



**ADMINISTRACIÓN DE PROYECTOS DE SOFTWARE**

7°

06

Asignatura

Clave

Semestre

Créditos

Ingeniería Eléctrica

Ingeniería en Computación

Ingeniería en Computación

División

Departamento

Carrera en que se imparte

**Asignatura:**

Obligatoria

Optativa

**Horas:**

Teóricas

Prácticas

**Total (horas):**

Semana

16 Semanas

**Modalidad:** Curso.

**Seriación obligatoria antecedente:** Ninguna.

**Seriación obligatoria consecuente:** Ninguna.

**Objetivo(s) del curso:**

El alumno aplicará en un proyecto el proceso general mediante el cual se administran los proyectos de software.

**Temario**

NÚM.	NOMBRE	HORAS
1.	Introducción	1.0
2.	Componentes clave del desarrollo del software	1.0
3.	Revisión de la literatura relevante	1.0
4.	Fuentes de información	1.0
5.	Administración de los recursos humanos	2.0
6.	Producción del software	2.0
7.	Desarrollo del software	3.0
8.	Aseguramiento de la calidad	2.0
9.	Pruebas del sistema	3.0

**ADMINISTRACIÓN DE PROYECTOS DE SOFTWARE**

(2 / 8)



NÚM.	NOMBRE	HORAS
10.	Control	3.0
11.	Planeación	3.0
12.	Un caso de estudio	4.0
13.	Comportamiento del modelo	3.0
14.	Exactitud en la estimación del software	2.0
15.	Portabilidad de los modelos de estimación	3.0
16.	Método de la analogía de la estimación del software	2.0
17.	El síndrome del 90%	2.0
18.	La economía del aseguramiento de la calidad del software	2.0
19.	Mejoras del modelo y la ley de Brook	2.0
20.	Teoría de inventarios, de decisiones y simulación	6.0
		48.0
	Prácticas de laboratorio	0.0
	<b>Total</b>	48.0



## 1 Introducción

**Objetivo:** El alumno explicará la administración del desarrollo de software, una panorámica del enfoque de modelación y un resumen de los conceptos de la dinámica de los proyectos de software y notación usada.

**Contenido:**

- 1.1 El síntoma de la “crisis del software”
- 1.2 El reto de la administración de los proyectos del software

## 2 Componentes clave del desarrollo del software

**Objetivo:** El alumno explicará el modelo dinámico de los procesos del desarrollo de software.

**Contenido:**

- 2.1 Aspectos administrativos del desarrollo de software
- 2.2 Un modelo simple del proceso del desarrollo del software
- 2.3 Una perspectiva dinámica del sistema integral
- 2.4 Límites del modelo
- 2.5 Estructura del modelo
- 2.6 Subsistemas del modelo
- 2.7 Convenciones esquemáticas de la dinámica del sistema

## 3 Revisión de la literatura relevante

**Objetivo:** El alumno revisará la literatura tocante a la modelación y el análisis de la administración de proyectos, y de la administración del proyecto de desarrollo del software.

**Contenido:**

- 3.1 Modelación de la dinámica del sistema en la administración de proyectos
- 3.2 Revisión de la literatura en la administración de proyectos de la ingeniería de software
- 3.3 Panorámica de los modelos y del marco de trabajo
- 3.4 Planeación
- 3.5 Administración de los recursos humanos
- 3.6 Control

## 4 Fuentes de información

**Objetivo:** El alumno explicará los tres pasos requeridos para obtener la información necesaria para construir el modelo de administración de desarrollo de software.

**Contenido:**

- 4.1 Pasos en la obtención de la información
- 4.2 Paso 1: Primero hacer las entrevistas
- 4.3 Paso 2: Revisión de la literatura
- 4.4 Paso 3: Fin de entrevistas



## 5 Administración de los recursos humanos

**Objetivo:** El alumno explicará las cuatro actividades principales de la dinámica de la administración de los proyectos del software: administración de recursos humanos, producción del software, control y planeación.

