

## 1. Datos Generales de la asignatura

<b>Nombre de la asignatura:</b>	Ingeniería de Software
<b>Clave de la asignatura:</b>	TIC-1014
<b>SATCA<sup>1</sup>:</b>	2-2-4
<b>Carrera:</b>	Ingeniería en Tecnologías de la Información y Comunicaciones

## 2. Presentación

<b>Caracterización de la asignatura</b>
<ul style="list-style-type: none"> <li>• La asignatura aporta competencias al perfil para: Desarrollar e implementar sistemas de información para el control y la toma de decisiones utilizando metodologías basadas en estándares internacionales. Utilizar tecnologías emergentes y herramientas actuales para atender necesidades acordes al entorno y gestionar proyectos que involucren Tecnologías de la Información y Comunicaciones para el logro de los objetivos organizacionales conforme a requerimientos establecidos.</li> <li>• 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, es la aplicación práctica del conocimiento científico, a través de los métodos y técnicas adecuados, para el desarrollo de software.</li> <li>• El estudio de la asignatura permite al estudiante conocer qué es la Ingeniería de Software, los sistemas de información, los modelos de la ingeniería de software, cómo elaborar la planificación y el análisis de proyectos de software asegurando la calidad del mismo.</li> <li>• La disciplina de Ingeniería de Software se relaciona con materias precedentes como: Fundamentos de Programación, Programación Orientada a Objetos, Estructura y Organización de Datos y Fundamentos de Base de Datos, ya que aportan las competencias de la programación y el tratamiento de información, y en donde probablemente se han iniciado proyectos que requieran formalizarse con las técnicas de la Ingeniería del Software. Especialmente se requiere de las competencias adquiridas en Matemáticas para la toma de decisiones porque aporta la habilidad en el uso de técnicas de planeación y análisis de ruta crítica. Aporta competencias en la aplicación de técnicas formales en el desarrollo de software para asignaturas posteriores como: Taller de Ingeniería de Software, Programación Web y Negocios Electrónicos I y II.</li> <li>• Es sumamente importante que se oferte mínimo a la par con la asignatura de Administración de Proyectos, ya que la disciplina de la administración de proyectos es conveniente aplicarla al proyecto de desarrollo de software que en ésta asignatura se propone inicie. Lo anterior con la intención de favorecer el desarrollo de proyectos integradores.</li> </ul>
<b>Intención didáctica</b>
<ul style="list-style-type: none"> <li>• La asignatura se compone de 5 temas distribuidos de la siguiente manera:</li> <li>• En el primer tema, el estudiante conocerá los fundamentos de la Ingeniería de Software y los sistemas de información.</li> <li>• En el segundo tema, se conocerán los distintos tipos modelos de la Ingeniería de Software, además el estudiante podrá seleccionar el modelo más adecuado para el desarrollo del software.</li> </ul>

<sup>1</sup> Sistema de Asignación y Transferencia de Créditos Académicos

- En el tercer tema, se abordan temas para que el estudiante elabore una excelente planificación del proyecto, así como el uso y aplicación de las diferentes métricas de software.
- En el cuarto tema, se tratan temas relacionados con el modelado, para que el estudiante elabore al análisis del proyecto de software. En la quinta unidad, se abordan los temas de calidad del software, para que el estudiante se asegure de desarrollar software de calidad.
- El docente debe abordar la asignatura de manera teórico-práctica de modo que el estudiante trabaje en equipo, desarrolle capacidad de análisis y síntesis, liderazgo y toma de decisiones, así como gestión de proyectos de software mediante el uso de herramientas informáticas y aseguramiento de la calidad.
- En el tema 3.3 se sugiere abordar mínimamente las métricas de COCOMO I y II y Puntos de función. Se deja a criterio del docente abordar algunas otras.
- En el tema 4.4 se sugiere abordar mínimamente los diagramas de casos de uso, clases, secuencia, colaboración, actividades, componentes y despliegue. Se sugiere el uso de herramientas de modelado: StartUML(Libre), Rational Rose (Bajo Licencia), EnterpriseArchitect (Bajo Licencia) o DIA (Libre).
- En el tema 5.3 de modelos de calidad se sugiere abordar conceptos, elementos, características generales, ventajas, desventajas y mejores prácticas. Para el tema de CMMI existe una cantidad considerable de mejores prácticas, se sugiere abordar las más relevantes.
- En el tema 5.4 se sugiere estudiar distintas métricas orientadas a la calidad como las de McCall.

