

Misión del Centro Universitario

Somos un centro que forma parte de la Red Universitaria de la Universidad de Guadalajara. Como institución de educación superior pública asumimos el compromiso social de satisfacer necesidades de formación y generación de conocimiento en el campo de las ciencias exactas y las ingenierías. La investigación científica y tecnológica, así como la vinculación y extensión, son parte fundamental de nuestras actividades para incidir en el desarrollo de la sociedad; por lo que se realizan con vocación internacional, humanismo, calidad y pertinencia.

1.- Identificación de la Unidad de Aprendizaje

Nombre de la Unidad de Aprendizaje

INGENIERÍA DE SOFTWARE I

Clave de la UA	Modalidad de la UA	Tipo de UA	Valor de créditos	Área de formación
I5898	Presencial	Curso	8	Básico particular
Horas/semana	Horas teoría/semestre	Horas práctica/ semestre	Total de horas	Seriación
4	51	17	68	N/A
Departamento		Academia		
Ciencias Computacionales		Ingeniería de Software		

Presentación

La Ingeniería del Software forma a profesionales preparados para la definición, especificación, diseño, evaluación e integración de sistemas informáticos o de información, así como para el ejercicio de cargos de responsabilidad en empresas del sector.

Competencia de la Unidad de Aprendizaje (UA)

El estudiante elaborará el modelado de un sistema de información empleando metodologías, técnicas y herramientas para construir una propuesta de solución a un problema determinado.

Tipos de saberes

Saber (conocimientos)	Saber hacer (habilidades)	Saber ser (actitudes y valores)
<ol style="list-style-type: none">Identificar diferentes metodologías de desarrollo de software.Identificar las características principales de las metodologías.Identificar el proceso de factibilidad desde un punto de vista técnico, económico y operativo con el cual se miden los aspectos del éxito de un proyectoIdentificar el proceso de recolección, clasificación y priorización de requerimientos basado en el dominio de la aplicación.Identificar el proceso de documentación de requerimientos con base en un estándar de calidad.Identificar técnicas de validación de requerimientos.Identificar los fundamentos del Lenguaje de Modelado Unificado (UML).Identificar los componentes de un caso de uso (actores, caso de uso).Identificar los componentes de un diagrama de clase (clase, atributos, métodos, relaciones).Identificar la interacción de los objetos y actores de los casos de uso del sistema.	<ol style="list-style-type: none">Seleccionar la metodología adecuada al desarrollo de la aplicación.Realizar un estudio de factibilidad: técnica, económica y operativa.Obtener requerimientos del sistema mediante las técnicas de recolección.Elaborar un documento de especificación de requerimientos del sistema utilizando un estándar de calidad.Comprobar que los requerimientos sean consistentes, completos y que correspondan con el dominio de la aplicación.Elaborar los diagramas de casos de uso.Elaborar el diagrama de clases.Elaborar los diagramas de secuencia del sistema.Elaborar los diagramas de colaboración del sistema.Elaborar los diagramas de estado del sistema.Elaborar los diagramas de componentes, implementación, despliegue y actividad.Diseñar casos de prueba de software.	<ol style="list-style-type: none">AnalíticoHábil para el trabajo en equipoDisciplinadoSistemáticoOrganizadoProactivoHábil para comunicarse de forma oral y escritaGenerar competencias de trabajo colaborativo.Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad.Capacidad para saber comunicar asertivamente y transmitir los conocimientos, habilidades y destrezasFomentar el espíritu emprendedor.

<p>11. Identificar el paso de mensajes entre los diferentes objetos del sistema.</p> <p>12. Identificar los estados de los procesos del sistema.</p> <p>13. Identificar los elementos de los diagramas de componentes, implementación, despliegue y actividad.</p> <p>14. Identificar el concepto de prueba de software y el objetivo de las mismas.</p> <p>15. Identificar los elementos de un manual técnico de un sistema de información.</p> <p>16. Identificar el plan y mecanismo de instalación.</p> <p>17. Identificar los elementos del manual de instalación de un sistema de Información.</p> <p>18. Conocimiento y aplicación de los principios, metodologías y ciclos de vida de la ingeniería de software.</p> <p>19. Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.</p> <p>20. Conocimiento, administración y mantenimiento sistemas, servicios y aplicaciones informáticas.</p>	<p>13. Elaborar un manual técnico de un sistema de información.</p> <p>14. Elaborar un manual usuario de un sistema de información.</p> <p>15. Elaborar un manual mantenimiento de un sistema de información.</p> <p>16. Capacidad de identificar y analizar problemas y diseñar, desarrollar, implementar, verificar y documentar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales.</p>	
--	---	--