**Contenido:**

- 5.1 Caracterización de la fuerza de trabajo
- 5.2 Diferencial de la productividad y sobrecarga de capacitación
- 5.3 Determinación del nivel de la fuerza de trabajo

## 6 Producción del software

**Objetivo:** El alumno explicará las cuatro actividades principales de la producción del software: Desarrollo, aseguramiento de la calidad, revisión y pruebas al sistema.

**Contenido:**

- 6.1 Los sectores de la producción del software
- 6.2 El sector de la asignación de la fuerza de trabajo

## 7 Desarrollo del software

**Objetivo:** El alumno explicará los procesos del desarrollo del software: Diseño y codificación del producto de software.

**Contenido:**

- 7.1 El sector del desarrollo del software
- 7.2 Productividad en el desarrollo del software
- 7.3 Productividad potencial
- 7.4 Productividad Real
- 7.5 El impacto de la productividad real debido a la motivación
- 7.6 El impacto de la productividad real debido a la comunicación

## 8 Aseguramiento de la calidad

**Objetivo:** El alumno explicará las dos metodologías complementarias del aseguramiento de la calidad: Primero, el diseño de un conjunto de requerimientos que sea coherente, completo, no ambiguo y no conflictivo. Segundo: La revisión y prueba del producto.

**Contenido:**

- 8.1 Sector de el aseguramiento de la calidad
- 8.2 Tasas en la generación de errores
- 8.3 Fuerza de trabajo mixta y las presiones de la calendarización
- 8.4 Detección del errores



## 9 Pruebas del sistema

**Objetivo:** El alumno explicará los modelos de dos conjuntos de procesos: el crecimiento de la población de errores no detectados y la prueba del sistema como consecuencia de la detección y corrección de estos errores.

### Contenido:

- 9.1 El sector de pruebas del sistema
- 9.2 Crecimiento de errores no detectados
- 9.3 Actividades de pruebas del sistema
- 9.4 Conclusión del análisis de la producción del software

## 10 Control

**Objetivo:** El alumno explicará los tres elementos de la función de control: medición, evaluación y comunicación.

### Contenido:

- 10.1 Subsistema de control
- 10.2 Medición del progreso en las nuevas actividades
- 10.3 Medición del progreso de la revisión y de las pruebas
- 10.4 Ajustes a la dimensión del trabajo
- 10.5 Ajustes en la asignación hombre-día

## 11 Planeación

**Objetivo:** El alumno elaborará los estimados de un proyecto para iniciarlo y luego los revisará conforme sea necesario a lo largo de la vida del proyecto.

### Contenido:

- 11.1 El subsistema de planeación
- 11.2 Ajustes al nivel de la fuerza de trabajo
- 11.3 Estabilidad de la calendarización
- 11.4 Determinación de la fecha de terminación
- 11.5 Resumen de las actividades del desarrollo del modelo

## 12 Un caso de estudio

**Objetivo:** El alumno informará de los resultados (examinar las bondades del modelo para reproducir los patrones dinámicos de un proyecto de software completo) de un caso de estudio conducido para probar el modelo.

### Contenido:

- 12.1 El proyecto de software
- 12.2 Parametrización del modelo



- 12.3 Administración de los recursos humanos
- 12.4 El ambiente del desarrollo de software
- 12.5 El ambiente de planeación
- 12.6 Estimaciones iniciales del proyecto
- 12.7 Resumen de los parámetros del modelo
- 12.8 Comportamiento actual y simulado del modelo
- 12.9 Resumen del caso de estudio

## 13 Comportamiento del modelo

**Objetivo:** El alumno experimentará con el modelo dinámico del sistema para hacer suposiciones de prueba, o cambiar las políticas administrativas.

