

	<b>UNIVERSIDAD POPULAR DEL CESAR</b>	CODIGO: 201-300-PRO05-FOR01
		VERSIÓN: 1
PLAN DE ASIGNATURA		PÁG: 1 de 7

### IDENTIFICACIÓN

Nombre de la asignatura	<b>INGENIERIA SOFTWARE II</b>		
Código de la asignatura	SS602		
Programa Académico	Ingeniería de Sistemas		
Créditos académicos	3		
Trabajo semanal del estudiante	Docencia directa: 4	Trabajo Independiente: 5	
Trabajo semestral del estudiante			
Pre-requisitos	Ingeniería de software I		
Co-requisitos			
Departamento oferente	Ingenierías y tecnologías		
Tipo de Asignatura	Teórico:	Teórico-Práctico:X	Práctico:
Naturaleza de la Asignatura	Habilitable:		No Habilitable:X
	Validable:		No Validable:
	Homologable:		No Homologable:

### PRESENTACIÓN

La Ingeniería del Software II como asignatura dentro del Plan de Estudios de Ingeniería de Sistemas, corresponde al ciclo profesional y consta de 4 créditos. Su principal objetivo es presentar las actividades técnicas e ingenieriles que se llevan a cabo en el desarrollo de un producto software. Además, describe los métodos y tecnologías asociadas con la Ingeniería del Software, en concreto con el desarrollo de proyectos de software y específicamente, con las partes de gestión, control de la calidad, y evolución del software. Por otra parte, se debe afianzar al alumno en la sistematización de la creación del software mediante el uso de métodos de ingeniería del software, y en la de proyectos software; partiendo de la base adquirida en la asignatura de Ingeniería del Software I.

### JUSTIFICACIÓN

Para un Ingeniero de sistemas es imprescindible tener conocimientos de cómo desarrollar un software con calidad, teniendo en cuenta las fases de su ciclo de vida, desde que surge la necesidad de realizar dicho sistema hasta su implementación. En el desarrollo de la asignatura de ingeniería de software II, el estudiante obtendrá las herramientas para participar en procesos de desarrollo de software, teniendo en cuenta criterios de calidad.

El desarrollo de software requiere de un proceso sistemático, cumplir con principios técnicos y de ética profesional, además de la entrega de productos de calidad y procesos productivos. La asignatura de Ingeniería de Software II, tiene como principal propósito consolidar la formación de un desarrollador de software basado en tecnología, principios y estándares de

	<b>UNIVERSIDAD POPULAR DEL CESAR</b>	CODIGO: 201-300-PRO05-FOR01
		VERSIÓN: 1
PLAN DE ASIGNATURA		PÁG: 2 de 7

desarrollo teniendo como desafío principal la calidad en el software. Además, los tópicos que en esta asignatura se desarrollan son fundamentales para la formación de cualquier ingeniero de sistemas, sea cual sea su perfil académico y su futuro perfil profesional.

### OBJETIVO GENERAL

Preparar al estudiante para participar en proyectos de desarrollo de software, utilizando criterios de calidad tanto del producto como del proceso, aplicando técnicas y herramientas de gestión, control y aseguramiento de la calidad.

### OBJETIVOS ESPECÍFICOS

- Estudiar diferentes métodos de pruebas de software para comprender la importancia de aplicar pruebas sistemáticas en las etapas de desarrollo de software.
- Adquirir conocimiento sobre gestión de proyectos de software, como: Planificación, métricas, estimación y gestión de riesgos.
- Analizar y comprender los mecanismos internacionales de certificación de la calidad en productos de software.

### COMPETENCIAS GENERALES Y ESPECÍFICAS

- Aplicar técnicas y procedimientos de pruebas del software
- Aplicar técnicas y procedimientos de gestión, control y aseguramiento de la calidad.
- Capacidad de resolución de problemas de estimación de software, aplicando conocimientos de matemáticas, ciencias e ingeniería.
- Capacidad para tomar iniciativas y espíritu emprendedor, el liderazgo, la dirección, la gestión de equipos y proyectos de tecnologías de información.
- Conocer y aplicar los principios de la ingeniería del software y de sus tecnologías para garantizar que las implementaciones de software sean robustas, fiables y apropiadas en todos los ámbitos.
- Capacidad para trabajar de forma efectiva como individuo, organizando y planificando su propio trabajo, de forma independiente o como miembro de un equipo.

### METODOLOGÍA

Durante una clase de teoría o lección magistral, el profesor realiza una exposición verbal de los contenidos sobre la materia objeto de estudio, mediante la cual suministra a los alumnos información esencial y organizada procedente de diversas fuentes con unos objetivos específicos predefinidos.