### 3. Participantes en el diseño y seguimiento curricular del programa

Lugar y fecha de elaboración o revisión	Participantes	Evento
Instituto Tecnológico Superior de Puerto Vallarta del 10 al 14 de agosto de 2009.	Representantes de los Institutos Tecnológicos de: Aguascalientes, Apizaco, Superior de Centla, Chetumal, Ciudad Cuauhtémoc, Ciudad Madero, Comitán, Delicias, León, Superior de Misantla, Pachuca, Pinotepa, Puebla, Superior de Puerto Vallarta, Roque, Tepic, Tijuana, Tuxtla Gutiérrez y Villahermosa.	Reunión Nacional de Diseño e Innovación Curricular para el Desarrollo y Formación de Competencias Profesionales de las Carreras de Ingeniería en Tecnologías de la Información y Comunicaciones, Ingeniería en Energías Renovables, Ingeniería Petrolera y Gastronomía.
Instituto Tecnológico de Villahermosa del 24 al 28 de mayo de 2010.	Representantes de los Institutos Tecnológicos de: Aguascalientes, Apizaco, Superior de Centla, Chetumal, León, Pachuca, Puebla, Roque, Tepic, Tuxtla Gutiérrez y Villahermosa.	Reunión Nacional de Consolidación de los Programas en Competencias Profesionales de las Carreras de Ingeniería en Geociencias, Ingeniería en Energías Renovables, Ingeniería en Tecnologías de la Información y Comunicaciones, y Gastronomía.

Instituto Tecnológico de Querétaro del 22 al 25 de octubre de 2012.	Representantes de los Institutos Tecnológicos de: Álvaro Obregón, Cd. Juárez, Cd. Valles, Cerro Azul, Chetumal, Coacalco, Delicias, Gustavo A. Madero, Cd. Madero, Múzquiz, Occidente del Estado de Hidalgo, Pachuca, Puerto Vallarta, Salvatierra, Tijuana, Villahermosa y Zacatepec.	Reunión Nacional de Seguimiento Curricular de los Programas en Competencias Profesionales de las Carreras de Ingeniería en Sistemas Computacionales, Ingeniería Informática e Ingeniería en Tecnologías de la Información y Comunicaciones.
Instituto Tecnológico de Toluca, del 10 al 13 de febrero de 2014.	Representantes de los Institutos Tecnológicos de: Cerro Azul, Colima, Lerdo, Toluca y Veracruz.	Reunión de Seguimiento Curricular de los Programas Educativos de Ingenierías, Licenciaturas y Asignaturas Comunes del SNIT.

#### 4. Competencia(s) a desarrollar

Competencia(s) específica(s) de la asignatura
<ul style="list-style-type: none"> <li>Conoce y aplica modelos y/o técnicas de desarrollo de software con la finalidad de implementar sistemas eficientes en base a requerimientos específicos bajo lineamientos y estándares para el aseguramiento de calidad.</li> </ul>

#### 5. Competencias previas

<ul style="list-style-type: none"> <li>Construye diagramas de tiempo para determinar ruta(s) crítica(s) en la administración de proyectos.</li> </ul>
---

#### 6. Temario

No.	Temas	Subtemas
1	Introducción a la ingeniería del software y sistemas de información	1.1 Concepto de Ingeniería de Software y Sistema de Información. 1.2 Historia de la Ingeniería de Software. 1.3 Importancia de la Ingeniería de Software 1.4 Los sistemas de información: concepto, características, estructuras, procesos, clasificación, ERP's, CRM, SCM.
2	Modelos de la Ingeniería de Software	2.1. Modelos de la Ingeniería de Software: modelo de cascada, modelo de prototipos, modelo de espiral, RUP 2.2. Tendencias modernas de modelos de la Ingeniería de Software
3	Planificación del Proyecto	3.1. Aplicación de herramientas para estimación de tiempos y costos de desarrollo de software: GANTT, PERT/CPM, uso de software para la estimación de tiempos y costos.