Competencia genérica	Competencia profesional
<p>Analizar, Diseñar e Implementar soluciones informáticas, acorde a los requerimientos establecidos, para la administración de la información.</p>	<p>Desarrollar un sistema de información empleando las técnicas, metodologías y herramientas de diseño, pruebas y liberación necesarias para garantizar la implementación, de acuerdo a los requerimientos del cliente.</p> <p>Esta asignatura aporta al perfil de egreso las competencias profesionales para aplicar métodos y técnicas que permitan desarrollar soluciones de software, conforme a las normas organizacionales de manejo y seguridad de la información, utilizando tecnologías emergentes.</p> <p>La importancia de esta asignatura, es que permite al alumno abordar todas las fases del ciclo de vida del desarrollo de cualquier tipo de software, aplicables a una infinidad de áreas. Esta asignatura, brinda el conocimiento científico, a través de los métodos y técnicas adecuados, para el desarrollo de software.</p>
Competencias previas del alumno	
<p>Establecer soluciones informáticas través de la implementación de la metodología orientada a objetos.</p> <p>Uso de algún Sistema Manejador de Bases de Datos.</p> <p>Dominio de algún Lenguaje de Programación.</p> <p>Uso de software de presentaciones gráficas.</p>	
Competencia del perfil de egreso	
<p>El egresado de ingeniería informática contará con la formación intelectual y los conocimientos básicos en los campos de los sistemas de información, la gestión de las tecnologías de la información, los sistemas robustos, paralelos y distribuidos y la computación</p>	

flexible (softcomputing), necesarios para mantenerse actualizado durante se ejercicio profesional, así como una formación ciudadana y humanista en beneficio de la sociedad.

El profesional de ingeniería en computación con especialidad en software de sistemas podrá diseñar y desarrollar sistemas de software de base (los sistemas de programación primordiales en una computadora); interactuar con subsistemas digitales y de telecomunicaciones (redes); diseñar e implantar sistemas operativos; diseñar y concebir nuevos lenguajes de programación, así como construir traductores (compiladores); especificar arquitecturas de computadora y desarrollar el software de aplicación que le compete. El profesional de ingeniería en computación con especialidad en sistemas digitales podrá diseñar, construir, instalar, operar y dar mantenimiento a sistemas digitales e interfaces aplicables a la tecnología computacional y a la teleinformática; diseñar e implantar organizaciones de computadoras y desarrollar la realización electrónica que le compete; diseñar e instrumentar herramientas de software necesarias para el manejo del hardware; concebir, diseñar y construir hardware computacional que satisfaga definiciones de funcionabilidad o fines específicos.

Perfil deseable del docente

- Conocer la disciplina, su origen y desarrollo para considerar este conocimiento al abordar los temas.
- Desarrollar la capacidad para coordinar y trabajar en equipo.
- Orientar el trabajo del estudiante y desarrollar en él la autonomía, el trabajo cooperativo y la toma de decisiones.
- Mostrar flexibilidad en el seguimiento del proceso formativo y propiciar la interacción entre los estudiantes.
- Tomar en cuenta el conocimiento de los estudiantes como punto de partida y como obstáculo para la construcción de nuevos conocimientos.
- Autoregular los procesos de aprendizaje.
- Propiciar actividades de búsqueda, selección y análisis de información en distintas fuentes.
- Fomentar actividades grupales que propicien la comunicación, el intercambio argumentado de ideas, la reflexión, la integración y la colaboración de y entre los estudiantes.
- Relacionar los contenidos de esta asignatura con las demás del plan de estudios a las que ésta da soporte para desarrollar una visión interdisciplinaria en el estudiante.
- Propiciar el desarrollo de capacidades intelectuales relacionadas con la lectura, la escritura y la expresión oral.
- Propiciar el desarrollo de actividades intelectuales de inducción-deducción y análisis-síntesis, que encaminen hacia la investigación. Desarrollar actividades de aprendizaje que propicien la aplicación de los conceptos, modelos y metodologías que se van aprendiendo en el desarrollo de la asignatura.
- Proponer problemas que permitan al estudiante la integración de contenidos de la asignatura y entre distintas asignaturas, para su análisis y solución.

COMPETENCIAS TÉCNICO PEDAGÓGICAS:

- Usa y maneja ambientes virtuales para el proceso de enseñanza aprendizaje
- Utiliza las TIC para diversificar y fortalecer las estrategias de aprendizaje por competencias

COMPETENCIAS PROFESIONALES:

- DISCIPLINA: Informática, Tecnología de la Información, Sistemas Computacionales, Computación o afines
- NIVEL ACADEMICO: igual o mayor a nivel superior
- EXPERIENCIA DOCENTE: 3 años en nivel superior
- EXPERIENCIA PROFESIONAL: 3 años en el área.