Se trabajará con el sistema de clases participativas donde los estudiantes, a través de actividades guiadas por el profesor, discutirán los temas fundamentales del curso, con base

	<b>UNIVERSIDAD POPULAR DEL CESAR</b>	CODIGO: 201-300-PRO05-FOR01
		VERSIÓN: 1
PLAN DE ASIGNATURA		PÁG: 3 de 7

en las lecturas asignadas previamente y las exposiciones del profesor. También desarrollarán ejercicios de aplicación.

Como complemento de la clase de teoría (lección magistral) se desarrollarán ejercicios en clases y análisis de casos, para que los estudiantes desarrollen soluciones adecuadas a un determinado fin, mediante la aplicación de fórmulas, la aplicación de procedimientos de transformación de la información disponible y la interpretación de los resultados. Es decir, la participación activa en el análisis y resolución de los ejercicios o realización de ejercicios y programas, planteados en clase. La intención principal es la de aplicar lo ya aprendido para favorecer la comprensión tanto de la importancia como del contenido de un nuevo tema, afianzar conocimientos y estrategias y su aplicación en las situaciones prácticas que se planteen.

Para lograr los objetivos de aprendizaje que se proponen es necesario que el estudiante realice las siguientes actividades:

- Antes de cada clase el estudiante debe haber estudiado previamente cada tema contenido en cada unidad. Como resultado de cada actividad el estudiante deberá realizar una elaboración conceptual o una aplicación conceptual, mediante la solución de un caso o ejercicio. El resultado de esta actividad debe ser presentada en la clase.
- Durante la clase cada tema será revisado por el profesor mediante preguntas realizadas a los estudiantes, se realizarán casos y ejercicios de aplicación de los conceptos como base, para el logro de los objetivos de aprendizaje y el desarrollo de las capacidades de análisis y solución de problemas.
- Después de la clase el estudiante deberá trabajar con los casos y ejercicios complementarios propuestos por el profesor para reforzar el aprendizaje y consolidar el desarrollo de las capacidades propuestas.

### ESTRATEGIAS METODOLÓGICAS

- **Trabajo individual:**  
Lecturas asignadas por temáticas, como resultado el estudiante deberá realizar una elaboración conceptual, representada en un mapa conceptual  
Estudio de casos o una aplicación conceptual, mediante la solución de un caso o ejercicio.
- **Trabajos en grupo:**  
Se conforman grupos para realizar el proyecto de aula asignado, el proyecto se divide y se gestiona la elaboración del proyecto por partes.  
Exposiciones  
Talleres en clases

	<b>UNIVERSIDAD POPULAR DEL CESAR</b>	CODIGO: 201-300-PRO05-FOR01
		VERSIÓN: 1
	PLAN DE ASIGNATURA	PÁG: 4 de 7

**CONTENIDO**



## 1. PRUEBAS DE SOFTWARE

- 1.1. Fundamentos de las pruebas
- 1.2. El proceso de las pruebas
- 1.3. Técnicas de prueba
  - 1.3.1. Pruebas de caja negra
  - 1.3.2. Pruebas de caja Blanca
- 1.4. Nivel de pruebas
  - 1.4.1. Pruebas de Unidad
  - 1.4.2. Pruebas de Integración
  - 1.4.3. Pruebas
  - 1.4.4. Pruebas de aceptación
- 1.5. Diseño de casos de pruebas
- 1.6. Herramientas de Pruebas

de

Sistema

## 2. METRICAS DEL SOFTWARE

- 2.1. Definición y características de las métricas.
  - 2.1.1. Conceptos básicos
  - 2.1.2. Tipos de escalas de medición
  - 2.1.3. Tipos Medidas
- 2.2. Tipos de métricas
  - 2.2.1. Métricas del producto
    - Medición de atributos internos del producto
    - Medición de atributos externos del producto
  - 2.2.2. Métricas del proceso
  - 2.2.3. Métricas de proyectos
- 2.3. Métodos de medición
  - 2.3.1. Método GQM
  - 2.3.2. Métodos de medición de productos
  - 2.3.3. Otros métodos de medición
- 2.4. Herramientas de métricas

## 3. ESTIMACION DEL SOFTWARE

- 3.1. Definición
- 3.2. Tipos de estimación
  - 3.2.1. Estimación del tamaño
  - 3.2.2. Estimación del esfuerzo
  - 3.2.3. Estimación de costos