		<p>3.2. <b>Ámbito del software:</b> recursos humanos, recursos de software reutilizables, recursos del entorno.</p> <p>3.3. Métricas orientadas al tamaño, al esfuerzo y a los costos.</p> <p>3.4. <b>Análisis y gestión del riesgo:</b> estrategias, identificación, proyección, refinamiento, reducción, supervisión y gestión del riesgo.</p>
4	Análisis y Modelado del proyecto de software	<p>4.1. Técnicas de recopilación de Información.</p> <p>4.2. Estudio de viabilidad.</p> <p>4.3. Análisis de requerimientos funcionales y no funcionales.</p> <p>4.4. Arquitectura del sistema basada en UML: Diagramas de Comportamiento y de funcionalidad.</p>
5	Calidad de Software	<p>5.1. Conceptos e importancia de Calidad y Calidad de Software.</p> <p>5.2. La calidad y la Globalización.</p> <p>5.3. Modelos de Calidad: ISO, SPICE, CMMI, Bootstrap, Moprosoft</p> <p>5.4. Métricas orientadas a la calidad.</p> <p>5.5. Aseguramiento de la calidad (SQA)</p> <p>5.6. Derecho informático aplicado al software: piratería y falsificación, autoría y creación, contratos y licencias.</p>

## 7. Actividades de aprendizaje de los temas

1. Introducción a la Ingeniería de Software y Sistemas de Información	
Competencias	Actividades de aprendizaje
<p><b>Específica(s):</b> Identifica los fundamentos teóricos que integran la ingeniería de software y los sistemas de información con la finalidad de comprender la relación entre sus elementos.</p> <p><b>Genéricas:</b></p> <ul style="list-style-type: none"> <li>• Habilidad para buscar y analizar información proveniente de fuentes diversas.</li> <li>• Capacidad de análisis y síntesis.</li> <li>• Habilidades básicas de manejo de la computadora.</li> </ul>	<ul style="list-style-type: none"> <li>• Realizar una investigación y síntesis sobre las características y elementos y de la ingeniería de software.</li> <li>• Investigar en diferentes fuentes de información la importancia e historia de la ingeniería del software y plasmarlo en una línea de tiempo.</li> <li>• Formar mesas de discusión de los temas investigados con la finalidad de enriquecer el conocimiento.</li> <li>• Realizar una investigación y síntesis sobre el concepto, las características, y elementos de los sistemas de información.</li> <li>• Discutir en plenarios la importancia que tiene los sistemas de información para la toma de decisiones.</li> <li>• Visitar alguna organización donde investigue los roles de los diferentes actores en los sistemas de información.</li> </ul>

	<ul style="list-style-type: none"> <li>• Identificar si existen procesos de Ingeniería de Software aplicados en el desarrollo de dichos sistemas</li> <li>• Discutir en clase los resultados de su investigación.</li> </ul>
<b>2. Modelos de la Ingeniería de Software</b>	
<b>Competencias</b>	<b>Actividades de aprendizaje</b>
<p><b>Específica(s):</b> Identifica los diferentes modelos de desarrollo de software para identificar el más adecuado y diseñar sistemas de información eficientes.</p> <p><b>Genéricas:</b></p> <ul style="list-style-type: none"> <li>• Capacidad de análisis y síntesis</li> <li>• Habilidades básicas de manejo de la computadora.</li> <li>• Solución de problemas.</li> <li>• Capacidad de aplicar conocimientos en la práctica.</li> </ul>	<ul style="list-style-type: none"> <li>• Investigar en diferentes fuentes los conceptos y características de cada modelo de desarrollo de software.</li> <li>• Discutir, en equipo, las características que diferencian a cada uno de los modelos de desarrollo de software.</li> <li>• Exponer en clase por equipos las características, actividades, ventajas, desventajas, recomendaciones de uso de un modelo de desarrollo de software.</li> <li>• Realizar un cuadro comparativo sobre los distintos modelos de desarrollo de software.</li> <li>• Determinar el modelo más adecuado para un tipo de sistema basado en propuestas de casos prácticos.</li> </ul>
<b>3. Análisis del Proyecto de Software</b>	
<b>Competencias</b>	<b>Actividades de aprendizaje</b>
<p><b>Específica(s):</b> Planifica actividades con la finalidad de estimar tiempos y costos del proyecto de software. Aplica métricas orientadas al tamaño para medir y controlar los recursos necesarios para el desarrollo del proyecto de software.</p> <p><b>Genéricas:</b></p> <ul style="list-style-type: none"> <li>• Capacidad de análisis y síntesis.</li> <li>• Capacidad de organizar y planificar.</li> <li>• Solución de problemas.</li> <li>• Toma de decisiones.</li> <li>• Trabajo en equipo.</li> <li>• Capacidad de aplicar los conocimientos en la práctica.</li> <li>• Capacidad de generar nuevas ideas (creatividad).</li> </ul>	<ul style="list-style-type: none"> <li>• Analizar el contexto del caso práctico para determinar las actividades necesarias.</li> <li>• Planificar el tiempo requerido para las actividades involucradas en el desarrollo del proyecto.</li> <li>• Aplicación de métricas orientadas al tamaño como COCOMO I ó II y Puntos de Función para determinar esfuerzo, recursos humanos y costos del software.</li> <li>• Realizar el estudio del costo beneficio para evaluar la viabilidad del proyecto desde el punto de vista económico.</li> <li>• Determinar la viabilidad técnica y operativa.</li> <li>• Realizar un análisis de los riesgos y determinar medidas preventivas y correctivas</li> </ul>
<b>4. Análisis y Modelado del Proyecto de Software</b>	
<b>Competencias</b>	<b>Actividades de aprendizaje</b>
<p><b>Específica(s):</b> Utiliza técnicas de análisis y modelado para construir la arquitectura del sistema</p>	<ul style="list-style-type: none"> <li>• Investigar y aplicar diversas técnicas de recopilación de información como cuestionarios o entrevistas.</li> </ul>



