

## 1.- DATOS DE LA ASIGNATURA

Nombre de la asignatura:	<b>Gestión de Proyectos de Software</b>
Carrera:	<b>Ingeniería en Sistemas Computacionales</b>
Clave de la asignatura:	<b>SCG-1009</b>
(Créditos) SATCA <sup>1</sup>	<b>3-3-6</b>

## 2.- PRESENTACIÓN

### **Caracterización de la asignatura.**

La asignatura de Gestión de proyectos de software, proporciona al alumno los conceptos que requiere y que debe contemplar para la gestión o administración de un proyecto de software. Por otro lado, le da la posibilidad de poner en práctica dicha gestión, ya que se sugiere que en esta asignatura, el alumno desarrolle un proyecto de gestión de software para una empresa real, adquiriendo las competencias necesarias para estar al frente de dichos proyectos.

La aportación que esta asignatura le da al perfil profesional es la siguiente:

- La capacidad de poder desarrollar, implementar y administrar software de sistemas o de aplicación que cumpla con los estándares de calidad con el fin de apoyar la productividad y competitividad de las organizaciones.
- La capacidad de coordinar y participar en proyectos interdisciplinarios, ya que en los diferentes proyectos en los que participará el alumno en el mundo laboral, hará trabajo en equipo con profesionales de diferentes áreas, así como personas involucradas en el proyecto en cuestión.
- Integrar soluciones computacionales con diferentes tecnologías, plataformas o dispositivos.

La asignatura de gestión de proyectos se relaciona con asignaturas previas como ingeniería de software, taller de administración, administración de bases de datos.

---

<sup>1</sup> Sistema de asignación y transferencia de créditos académicos

## **Intención didáctica.**

Los contenidos de la asignatura de gestión de proyectos de software deben ser abordados de tal manera que cada unidad este dividida en 2 partes, la primera parte de la unidad será para que el profesor le presente al alumno los conceptos que la conforman, y la segunda parte deberá abarcar el diseño de la práctica a realizar. Por la naturaleza de los temas, el alumno inicia el proyecto desde la segunda unidad, al cual se le dará seguimiento a lo largo de la asignatura.

En el primer bloque, se podrá conocer los conceptos básicos para la gestión de proyectos, así como sus fases. En el segundo bloque, se conocerán los factores que más afectan la calidad del SW, las normas, estándares y herramientas para mejorar la calidad del producto de SW a desarrollar.

En el tercer bloque el alumno podrá comenzar a planificar un proyectos que solucione una problemática real iniciando con la determinación de objetivos, el análisis costo beneficio, análisis y recuperación de riesgos, todo esto con la finalidad de determinar si el proyecto propuesto es viable o no.

Posteriormente, conocerá la forma en que se debe presentar una propuesta. En el quinto bloque se analizará la metodología de selección y evaluación del personal idóneo para integrar el grupo de trabajo. El sexto bloque, proporcionará al alumno los conocimientos de administración de recursos, tiempo y de cómo llevar a cabo el seguimiento del desarrollo del proyecto.

De tal manera que al finalizar la asignatura el alumno debe entregar toda la documentación y evidencias generadas referentes a la gestión del software sobre el que esta trabajando.

Se sugiere que el profesor, presente al alumno ejemplos reales de gestión de software para que el alumno tenga una visión clara de la actividad que debe de realizar.

El alumno debe:

- Identificar una problemática real en una empresa y presentarla ante el grupo para su análisis.

- Elaborar una propuesta de solución a la problemática detectada.
- Llevar a cabo la gestión de un proyecto de software elegido por los equipos de trabajo, para lo cual es necesario:
  - ◆ Integrar y justificar un equipo de desarrollo acorde a la metodología seleccionada para el desarrollo del proyecto de software.
  - ◆ Presentar durante el semestre avances
  - ◆ Exponer al final del semestre los resultados

En estas actividades las competencias genéricas que el alumno desarrolla en esta asignatura son:

- Aplicar las actividades que involucra cada una de las etapas del ciclo de vida de un proyecto de software
- Analizar, diseñar, desarrollar y probar un proyecto de software
- Aplicar los procesos de Ingeniería de software

### 3.- COMPETENCIAS A DESARROLLAR

<p><b>Competencias específicas:</b></p> <p>Gestionar proyectos de software aplicando los elementos, técnicas y herramientas, en apego a los compromisos de costos, tiempo y alcance.</p>	<p><b>Competencias Genéricas</b></p> <p><b>Competencias instrumentales:</b></p> <ul style="list-style-type: none"> <li>• Capacidad de análisis y síntesis</li> <li>• Capacidad de organizar y planificar</li> <li>• Conocimientos generales básicos</li> <li>• Conocimientos básicos de la carrera</li> <li>• Comunicación oral y escrita en su propia lengua</li> <li>• Conocimiento de una segunda lengua</li> <li>• Habilidades básicas de manejo de la computadora</li> <li>• Habilidades de gestión de información(habilidad para buscar y analizar información proveniente de fuentes diversas</li> <li>• Solución de problemas</li> <li>• Toma de decisiones.</li> </ul> <p><b>Competencias interpersonales</b></p> <ul style="list-style-type: none"> <li>• Capacidad crítica y autocrítica</li> <li>• Trabajo en equipo</li> </ul>
--	---

