

## 1. DATOS DE LA ASIGNATURA

Nombre de la asignatura:	<b>Proyecto Integrador de Ingeniería del Software</b>
Carrera:	<b>Ing. en Sistemas Computacionales</b>
Clave de la asignatura:	<b>ISD-2107</b>
SATCA	<b>2-3-5</b>

## 2. HISTORIA DEL PROGRAMA

<b>Lugar y fecha de elaboración o revisión</b>	<b>Participantes</b>	<b>Observaciones (cambios y justificación)</b>
Instituto Tecnológico de Culiacán. Culiacán, Sinaloa México. 30 de abril de 2021.	<ul style="list-style-type: none"><li>• MC Martha E. Valenzuela Tirado.</li><li>• Dr. Clemente García Gerardo.</li><li>• Dr. Ricardo R. Quintero Meza.</li></ul>	Elaboración de asignatura: Proyecto Integrador de Ingeniería de Software.

## 3. PRESENTACIÓN

### **Caracterización de la asignatura**

Esta materia se encuentra ubicada en la especialidad de Ingeniería del Software de la carrera de Ingeniería en Sistemas Computacionales, se recomienda que se curse en el noveno semestre. Esta materia sucede las materias; Pruebas del Software, Métodos Ágiles y Temas Selectos de Bases de Datos. Estas materias han otorgado al alumno diferentes competencias que podrán poner en práctica en el desarrollo de este proyecto.

Este curso aporta al perfil del alumno las competencias necesarias para desarrollar una aplicación software orientada a objetos, con un enfoque sistemático y disciplinado, utilizando como marco de trabajo el Proceso Unificado. Además, promueve el fortalecimiento de las diversas competencias adquiridas en el transcurso de la carrera, ya que será necesario ponerlas en práctica para lograr la complejidad y calidad del proyecto bajo desarrollo.

En esta asignatura se promueve la aplicación práctica de un proceso bien definido y bien manejado como factor clave para el desarrollo de proyectos de software exitosos.

### **Intención didáctica**

Esta materia debe ser abordada bajo un enfoque teórico-práctico en el cual el maestro se constituye en un facilitador del desarrollo de las competencias generales, específicas y genéricas en el alumno. El alumno deberá participar en cada una de las estrategias de aprendizaje diseñadas por el maestro, donde éstas favorecerán principalmente el saber y el saber hacer.

La materia se organiza en seis unidades que pretenden dirigir al alumno hacia el desarrollo de una aplicación software bajo un marco de proceso sistemático, disciplinado y cuantificable. La intención didáctica de cada unidad es la siguiente:

En la Unidad 1 se abordan conceptos relativos al paradigma orientado a objetos, identificación del espacio; del problema y de la solución, así como una descripción de los elementos que caracterizan al Proceso Unificado. En esta unidad el alumno deberá desarrollar competencias específicas relativas al saber y competencias genéricas relativas a su capacidad de comunicación oral y escrita, de relación interpersonal y ética.

En la Unidad 2 Se abordan conceptos relativos a la Fase de Inicio del Proceso Unificado; objetivos, procesos elementales de negocio, requisitos funcionales y requisitos no funcionales. En esta unidad se promueve la construcción (saber hacer) del Modelo de Requisitos para un caso de estudio, basado en los principios, prácticas y valores del Proceso Unificado.

En la Unidad 3 se aborda la Fase de Elaboración describiendo elementos relativos al Modelo del Dominio; clases conceptuales, atributos y relaciones, así como las actividades y artefactos que establece el proceso unificado para lograr su desarrollo. En esta unidad se promueve la construcción (saber hacer) del Modelo del Dominio para un caso de estudio, basado en los principios, prácticas y valores del Proceso Unificado.

En la Unidad 4 se aborda la Fase de Elaboración describiendo elementos relativos al Modelo de Diseño; clases software, diagramas de interacción, principios y patrones de diseño, así como las actividades y artefactos que establece el Proceso Unificado para lograr su desarrollo. En esta unidad se promueve la construcción (saber hacer) del Modelo de Diseño para un caso de estudio, basado en los principios, prácticas y valores del Proceso Unificado.