<p>mediante el uso de herramientas informáticas específicas.                  Genéricas:</p> <ul style="list-style-type: none"> <li>• Habilidades básicas de manejo de la computadora.</li> <li>• Solución de problemas.</li> <li>• Trabajo en equipo.</li> <li>• Capacidad de aplicar los conocimientos en la práctica.</li> <li>• Capacidad de generar nuevas ideas(creatividad).</li> </ul>	<ul style="list-style-type: none"> <li>• En base a la información recopilada realizar un análisis de viabilidad.</li> <li>• Desarrollar los requerimientos funcionales y no funcionales.</li> <li>• En base a los requerimientos obtenidos redactarlos desde el enfoque del sistema y el enfoque de usuario.</li> <li>• Elegir las métricas a aplicar en el proyecto planteado.</li> <li>• Desarrollar diagramas de casos de uso.</li> <li>• Desarrollar diagramas de actividades.</li> <li>• Desarrollar diagramas de clases.</li> <li>• Desarrollar diagramas de secuencia.</li> <li>• Desarrollar diagramas de colaboración.</li> <li>• Desarrollar diagramas de componentes.</li> <li>• Desarrollar diagramas de despliegue.</li> <li>• Presentar una propuesta con el caso práctico las fases que integran el análisis (recopilación de información, estudio de viabilidad y modelado del sistema).</li> </ul>
<p>5. Calidad de Software</p>	
<p>Competencias</p>	<p>Actividades de aprendizaje</p>
<p>Específica(s):                  Analiza y comprende el enfoque de calidad aplicada al software para determinar los procesos de implementación y aseguramiento de la misma en un entorno globalizado.                  Genéricas:</p> <ul style="list-style-type: none"> <li>• Capacidad de análisis y síntesis.</li> <li>• Comunicación oral y escrita.</li> <li>• Habilidad para buscar y analizar información proveniente de fuentes diversas.</li> <li>• Solución de problemas.</li> <li>• Capacidad de aplicar los conocimientos en la práctica.</li> </ul>	<ul style="list-style-type: none"> <li>• Realizar un análisis comparativo de las definiciones de calidad en distintas fuentes de información y elaborar mediante una lluvia de ideas una definición propia.</li> <li>• Investigar en diferentes fuentes de información, la definición y propósito del SQA y elaborar un reporte.</li> <li>• Debatir sobre la importancia de la calidad en el software.</li> <li>• Investigar en distintas fuentes las características y elementos de los distintos modelos de calidad de software.</li> <li>• Exponer en equipos las temáticas sobre los modelos de calidad de software.</li> <li>• Realizar un cuadro comparativo sobre los distintos modelos de calidad.</li> <li>• Investigar en distintas fuentes las características de las distintas métricas de calidad de software y exponer en equipos.</li> <li>• Realizar un cuadro comparativo sobre las distintas métricas de calidad.</li> </ul>