	<ul style="list-style-type: none"> <li>• Habilidades interpersonales</li> <li>• Capacidad de trabajar en equipo interdisciplinario</li> <li>• Capacidad de comunicarse con profesionales de otras áreas</li> <li>• Apreciación de la diversidad y multiculturalidad</li> <li>• Habilidad para trabajar en un ambiente laboral</li> <li>• Compromiso ético</li> </ul> <p><b>Competencias sistémicas</b></p> <ul style="list-style-type: none"> <li>• Capacidad de aplicar los conocimientos en la práctica</li> <li>• Habilidades de investigación</li> <li>• Capacidad de aprender</li> <li>• Capacidad de adaptarse a nuevas situaciones</li> <li>• Capacidad de generar nuevas ideas (creatividad)</li> <li>• Liderazgo</li> <li>• Conocimiento de culturas y costumbres de otros países</li> <li>• Habilidad para trabajar en forma autónoma</li> <li>• Capacidad para diseñar y gestionar proyectos</li> <li>• Iniciativa y espíritu emprendedor</li> <li>• Preocupación por la calidad</li> <li>• Búsqueda del logro</li> </ul>
--	--

#### 4.- HISTORIA DEL PROGRAMA

<b>Lugar y fecha de elaboración o revisión</b>	<b>Participantes</b>	<b>Observaciones (cambios y justificación)</b>
Instituto Tecnológico de Saltillo Fecha del 5 al 9 de Octubre de 2009	Representantes de los Institutos Tecnológicos de: I.T. de Ciudad Valles I.T. de Matamoros I.T. de Mérida I.T. de Orizaba I.T.S. del Sur del Estado de Yucatán.	Reunión nacional de Diseño e innovación curricular de la carrera de Ingeniería en sistemas computacionales
I.T. de Mérida	Representantes de las	Análisis, enriquecimiento y

I.T.S. de Coatzacoalcos I.T. de Ciudad Valles I.T. de Matamoros I.T. de Orizaba I.T. de Toluca I.T.S. del Sur del Estado de Yucatán. Fecha del 12 de Octubre/2009 Al 15 de Febrero/2010	Academias de los Institutos Tecnológicos.	elaboración del programa de estudio propuesto en la Reunión Nacional de Diseño Curricular de la carrera de Ingeniería en Sistemas Computacionales
Instituto Tecnológico Superior de Poza Rica, Veracruz. Fecha del 22 al 26 de Febrero/2010	Representantes de los Institutos Tecnológicos participantes: I.T. de Coatzacoalcos I.T. de Ciudad Valles I.T. de Matamoros I.T. de Mérida I.T. de Orizaba I.T.S. del Sur del Estado de Yucatán.	Reunión nacional de consolidación de la carrera de Ingeniería en Sistemas Computacionales.

## 5.- OBJETIVO(S) GENERAL(ES) DEL CURSO (competencias específicas a desarrollar en el curso)

Gestionar proyectos de software aplicando los elementos, técnicas y herramientas, en apego a los compromisos de costos, tiempo y alcance.

## 6.- COMPETENCIAS PREVIAS

- Aplicar las actividades que involucra cada una de las etapas del ciclo de vida de un proyecto de software
- Analizar, diseñar, desarrollar y probar un proyecto de software
- Conocer los procesos de la Ingeniería de software
- Manejar alguna metodología para el desarrollo de software

## 7.- TEMARIO

Unidad	Temas	Subtemas
1	Introducción a la gestión de proyectos	1.1. Conceptos básicos para la gestión de proyectos 1.2. Fases de la gestión de proyectos

2	Calidad de Software	<ul style="list-style-type: none"> <li>1.2.1. Planificación de proyectos</li> <li>1.2.2. Propuesta</li> <li>1.2.3. Selección y Evaluación de personal</li> <li>1.2.4. Supervisión y Revisión del proyecto</li> <li>1.2.5. Informes</li> <li>1.3 Fundamentos de P.M.I.</li> <li>2.1 La gestión de proyectos usando un marco de calidad</li> <li>2.2 Estándares y Métricas de calidad en la ingeniería de SW <ul style="list-style-type: none"> <li>2.2.1 PSP y TSP</li> <li>2.2.2 CMM</li> <li>2.2.3 MOPROSOFT</li> </ul> </li> <li>2.3. Impacto de la calidad en tiempo, costo y alcance del proyecto</li> </ul>
3	Planificación del proyecto	<ul style="list-style-type: none"> <li>3.1 Objetivo del proyecto</li> <li>3.2 Estimaciones de tiempo</li> <li>3.3 Estimaciones de costos</li> <li>3.4 Estimación de personal requerido</li> <li>3.5 Análisis de riesgos <ul style="list-style-type: none"> <li>3.5.1 Tipos de riesgos</li> <li>3.5.2 Identificación, Impacto y proyección del riesgo</li> <li>3.5.3 Evaluación del riesgo</li> <li>3.5.4 Estrategias frente al riesgo</li> </ul> </li> <li>3.6 Análisis de la viabilidad del proyecto</li> </ul>
4	Presentación de la información	<ul style="list-style-type: none"> <li>4.1. Propuesta <ul style="list-style-type: none"> <li>4.1.1. Justificación del proyecto</li> <li>4.1.2. Calendario de actividades</li> <li>4.1.3. Personal involucrado</li> </ul> </li> </ul>

		4.1.4. Políticas de comunicación y seguimiento
		4.2. Lineamientos de comunicación y seguimiento
		4.2.1. Formatos
		4.2.2. Herramientas
		4.3. Contrato
5	Selección y Evaluación de personal	5.1. Roles y Actividades
		5.2. Carga de trabajo
		5.3. Asignación de tareas
		5.4. