



No.	Descripción		
	<b>Código</b> 283	<b>Créditos</b> 4	
1	<b>Escuela</b> Ciencias y Sistemas	<b>Área a la que pertenece</b> Software	<b>Vigencia</b> Primer Semestre 2025
2	<b>Horas por semana</b> 3.5	<b>Horario</b> Sábados de 7:10 a 10:30 horas – Edificio : Meet	
3	<b>Pre-requisitos:</b> Sistemas de Bases de Datos 1 (774)		
4	<b>Postrequisitos:</b> Análisis y Diseño de Sistemas 2 (785)		
5	<b>Secciones:</b> A, B		
6	<p><b>I. Descripción General</b></p> <p>El curso de Análisis de Sistemas es un curso especializado de la carrera de Ingeniería en Ciencias y Sistemas, comprendido en el área de Software. Es un curso que describe actividades a ser tomadas en cuenta en un proyecto de desarrollo de software, en el contexto de las metodologías establecidas para ello. En el curso se presentan diferentes metodologías de desarrollo de software. Además los conceptos y técnicas relacionadas con crear modelos que representen al negocio, para utilizarlos como base en la definición de un sistema informático. Por último se considera la definición de requerimientos para la construcción de un sistema informático, considerando la satisfacción de las necesidades de los usuarios.</p> <p><b>II. Objetivos</b></p> <p>Proveer al alumno una definición de lo que es una metodología de desarrollo de software</p> <ul style="list-style-type: none"> <li>Presentar al alumno las principales metodologías de desarrollo de software aplicadas en la actualidad.</li> <li>Proveer de contenidos que ayuden al estudiante a entender y desarrollar modelos de negocio</li> <li>Proveer de contenidos que ayuden al estudiante a entender y desarrollar métodos para la definición de requerimientos para construcción de sistemas informáticos.</li> </ul> <p><b>III. Contenido</b></p> <ol style="list-style-type: none"> <li><b>1. Introducción a la ingeniería de software</b> <ol style="list-style-type: none"> <li>1.1. Desarrollo de Software Profesional</li> <li>1.2. Ingeniería de Software</li> <li>1.3. Diversidad de la Ingeniería de Software</li> <li>1.4. Ingeniería de Software y la Web</li> <li>1.5. Ética en la ingeniería de Software</li> </ol> </li> <li><b>2. Procesos de Software</b> <ol style="list-style-type: none"> <li>2.1. Modelos de proceso de Software</li> <li>2.2. Actividades del proceso</li> <li>2.3. Como enfrentar el cambio</li> </ol> </li> <li><b>3. Ingeniería de Requerimientos</b> <ol style="list-style-type: none"> <li>3.1. Requerimientos funcionales y no funcionales</li> <li>3.2. El documento de requerimientos de software</li> <li>3.3. Especificación de Requerimientos</li> <li>3.4. Procesos de ingeniería de requerimientos</li> <li>3.5. Adquisición y análisis de requerimientos</li> <li>3.6. Validación de requerimientos</li> <li>3.7. Administración de requerimientos</li> </ol> </li> <li><b>4. Modelado del Sistema</b> <ol style="list-style-type: none"> <li>4.1. Modelos de Contexto</li> <li>4.2. Modelos de Interacción</li> <li>4.3. Modelos estructurales</li> <li>4.4. Modelos de comportamiento</li> <li>4.5. Ingeniería dirigida por modelo</li> </ol> </li> <li><b>5. Métodos Ágiles de Desarrollo</b> <ol style="list-style-type: none"> <li>5.1. Métodos ágiles</li> <li>5.2. Desarrollo dirigido por un plan y desarrollo ágil</li> <li>5.3. Programación Extrema</li> <li>5.4. Administración de un proyecto ágil</li> <li>5.5. Escalamiento de métodos ágiles</li> <li>5.6. SCRUM</li> </ol> </li> <li><b>6. Gestión de Software</b> <ol style="list-style-type: none"> <li>6.1. Gestión de proyectos</li> </ol> </li> </ol>		



- 6.1.1. Gestión del riesgo
- 6.1.2. Gestión de personal
- 6.1.3. Trabajo en equipo
- 6.2. Planeación de proyectos
  - 6.2.1. Fijación de precio al software
  - 6.2.2. Desarrollo dirigido por un plan
  - 6.2.3. Planeación ágil
  - 6.2.4. Técnicas de estimación
- 6.3. Gestión de la Calidad
  - 6.3.1. Calidad del software
  - 6.3.2. Estándares de software
  - 6.3.3. Revisiones e inspecciones
  - 6.3.4. Medición y métricas del software

**Laboratorio:**

1. Metodologías de Desarrollo
2. Metodologías ágiles (XP y SCRUM)
3. Administración de procesos de negocio (BPMN)
4. Administración de requerimientos con RUP
5. Pruebas de Software

**IV. Metodología:**

Clase magistral impartida 2 veces por semana  
 Clase laboratorio impartida 1 vez por semana  
 Proyecto de clase  
 Tareas y cortos de lecturas  
 Presentaciones, discusiones de lecturas y participación  
 Autoestudio

**V. Evaluación:**

Aspecto	Valor
Evaluaciones parciales	35
1er parcial (20)	
2do parcial (15)	
Tareas, presentaciones y participación en clase	5
Exámenes cortos en clase	5
2 Proyectos de clase	20
Laboratorio	10
<b>Total zona</b>	<b>75</b>
<b>Examen final</b>	<b>25</b>

**VI. Requisitos para el curso**

Es obligatorio aprobar el proyecto de clase para tener derecho a examen final, se aprueba con 61/100

Es obligatorio aprobar el laboratorio para tener derecho a examen final, se aprueba con 61/100

Es obligatorio aprobar el curso ISE.

Solo se calificarán exámenes y proyectos de estudiantes asignados en el curso. NO se agregan estudiantes a actas

Es obligatoria una asistencia a la clase con promedio del 70%.

7	Bibliografía	<b>Ingeniería de Software Novena Edición, Ian Sommerville, Pearson</b>
8	No. De Secciones	2
9	Catedráticos titulares y auxiliares	Ing. Edgar Francisco Rodas Robledo
10	Coordinador de Área	Ing. Marlon Orellana
11	Director de Escuela	Ing. Carlos Alonso
12	Correo Electrónico	