necesitarán de la aprobación previa del profesor.
 - Estos trabajos, tras su evaluación por el profesor, podrán subir de 0,5 a 1,5 puntos que se sumarán a la nota conseguida en los apartados anteriores, siempre que en los mismos se obtenga la calificación mínima exigida.
 - La entrega de trabajos sólo se aplicará a la convocatoria de Febrero.
- La nota final será igual a:
 - **Si** (teoría $\geq 4,5$) y (práctica $\geq 4,5$) y $[(\text{teoría} * 0,65) + (\text{práctica} * 0,35)] \geq 5$
 - nota final = (teoría * 0,65) + (práctica * 0,35) + nota trabajos
 - **Si no**
 - nota final = Suspenso
 - **Fin si**
- En todo caso, sólo se aprobará si la nota final es igual o superior a 5.
- Si un alumno tiene una parte aprobada, nota mayor o igual a 5, y tiene la otra parte suspensa (no compensable), con nota menor que 4,5, la nota final obtenida será la de la parte suspensa.
- Tanto la nota del examen teórico como la nota de la práctica podrán ser guardadas hasta la convocatoria de septiembre del presente curso, siempre que sea superior a 5 puntos.
- Si un alumno presenta un trabajo voluntario en la convocatoria de Junio, y en cambio no aprueba la asignatura, la nota de dicho trabajo se guarda hasta la convocatoria de septiembre del presente curso.



OBJETIVO GENERAL:

- Obtener los conocimientos necesarios en el desarrollo de sistemas informáticos, a fin de conseguir que el alumno sea consciente de la importancia que tiene la planificación previa al proceso de desarrollo de un proyecto software para que éste no fracase, se obtenga un producto de calidad, se realice dentro del presupuesto estimado y en el plazo de entrega previsto.

OBJETIVOS ESPECIFICOS:

- Descripción de los problemas, principios, métodos y tecnologías asociadas con la Ingeniería del Software.
- Conocer el concepto de ciclo de vida de un sistema, los diferentes modelos de desarrollo actuales de software y como afectan éstos a la planificación, gestión y explotación de una aplicación informática.
- Adquirir conciencia de la necesidad de una planificación del proceso de desarrollo de sistemas, centrándose en los aspectos de estimación y análisis y gestión de riesgos.
- Conocer el concepto de Herramienta CASE, así como los distintos tipos de herramientas que existen y que grado de tareas de desarrollo cubren.
- Saber desarrollar un Plan de Desarrollo del Software, aplicando los conceptos de calidad tanto en el software construido, como en el proceso de desarrollo, determinando las distintas tareas para llevar a cabo para su consecución.
- Presentación de la importancia de los requisitos en el ciclo de vida del Software.

METODOLOGÍA:

- En las tres clases de teoría semanales se expondrán los conceptos teóricos básicos sobre gestión y desarrollo de proyectos informáticos, y se introducen en mayor detalle técnicas y metodologías estructuradas de desarrollo del software.
- Cuando en el tema que se esté impartiendo en teoría haya problemas a resolver, de las tres horas de teoría semanales, una hora semanal se dedicará a la resolución de problemas. Para ello, se facilitará a los alumnos una lista de problemas sobre dicho tema, y algunos de los problemas de dicha lista se resolverán en dicha hora de problemas.
- Así mismo, están las horas de tutorías en las que los alumnos pueden consultar con el profesor la resolución de cualquier duda planteada sobre cualquier aspecto de la asignatura.
- En las clases teóricas se utilizan medios audiovisuales de divulgación informática.
- Las clases prácticas se destinan a realizar diversos ejemplos prácticos a fin de conocer y manejar diversos programas de estimación y planificación de proyectos software.
- En la página Web del profesor y en la copistería del centro se encuentra a disposición de los alumnos, los horarios de tutorías, temarios de las asignaturas, criterios de evaluación así como todo el material didáctico utilizado en la asignatura.



PROGRAMA TEÓRICO:

TEMA 1.- GESTIÓN DE PROYECTOS SOFTWARE.

Introducción.
El desarrollo de software.
La Ingeniería del Software.
Tareas de gestión de la Ingeniería del Software.
Tareas técnicas de la Ingeniería del Software.
Planificación del proyecto software.

TEMA 2.- MÉTRICAS DEL SOFTWARE.

Introducción.
Conceptos básicos.
Medidas y modelos.
Alcance de las métricas del software.
Clasificación de las métricas del software.
Recogida de datos métricos.
Medición de atributos internos del producto.
Medición de atributos externos del producto.
Medición de recursos.