2.- Contenidos temáticos

Contenido	Horas
UNIDAD I. Procesos del Software	8
1. Definiciones básicas y conceptos	(1 Hora)
2. Ciclo de vida del software	(2 Horas)
3. Metodologías de desarrollo	(4 Horas)
a. Definición	
b. Herramientas e instrumentos	
c. Notación	
4. Características y tipos de Software	(1 Hora)
UNIDAD II. Análisis	12
1. Ingeniería de Requerimientos	(2 Horas)
a. Técnicas e instrumentos de recolección de datos	
b. Plan de aplicación de técnicas	
c. Especificación de requerimientos IEEE 830 rev 1998	
2. Planeación de Recursos	(2 Horas)

3. Diagrama de Gantt	(4 Horas)	
4. Diagrama de Pert	(4 Horas)	
UNIDAD III. Diseño Arquitectónico del Software		28
1. Diagrama de Casos de Uso	(2 Horas)	
2. Definición de Casos de Uso	(1 Hora)	
3. Identificación de Actores	(1 Hora)	
4. Diagrama de Bloques	(2 Horas)	
5. Diseño Semántico de Datos	(2 Horas)	
6. Tarjetas CRC	(2 Horas)	
7. Diccionario de Clases	(2 Horas)	
8. Diagrama de Clases	(2 Horas)	
9. Diagrama de Objetos	(2 Horas)	
10. Diagrama de Secuencia	(2 Horas)	
11. Diagrama de comunicación	(2 Horas)	
12. Diagrama de Estados	(2 Horas)	
13. Diagrama de Actividades	(2 Horas)	
14. Diagrama de Componentes	(2 Horas)	
15. Diagrama de Despliegue	(2 Horas)	
UNIDAD IV. Construcción de Software		12
1. Plan de Desarrollo	(4 Horas)	
2. Diseño e Implementación de Interfaces	(4 Horas)	
3. Implementación del Modelo Relacional	(4 Horas)	
UNIDAD V. Pruebas y Mantenimiento del Software		8
1. Pruebas del Software	(4 Horas)	
a. Definición, conceptos, tipos de pruebas y estándares		
b. Diseño de plan de pruebas		
2. Mantenimiento del Software	(4 Horas)	
a. Definición, conceptos y tipos		
b. Manual de usuario		
c. Manual Técnico		
Estrategias docentes para impartir la unidad de aprendizaje		
<ol style="list-style-type: none"> 1. Aprendizaje basado en resolución de problemas 2. Aprendizaje basado en casos de estudio 3. Diseño de organizadores gráficos. 4. Elaboración de estructuras textuales 5. Prácticas guiadas. 6. Aprendizaje basado en proyectos 7. Solución de problemas de un contexto específico. 8. Mapas Mentales. 9. Textos argumentativos. 10. Participación en foros y debates. 11. Presentaciones gráficas. 		
Bibliografía básica		
<ol style="list-style-type: none"> 1. IEEE Computer Society, USA. (2004) <i>Guide to the Software Engineering Body of Knowledge</i>. USA: IEEE 2. Peters, James F. & Pedrycz, Witold (2000). <i>Software Engineering: An Engineering Approach</i>. USA: WILEY 3. Sommersville Ian (2011) <i>Ingeniería del Software</i>. México: McGraw-Hill 		
Bibliografía complementaria		
<ol style="list-style-type: none"> 1. Pressman, Roger S. (2010) <i>Ingeniería del Software: Un enfoque práctico</i>. México: McGraw Hill 2. Braude, Erik J. (2007) <i>Ingeniería del Software: Una perspectiva orientada a objetos</i>. México: McGraw Hill 		
3.-Evaluación		
Evidencias		
Documentos de Recolección de Datos, Documento con la Especificación de Requerimientos, Diseño Arquitectónico, Modelo de Datos, Interfaz Gráfica de Usuario, Plan de Pruebas, Manual Técnico, Manual de Usuario.		

UNIDAD I: Documento que describa los conceptos de proceso del software, definiciones básicas, ciclo de vida del software y metodologías de desarrollo, así como la selección y justificación de la metodología seleccionada para el proyecto asignado por el profesor que se desarrollará en el transcurso de la materia.