En la Unidad 5 se aborda la Fase de Construcción describiendo elementos relativos al Modelo de Implementación; transformación del modelo de diseño a su representación software y, al Modelo de Pruebas; tipos de pruebas y diseño de los casos, así como las actividades y artefactos que establece el Proceso Unificado para lograr su desarrollo. En esta unidad se promueve la construcción (saber hacer) del Modelo de Implementación y el Modelo de Pruebas para un caso de estudio, basado en los principios, prácticas y valores del Proceso Unificado.

En la Unidad 6 se aborda la Fase de Transición describiendo elementos relativos al Modelo de Componentes y al Modelo de Despliegue, así como las actividades y artefactos que establece el proceso unificado para lograr su desarrollo. En esta unidad se promueve la construcción (saber hacer) del Modelo de Componentes y el Modelo de Despliegue para un caso de estudio, basado en los principios, prácticas y valores del Proceso Unificado.

En las unidades 2 a la 6 el alumno realizará actividades relativas al desarrollo de software bajo un marco de proceso, utilizando una notación y una herramienta CASE, es muy importante que el estudiante entienda, comprenda y valore dichas actividades, debido a que se corresponden con su quehacer futuro y deberá actuar en consecuencia de una manera profesional; de igual manera, aprecie la importancia del conocimiento y los hábitos de trabajo; desarrolle la precisión y la curiosidad, la puntualidad, el entusiasmo y el interés, la tenacidad, la flexibilidad y la autonomía. Por lo que el alumno deberá desarrollar su capacidad de aplicar los conocimientos en la práctica,

de organizar y planificar, de resolver problemas, de trabajar en equipos interdisciplinarios, de diseño y gestión de proyectos y de liderazgo bajo un compromiso ético.

#### 4. COMPETENCIAS A DESARROLLAR:

##### **Competencia general:**

Desarrollar una aplicación software mediante el uso integral de las diversas competencias de ingeniería del software que el alumno ha desarrollado a lo largo de su carrera.

##### **Competencias específicas:**

- Identificar los elementos que caracterizan el marco de proceso del Proceso Unificado.
- Elaborar los artefactos correspondientes a la Fase de Inicio para un caso de estudio, basado en los principios de comunicación, descubrimiento y registro de requisitos.
- Identificar los elementos que componen un modelo de domino y las actividades y tareas específicas que establece el Proceso Unificado para realizarlo.
- Formular una solución software aplicando principios y patrones de diseño, así como los principios, prácticas y valores del Proceso Unificado.
- Transformar los modelos de diseño a sus representaciones software para la generación y despliegue de la aplicación software, así como diseñar los casos de prueba necesarios para asegurar la calidad de la misma.
- Construir el modelo de componentes (arquitectura software) y de instalación de la aplicación software, basado en los artefactos que establece el Proceso Unificado.

##### **Competencias genéricas:**

###### **Competencias instrumentales**

- Capacidad de planificar y organizar.
- Solución de problemas.
- Comunicación oral y escrita.

###### **Competencias interpersonales**

- Habilidades interpersonales.
- Capacidad de trabajar en equipo interdisciplinario.
- Compromiso ético.

###### **Competencias sistémicas**

- Capacidad de aplicar los conocimientos en la práctica.
- Capacidad para diseñar y gestionar proyectos.
- Liderazgo.

#### 5. COMPETENCIAS PREVIAS

- Diseñar el proceso de pruebas orientado a asegurar y comprobar la calidad del software.
- Desarrollar soluciones software que hagan uso del modelo de bases de datos apropiado al problema planteado.
- Desarrolla soluciones software utilizando métodos de ingeniería de software con enfoque ágil.