	<ul style="list-style-type: none"> <li>• Investigar sobre diferentes empresas que han implementado en sus procesos de desarrollo de software estándares de calidad.</li> <li>• Realizar ejercicios donde aplique las actividades del SQA aplicando los métodos y herramientas vistas en clase.</li> <li>• Elegir el modelo de calidad y las métricas a aplicar en el proyecto planteado.</li> </ul>
--	---

## 8. Práctica(s)

<ul style="list-style-type: none"> <li>• Utilizar alguna herramienta de software para administración de proyectos y así realizar el estudio de factibilidad de un caso práctico.</li> <li>• Utilizar técnicas de recolección de requerimientos para identificar y clasificar requerimientos de sistemas.</li> <li>• Utilizar alguna herramienta de software para el análisis de requerimientos y complementar el análisis y documentación del sistema.</li> <li>• Utilizar alguna herramienta de software para realizar el prototipo del sistema.</li> <li>• Medir los requerimientos del proyecto de acuerdo con las diferentes métricas de evaluación del análisis.</li> <li>• Diseñar un sistema, identificando y aplicando la metodología a la cual se pueda adaptar.</li> <li>• Transcribir dicho sistema a un lenguaje de modelado.</li> </ul>
--

## 9. Proyecto de asignatura

<p>El objetivo del proyecto que planteé el docente que imparta esta asignatura, es demostrar el desarrollo y alcance de la(s) competencia(s) de la asignatura, considerando las siguientes fases:</p> <ul style="list-style-type: none"> <li>• <b>Fundamentación:</b> marco referencial (teórico, conceptual, contextual, legal) en el cual se fundamenta el proyecto de acuerdo con un diagnóstico realizado, mismo que permite a los estudiantes lograr la comprensión de la realidad o situación objeto de estudio para definir un proceso de intervención o hacer el diseño de un modelo.</li> <li>• <b>Planeación:</b> con base en el diagnóstico en esta fase se realiza el diseño del proyecto por parte de los estudiantes con asesoría del docente; implica planificar un proceso: de intervención empresarial, social o comunitario, el diseño de un modelo, entre otros, según el tipo de proyecto, las actividades a realizar los recursos requeridos y el cronograma de trabajo.</li> <li>• <b>Ejecución:</b> consiste en el desarrollo de la planeación del proyecto realizada por parte de los estudiantes con asesoría del docente, es decir en la intervención (social, empresarial), o construcción del modelo propuesto según el tipo de proyecto, es la fase de mayor duración que implica el desempeño de las competencias genéricas y específicas a desarrollar.</li> <li>• <b>Evaluación:</b> es la fase final que aplica un juicio de valor en el contexto laboral-profesión, social e investigativo, ésta se debe realizar a través del reconocimiento de logros y aspectos a mejorar se estará promoviendo el concepto de “evaluación para la mejora continua”, la metacognición, el desarrollo del pensamiento crítico y reflexivo en los estudiantes.</li> </ul>
--

## 10. Evaluación por competencias

<ul style="list-style-type: none"> <li>• Utilizar alguna herramienta de software para administración de proyectos y así realizar el estudio de factibilidad de un caso práctico.</li> </ul>
---

- Utilizar técnicas de recolección de requerimientos para identificar y clasificar requerimientos de sistemas.
- Utilizar alguna herramienta de software para el análisis de requerimientos y complementar el análisis y documentación del sistema.
- Utilizar alguna herramienta de software para realizar el prototipo del sistema.
- Medir los requerimientos del proyecto de acuerdo con las diferentes métricas de evaluación del análisis.
- Diseñar un sistema, identificando y aplicando la metodología a la cual se pueda adaptar.
- Transcribir dicho sistema a un lenguaje de modelado.

### 11. Fuentes de información

- Pressman, R. S. (2010). Ingeniería de Software, Un enfoque Práctico (7ª. Edición). México: McGraw Hill. 7
- Somerville, I. (2011). Ingeniería de Software (9ª. Edición). Madrid: Pearson.
- Howard, M. Puntos críticos sobre seguridad de software. McGraw-Hill interamericana. España. 2006.
- Laudon, K.C. Sistemas de Información Gerencial. Pearson Educación. México, 2004.
- Minguet M. J. M. La calidad del software y su medida. Editorial CERASA. Madrid, España. 2003.
- Weitzenfeld, A. Ingeniería de software orientada a objetos. Cengage learning editores. México. 2005.