	<b>UNIVERSIDAD POPULAR DEL CESAR</b>	CODIGO: 201-300-PRO05-FOR01
		VERSIÓN: 1
PLAN DE ASIGNATURA		PÁG: 6 de 7

- 3.3. Modelos de estimación
  - 3.3.1. Modelos de estimación del tamaño
  - 3.3.2. Modelos de estimación del esfuerzo
  - 3.3.3. Modelos de estimación de costos
- 3.4. Herramientas de estimación

#### 4. CALIDAD EN EL SOFTWARE

- 4.1. Conceptos y generalidades
- 4.2. Modelos de calidad
  - 4.2.1. Modelos de calidad de producto
  - 4.2.2. Modelos de calidad de proceso

### EVALUACIÓN

- Primer parcial 30%. Unidades 1
- Segundo parcial 30% Unidades 2 y 3
- Examen final 40%
  - Tercer parcial 20%. Unidades 3 y 4
  - Proyecto 20 % Aplicación de la materia.

#### Primer, segundo y tercer parcial

La evaluación se conformará: Con pruebas escrita, talleres, exposiciones, casos de estudios, prácticas donde se apliquen los conceptos adquiridos. Además, se asignará un porcentaje del 5%, medirá el desempeño, comportamiento, habilidades y competencias del estudiante durante el desarrollo, con el propósito de estimular e incentivar al estudiante a ser cada día más competitivo

#### Proyecto

La evaluación del proyecto final de la materia se realizará en tres fases: La primera fase, será la presentación de del proyecto aplicando las técnicas de pruebas, el diseño de casos de pruebas y la ejecución de pruebas. Segunda fase se aplicara al proyecto las técnicas de las métricas del producto. Tercera la aplicación de las métricas y estimación del software.

### REFERENCIAS BIBLIOGRÁFICAS

- R. Pressman. Ingeniería del Software: Un Enfoque Práctico. 7 Ed. McGraw Hill, 2010.
- Sommerville Ian. Ingeniería del Software. Pearson 9a. Edición. 2011.



- S. Sanchez, M. Sicilia, D. Rodriguez. Ingeniería del software. Un enfoque desde la guía SWEBOK. Alfaomega. 2012.
- Campos Olguín, Víctor. Ingeniería de Software. Pearson Educación, 2011.
- ega Fagoaga, Juan Carlos. Estimación de costos y administración de proyectos de Software. Mcgraw-Hill Interamericana, 2008.
- Braude Eric. Ingeniería de Software. Una perspectiva orientada a objetos. Alfaomega Grupo Editor S.A. 2003.
- Daniel Bolaños, Almudena Sierra, Idoia Alarcón. Pruebas de software y JUnit. Un análisis en profundidad y ejemplos prácticos. Pearson Education. 2007.
- Rumbaugh J., Jacobson I., Booch G. El Lenguaje Unificado de Modelado. Manual de Referencia. Editorial Addison-Wesley – 2000
- Booch G; Rumbaugh J; Jacobson I. El Lenguaje Unificado de Modelado. UML 2.0 Addison Wesley. 2006.

### Lecturas Obligatorias.

Las lecturas más afines que el estudiante encontrará para la comprensión del programa de esta asignatura son:

- S. Sánchez, M. Sicilia, D. Rodríguez. Ingeniería del software. Un enfoque desde la guía SWEBOK. Capítulos 7, 9 y 10.
- Roger S. Pressman. Ingeniería del Software. Capítulos 13, 14,15, 21,22,23,24,25 y 26
- Ian Sommerville. Ingeniería de Software. Novena edición. Addison Wesley. Capítulos 23 y 26.

#### - Lecturas Complementarias.

- Juan Fernandez, Guia de pruebas del software orientada a objetos, <http://www.uv.mx/personal/jfernandez/files/2010/07/Pruebas-de-Integracion.pdf>
- Metricas para sistemas orientados a objetos, [http://catarina.udlap.mx/u\\_dl\\_a/tales/documentos/lis/gonzalez\\_d\\_h/capitulo6.pdf](http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/gonzalez_d_h/capitulo6.pdf)
- Modelos de estimación del software, [http://sinbad2.ujaen.es/cod/archivosPublicos/pfc/pfc\\_jesus\\_alvarez.pdf](http://sinbad2.ujaen.es/cod/archivosPublicos/pfc/pfc_jesus_alvarez.pdf)
- Juan Fernandez, Calidad del software, <http://cmapspublic3.ihmc.us/rid=1MTZYCKHQ-1GTD1CC-15CQ/Calidad.pdf>

#### - Lecturas Sugeridas.

- Pruebas de software con JUnit, <http://elvex.ugr.es/decsai/java/pdf/8C-JUnit.pdf>