TEMA 3.- ESTIMACIÓN DEL SOFTWARE.

Introducción.
Problemática del proceso de estimación del software.
Requisitos que debe cumplir un buen estimador.
Marco temporal de la estimación de proyectos.
Salidas del proceso de estimación.
Técnicas de estimación.
Método de estimación de puntos de función.
Modelos de coste y esfuerzo: modelo COCOMO.
Modelo COCOMO II.

TEMA 4.- GESTIÓN DE RIESGOS.

Introducción.
Definición y clasificación de riesgos.
Actividades de análisis y gestión de riesgos.
Métodos de análisis y gestión del riesgo.
Método de Boehm.
Metodología Magerit.
Taxonomy Based Risk Management.

TEMA 5.- PLANIFICACIÓN.

Introducción.
¿Qué es la planificación temporal?
Conceptos básicos sobre grafos.
Técnicas de planificación.
Técnica PERT.
Método de ROY.

TEMA 6.- HERRAMIENTAS CASE.

Introducción.
Conceptos básicos.
Entornos case integrados.
Perspectivas futuras.
Adopción de la tecnología CASE.
Evaluación de CASE.

TEMA 7.- MANTENIMIENTO.

Conceptos generales.
Dificultades del mantenimiento.
Soluciones al problema del mantenimiento.
El proceso de mantenimiento.
Ingeniería Inversa y Reingeniería.

TEMA 8.- GESTIÓN DE CALIDAD.

Calidad del software y Garantía de Calidad del Software (SQA).
Revisión del software.
Métricas de calidad.
Enfoque para la SQA.

PROGRAMA PRÁCTICO:

La práctica consistirá en una serie de trabajos prácticos relacionados con el contenido teórico de la asignatura. Para el desarrollo de dichos trabajos se utilizará Microsoft Project 2002, con licencia del programa MSNDAA del Departamento de Informática, y una serie de programas de libre distribución.

Microsoft Project 2002 se puede obtener a través del Departamento de Informática.

Los programas de software de libre distribución para la realización de las prácticas son:

- Seat.
- Cosmos22.
- Cosmos41.
- USC Cocomo 87.
- USC Cocomo II.

Enunciados de la prácticas.

- Estimación básica: Puntos de Función y Macromodelo COCOMO.
- Estimación Intermedia: Micromodelo COCOMO Intermedio.
- Estimación de Costes del Software: COCOMO II.
- Definición de tareas y dependencias entre tareas con MS-Project.
- Gestión de Recursos y Proyectos con MS-Project.

BIBLIOGRAFÍA:

- **Referencias principales.**

1. *Boehm, B. W.*
"Software Engineering Economics",
Prentice Hall, 1981.
2. *Colmenar Santos, Antonio y otros,*
"Gestión de proyectos con Microsoft Project 2002",
RaMa, 2002.
3. *Piattini, M. G., Calvo-Manzano, J. A., Cervera, J., Fernández, L.*
Análisis y Diseño Detallado de Aplicaciones Informáticas de Gestión. 5ª Edición
Ra-Ma, 1996.
4. *Pressman, R. S.*
"Ingeniería del Software: Un Enfoque Práctico. 5ª Edición",
McGraw-Hill. 2002.
5. *Sommerville, Y.*
"Ingeniería del Software. 6ª Edición",
Addison-Wesley. 2002.

- **Otras referencias.**

1. *Brooks, F.*
"The mythical Man-Month: essays on Software Engineering",
Addison Wesley, 1982.
2. *Kan, Stephen H.*
"Metrics and Models in Software Quality Engineering",
Addison-Wesley Pub Co. 1995.
3. *Kan, Stephen H.,*
"Metrics and Models in Software Quality Engineering",
Addison-Wesley Pub Co. 1995.
4. **Metodología de planificación y desarrollo de sistemas de información. Métrica Versión 3,**
Ministerio para las Administraciones Públicas. 2001.
Enlace: <http://www.csi.map.es/csi/metrica3/index.html>
5. *Pressman, Roger S.,*
"Ingeniería del Software. Un enfoque práctico", 4ª edición,
McGraw-Hill, 1998.
6. *Pressman, Roger S.,*
"Ingeniería del Software. Un enfoque práctico", 3ª edición,
McGraw-Hill, 1993.

Además en cada tema se especificará las referencias bibliográficas que se hayan utilizado.