UNIDAD II: Documento que describa los conceptos de ingeniería de requerimientos, técnicas e instrumentos de recolección de datos y planeación de recursos, así como la elaboración del plan de aplicación de técnicas, especificación de requerimientos y la planeación de los recursos para el proyecto asignado por el profesor que se desarrollará en el transcurso de la materia.

UNIDAD III: Documento que describa los elementos y características de los diagramas de casos de uso, definición de casos de uso, identificación de actores, diagrama de bloques, diseño semántico de datos, tarjetas CRC, diccionario de clases, diagramas de clases, objetos, secuencia, colaboración, estados, actividades, componentes y despliegue, así como la elaboración de los diagramas de casos de uso, definición de casos de uso, identificación de actores, diagrama de bloques, diseño semántico de datos, tarjetas CRC, diccionario de clases, diagramas de clases, objetos, secuencia, colaboración, estados, actividades, componentes y despliegue para el proyecto asignado por el profesor que se desarrollará en el transcurso de la materia.

UNIDAD IV: Documento que describa e identifique los conceptos de construcción de software con el plan de desarrollo, diseño de interfaces y el modelo relacional, así como la elaboración del plan de desarrollo, el diseño e implementación de interfaces y la implementación del modelo relacional para el proyecto asignado por el profesor que se desarrollará en el transcurso de la materia.

UNIDAD V: Documento que describa e identifique los conceptos de pruebas y mantenimiento de software, así como la elaboración del manual de pruebas, mantenimiento, técnico y de usuario para el proyecto asignado por el profesor que se desarrollará en el transcurso de la materia.

Tipo de evaluación

La evaluación de la asignatura debe ser continua y formativa a través de las actividades siguientes:

- Solución de casos prácticos solicitados durante las actividades, así como sus conclusiones de forma escrita.
- Tareas.
- Exposición.
- Participación en clase.
- Proyecto.
- Exámenes teóricos o prácticos.

Coevaluación para los entregables de la unidad I, en el caso de las unidades II, III, IV y V serán evaluadas solo con el profesor realizando retroalimentación de todas las actividades al término de su evaluación.

Criterios de Evaluación (% por criterio)

Actividades de Aprendizaje 50%

Consiste en las evidencias, producto del trabajo en clases, tareas y actividades; entregadas de manera presencial o a través de algún medio electrónico de acuerdo con la selección del profesor.

Exámenes Parciales 40%

Aplicación de dos exámenes durante el curso con valor de 20 puntos cada uno, en los cuales se evaluarán los aspectos del Saber y el Saber Hacer descritos en el documento, los cuales podrán ser aplicados de manera escrita o a través del apoyo de algún entorno de aprendizaje o plataforma virtual de acuerdo con la selección del profesor.

Actividad Integradora 10%

Aplicación informática que deberá estar conectada a una base de datos almacenada en un SBGD, la cual deberá contar con la documentación de análisis, diseño arquitectónico, diseño del modelo de datos, manuales de mantenimiento y de pruebas.

4.-Acreditación

De acuerdo al "REGLAMENTO GENERAL DE EVALUACIÓN Y PROMOCIÓN DE ALUMNOS DE LA UNIVERSIDAD DE GUADALAJARA":
Artículo 5. "El resultado final de las evaluaciones será expresado conforme a la escala de calificaciones centesimal de 0 a 100, en números enteros, considerando como mínima aprobatoria la calificación de 60."
Artículo 20. "Para que el alumno tenga derecho al registro del resultado final de la evaluación en el periodo ordinario, establecido en el calendario escolar aprobado por el H. Consejo General Universitario, se requiere:

- I. Estar inscrito en el plan de estudios y curso correspondiente, y
- II. Tener un mínimo de asistencia del 80% a clases y actividades registradas durante el curso.”

De acuerdo al “REGLAMENTO GENERAL DE EVALUACIÓN Y PROMOCIÓN DE ALUMNOS DE LA UNIVERSIDAD DE GUADALAJARA”:
 Artículo 27. “Para que el alumno tenga derecho al registro de la calificación en el periodo extraordinario, se requiere:

- I. Estar inscrito en el plan de estudios y curso correspondiente.
- II. Haber pagado el arancel y presentar el comprobante correspondiente.
- III. Tener un mínimo de asistencia del 65% a clases y actividades registradas durante el curso.”

5.- Participantes en la elaboración

Código	Nombre
2957038	MARÍA ISABEL CIBRIAN DECENA
2956582	ANGEL TONATIUH HERNÁNDEZ CASAS
2949676	HASSEM RUBÉN MACÍAS BRAMBILA
2006537	LUIS FELIPE MARISCAL LUGO
2127865	VÍCTOR MANUEL ZAMORA RAMOS