## 6. TEMARIO

Unidad	Temas	Subtemas
1	Introducción	1.1 ¿Qué es el análisis y el diseño OO? 1.2 El espacio del problema (análisis) y el espacio de la solución (diseño). 1.3 Desarrollo iterativo, evolutivo y ágil. 1.4 El Framework del UP.
2	Modelado de requisitos	2.1 Requisitos funcionales y no funcionales. 2.2 Identificación de objetivos y procesos elementales de negocio. 2.3 Trazabilidad de procesos elementales de negocio a casos de uso (UC). 2.4 Escritura de UC. 2.5 Diagramas de UC básico y avanzado. 2.6 Trazabilidad de UC a requisitos. 2.7 Prototipos de interfaz de usuario. 2.8 El glosario. 2.9 Requisitos no funcionales: el documento de especificación complementaria. 2.10 El glosario y el Development Case.
3	Modelado estructural	3.1 El modelo del dominio: conceptos, atributos y relaciones. 3.2 Identificación de conceptos. 3.3 Generalización y especialización. 3.4 Arquitectura software y paquetes UML. 3.5 Diagrama de clases de diseño. 3.6 Cardinalidad y roles. 3.7 Polimorfismo. 3.8 Interfaces. 3.9 Agregación y composición. 3.10 Visibilidad de atributos, parámetros, local y global.
4	Modelado dinámico	4.1 La esencia del modelado dinámico: asignación de responsabilidades. 4.2 Tarjetas CRC. 4.3 El Diagrama de Secuencia de Sistema. 4.4 Los contratos. 4.5 Diagramas de interacción: de secuencia y colaboración. 4.6 Asignación de responsabilidades (patrones GRASP: experto en información, creador, alta cohesión, bajo acoplamiento, controlador). 4.7 Introducción a patrones GoF: Adapter, Strategy, Observer, Facade. 4.8 Diagramas de estado y actividad.
5	Mapeo del diseño al código	5.1 Mapeo a clases software a partir de los modelos estructural y dinámico.

		5.2 Mapeo a colecciones a partir del modelo estructural. 5.3 Orden de construcción de las clases software a partir del modelo estructural. 5.4 Definición de pruebas unitarias y refactoring.
6	Modelado de componentes e instalación	6.1 Definición de componentes a partir de los modelos estructurales (arquitectura software y clases). 6.2 Definición del modelo de instalación (nodos y enlaces entre nodos). 6.3 Definición de instalación de componentes en nodos.

## 7. SUGERENCIAS DIDÁCTICAS PARA EL DESARROLLO DE COMPETENCIAS ESPECÍFICAS

Competencias específicas	Actividades de aprendizaje
<b>Unidad 1. Introducción</b> <ul style="list-style-type: none"> <li>Identificar los elementos que caracterizan el marco de proceso del Proceso Unificado.</li> </ul>	<ul style="list-style-type: none"> <li>Elaborar un mapa mental/conceptual sobre los elementos que caracterizan al UP.</li> <li>Identificar las buenas prácticas que ofrece el marco del Proceso Unificado mediante debates o mesas redondas.</li> </ul>
<b>Unidad 2. Modelado de requisitos</b> <ul style="list-style-type: none"> <li>Elaborar los artefactos correspondientes a la Fase de Inicio para un caso de estudio, basado en los principios de comunicación, descubrimiento y registro de requisitos.</li> </ul>	<ul style="list-style-type: none"> <li>Construir los artefactos correspondientes a la Fase de Inicio, utilizando una herramienta CASE, para un caso de estudio basado en los principios, prácticas y valores del Proceso Unificado.</li> </ul>
<b>Unidad 3. Modelado estructural</b> <ul style="list-style-type: none"> <li>Identificar los elementos que componen un modelo de dominio y las actividades y tareas específicas que establece el Proceso Unificado para realizarlo.</li> </ul>	<ul style="list-style-type: none"> <li>Construir un Modelo del Dominio, utilizando una herramienta CASE, para un caso de estudio, mediante las disciplinas y artefactos que establece el Proceso Unificado.</li> </ul>
<b>Unidad 4. Modelado dinámico</b> <ul style="list-style-type: none"> <li>Formular una solución software aplicando principios y patrones de diseño, así como los principios, prácticas y valores del Proceso Unificado.</li> </ul>	<ul style="list-style-type: none"> <li>Desarrollar un Modelo de Diseño, utilizando una herramienta CASE, para un caso de estudio, mediante las disciplinas y artefactos que establece el Proceso Unificado.</li> <li>Desarrollar el modelo de base de datos para el caso de estudio, mediante el uso de un sistema gestor de bases de datos orientado a objetos.</li> </ul>

