

**GOBIERNO CONSTITUCIONAL DEL ESTADO LIBRE Y SOBERANO DE OAXACA  
INSTITUTO ESTATAL DE EDUCACIÓN PÚBLICA DE OAXACA  
COORDINACIÓN GENERAL DE PLANEACIÓN EDUCATIVA  
COORDINACIÓN GENERAL DE EDUCACIÓN MEDIA SUPERIOR Y SUPERIOR**

**PROGRAMA DE ESTUDIO**

<b>NOMBRE DE LA ASIGNATURA</b>	<b>Ingeniería de Software</b>
--------------------------------	-------------------------------

<b>CICLO</b> <b>Cuarto Semestre</b>	<b>CLAVE DE LA ASIGNATURA</b> <b>2071</b>	<b>TOTAL DE HORAS</b> <b>85</b>
--	--	------------------------------------

**OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA**

Proporcionar al estudiante el conocimiento necesario para analizar, diseñar e implementar un sistema de computación nuevo o mejorar uno existente; aplicar técnicas de estimación de proyectos; identificar los factores que determinan el rendimiento de un sistema y asignar recursos para la resolución de problemas de computación a gran escala.

**TEMAS Y SUBTEMAS**

- 1. Introducción a la Ingeniería de Software**
  - 1.1. Importancia del Software.
  - 1.2. Evolución y características del software
  - 1.3. Tipos de software.
  - 1.4. La crisis del software.
  - 1.5. Definición de Ingeniería de Software.
  - 1.6. Paradigmas de ciclos de vida de la Ingeniería de software.
  - 1.7. Herramientas CASE
- 2. Administración de proyectos de software**
  - 2.1. Gráficas PERT, GANTT.
  - 2.2. Métricas del proyecto.
  - 2.3. Mediciones del software.
  - 2.4. Métricas orientadas al tamaño (LDC).
  - 2.5. Modelo de estimación de costos COCOMO.
  - 2.6. Métricas orientadas a los puntos de función.
  - 2.7. Análisis de riesgo.
- 3. Análisis de requerimientos**
  - 3.1. Técnicas de recolección de información.
  - 3.2. Identificación de requerimientos
  - 3.3. Análisis de requisitos basados en el estándar 830-1993 IEEE.
  - 3.4. Introducción y aplicación a los métodos estructurados.
  - 3.5. Introducción del método orientado a objetos en el análisis.
  - 3.6. Validación de requerimientos.
- 4. Diseño e implantación de Software**
  - 4.1. Niveles de diseño.
  - 4.2. Aplicación del diseño estructurado.
  - 4.3. Introducción al diseño orientado a objetos.
- 5. Métodos de prueba del software**
  - 5.1. El proceso de prueba
  - 5.2. Métodos de prueba
  - 5.3. Diseño de pruebas
  - 5.4. Caso de estudio: estándar de prueba



COORDINACIÓN  
GENERAL DE EDUCACIÓN  
MEDIA SUPERIOR Y SUPERIOR

**6. Administración de calidad del software**

- 6.1. El concepto de calidad del software
- 6.2. Estándares de calidad existentes
- 6.3. Métricas de calidad del software

**7. Mantenimiento de software**

- 7.1. Aspectos generales
- 7.2. Características
- 7.3. Categorías de mantenimiento: correctivo, adaptativo, preventivo, perfectivo
- 7.4. Reingeniería

**ACTIVIDADES DE APRENDIZAJE**

Revisión bibliográfica del tema por los alumnos en libros.

Dar seguimiento en cada tema de su proyecto en equipo elegido al inicio del curso, aplicando el ciclo de vida más adecuado. Para ello puede hacer uso de las siguientes herramientas:

- Software para la planificación temporal de proyectos (Microsoft Project, etc)
- Software para métricas y estimación de proyectos (BYL, COSMOS, etc.)
- Software para el análisis de requerimientos (Requisite pro, etc)
- Software para el diseño (UML de Rational rose, etc.)
- Software para la implementación del proyecto (Algún entorno de desarrollo como: Eclipse, Cbuilder, Delphi, etc)
- Software para pruebas del proyecto (Eclipse, etc)

**CRITERIOS Y PROCEDIMIENTOS DE EVALUACIÓN Y ACREDITACIÓN**

Examen departamental, evaluación parcial mediante examen y avances de su proyecto.

El proyecto idealmente se desarrollará en coordinación con la materia de Programación Orientada a Objetos II.

La suma de todos los criterios y procedimientos de evaluación y acreditación deberán integrar el 100% de la calificación.

**BIBLIOGRAFÍA**

Bibliografía básica:

- **Ingeniería del Software: Un enfoque práctico**, Pressman, S. Roger. 4a edición McGraw-Hill. 2001.
- **Ingeniería de software**. Somerville, Ian. 6a ed. Addison Wesley. 2002.
- **Ingeniería de Software: Una perspectiva orientada a objetos**. Braude, J. Eric. 1ª ed. Alfa Omega. 2003
- **Object-oriented Software Engineering**. Halladay, Steve 1993.

Bibliografía de consulta:

- **Introduction to Software Engineering**. Leach, Ronald J. 2000.
- **Object-oriented and Classical Software Engineering**. Schach, Stephen R. 2002.
- **Software Engineering with Reusables Components**. Sametinger, Johannes, Springer.
- **Component Software: Beyond Object-Oriented Programming**. Syperski, Clemens. 1a. ed. Addison-Wesley 1998
- **Requeriments Engineering: Processes and Techniques**. Somerville Ian. 2000.

**PERFIL PROFESIONAL DEL DOCENTE**

Ingeniero en computación o en Sistemas computacionales con Maestría en computación o Doctorado en computación.



COORDINACIÓN  
GENERAL DE EDUCACIÓN  
MEDIA SUPERIOR Y SUPERIOR

I.E.E.P.O