### Contenido:

- 13.1 Experimentación controlada
- 13.2 El proyecto de software EXAMPLE
- 13.3 Especificación de la productividad potencial nominal
- 13.4 Inicialización de la calendarización y de la fuerza de trabajo
- 13.5 Progreso del proyecto
- 13.6 Distribución de la fuerza de trabajo
- 13.7 Intensidad del trabajo
- 13.8 Comentarios finales del modelo del proyecto EXAMPLE

## 14 Exactitud en la estimación del software

**Objetivo:** El alumno utilizará un modelo teórico, basado en suposiciones globales, para hacer un análisis cuantitativo del costo y calendarización del software.

### Contenido:

- 14.1 El costo del software y la estimación de la calendarización
- 14.2 La exactitud de la estimación del software
- 14.3 Los estimados diferentes crean proyectos diferentes
- 14.4 Necesidad de la experimentación simulada
- 14.5 Caso de estudio en política de factores de seguridad
- 14.6 Vuelta al aspecto de exactitud de estimación del software

## 15 Portabilidad de los modelos de estimación

**Objetivo:** El alumno mostrará la importancia de la portabilidad de los modelos de estimación (la habilidad de usarlos en diferentes entornos) del software

### Contenido:

- 15.1 Modelos de estimación cuantitativos del software
- 15.2 Portabilidad de los modelos de estimación
- 15.3 Variables del personal y de la adquisición de la fuerza de trabajo
- 15.4 Variables de la estimación del esfuerzo
- 15.5 Un experimento que combina el efecto de las cuatro variables



### 16 Método de la analogía para la estimación del software

**Objetivo:** El alumno aplicará el método de estimación por analogía, el método más utilizado.

**Contenido:**

- 16.1 Estimación por analogía
- 16.2 Experimento para evaluar el método de analogía
- 16.3 Análisis de experimentos

### 17 El síndrome del 90%

**Objetivo:** El alumno explicará el problema de control conocido como el síndrome del 90%.

**Contenido:**

- 17.1 Descripción del síndrome del 90%
- 17.2 Reproducción del modelo del 90%
- 17.3 Implicaciones del síndrome del 90%

### 18 La economía del aseguramiento de la calidad del software

**Objetivo:** El alumno explicará el conjunto de actividades conocido como *aseguramiento de la calidad* (conjunto de actividades para reducir dudas y riesgos en el rendimiento del producto en el ambiente objetivo).

**Contenido:**

- 18.1 Aseguramiento de la calidad y metodologías
- 18.2 Economía del aseguramiento de la calidad
- 18.3 Esfuerzo del aseguramiento de la calidad óptima
- 18.4 Generalidad de optimalidad

### 19 Mejoras del modelo y la ley de Brooks

**Objetivo:** El alumno explicará la ley de Brooks.

**Contenido:**

- 19.1 Mejoras al modelo
- 19.2 La ley de Brooks
- 19.3 Realces al modelo para resaltar la ley de Brooks
- 19.4 Experimentación con el modelo



### 20 Teoría de Inventarios, de decisiones y simulación

**Objetivo:** El alumno explicará la teoría de inventarios, la de decisiones y los principios de la simulación.

**Contenido:**

- 20.1 Teoría de inventarios
- 20.2 Teoría de decisiones
- 20.3 Simulación

**Bibliografía básica:**

ABDEL-Hamid, TAREK Y Madnick; STUART, E.  
*Software Project Dynamics: an integrated approach*  
Portland  
Prentice Hall, 1991

**Bibliografía complementaria:**

WELLS, Timothy D.  
*Dynamic Software Development: Mananning Projects in Flux*  
Boca Raton, Florida, USA  
Auberach Publications, 2002

**Sugerencias didácticas:**

Exposición oral

Exposición audiovisual

Ejercicios dentro de clase

Ejercicios fuera del aula

Seminarios

Lecturas obligatorias

Trabajos de investigación

Prácticas de taller o laboratorio

Prácticas de campo

Otras

**Forma de evaluar:**

Exámenes parciales

Exámenes finales

Trabajos y tareas fuera del aula

Participación en clase

Asistencias a prácticas

Otras

**Perfil profesiográfico de quienes pueden impartir la asignatura**

Profesional con experiencia en administración de proyectos de software.