<b>Unidad 5. Mapeo del diseño al código</b> <ul style="list-style-type: none"> <li>Transformar los modelos de diseño a sus representaciones software para la generación y despliegue de la aplicación software, así como identificar los tipos de pruebas necesarias para asegurar la calidad de la misma.</li> </ul>	<ul style="list-style-type: none"> <li>A partir de los modelos de diseño codificar la aplicación software, utilizando un lenguaje de programación orientado a objetos, mediante un proceso manual-semiautomático.</li> <li>Diseñar el modelo de pruebas adecuado para asegurar la calidad de la aplicación.</li> <li>Efectuar una retrospección del proyecto realizado para su evaluación y posible mejora.</li> </ul>
<b>Unidad 6. Modelado de componentes e instalación</b> <ul style="list-style-type: none"> <li>Construir el modelo de componentes (arquitectura software) y de instalación de la aplicación software, basado en los artefactos que establece el Proceso Unificado.</li> </ul>	<ul style="list-style-type: none"> <li>Crear modelos de componentes y de instalación, utilizando una herramienta CASE, para una aplicación software, mediante los principios, prácticas y valores del Proceso Unificado.</li> </ul>

## 8. SUGERENCIAS DIDÁCTICAS PARA EL DESARROLLO DE COMPETENCIAS GENÉRICAS.

<ul style="list-style-type: none"> <li>Propiciar actividades de búsqueda, selección y análisis de información en distintas fuentes.</li> <li>Fomentar actividades grupales que propicien la comunicación, el intercambio argumentado de ideas, la reflexión, la integración y la colaboración de los estudiantes.</li> <li>Desarrollar actividades de aprendizaje que propicien la aplicación de los conceptos que se van aprendiendo en el desarrollo de la asignatura.</li> <li>Definir proyectos de desarrollo de software que busquen resolver problemas reales organizados con equipos que definan roles y responsabilidades bien establecidos. Resaltar la importancia del liderazgo y la ética en el funcionamiento del equipo.</li> <li>Propiciar el uso adecuado de conceptos y de terminología científico-tecnológica.</li> </ul>
---

## 9. SUGERENCIAS DE EVALUACIÓN

Diagnóstica	Formativa	Sumativa
<ul style="list-style-type: none"> <li>Aplicación de Cuestionario de intereses profesionales</li> <li>Realización de Lluvia de ideas enfocada a la exploración de conocimientos previos de Ingeniería de Software, Bases de Datos, Pruebas de Software y métodos de</li> </ul>	El alumno mostrará al grupo el avance del proyecto para recibir retroalimentación.	Exámenes parciales (30%).  Proyecto (60%).  Desempeño en actividades genéricas (10%).

desarrollo de software.		
-------------------------	--	--

## 10. FUENTES DE INFORMACIÓN

- **Libro de texto**
  - Craig Larman, *UML y Patrones, Una introducción al análisis y diseño orientado a objetos y al Proceso Unificado*, Prentice Hill.2003. ISBN: 84-205-3438-2.
- **Lecturas complementarias**
  - Stephen R. Schach, *Análisis y diseño orientado a objetos con UML y el proceso unificado*, McGrawHill.
  - Ivar Jacobson, Grady Booch, James Rumbaugh. *The Unified Software Development Process*, Addison Wesley, 1999. ISBN 0-201-57169-2.

## 11. PERFIL DEL PROFESOR QUE IMPARTIRÁ LA MATERIA

- Ingeniero en Sistemas Computacionales, Licenciado en Informática o carrera afín con experiencia práctica en desarrollo de aplicaciones orientadas a objetos con el Proceso Unificado.
- Maestro en Ciencias Computacionales.
- Doctor en Ciencias Computacionales.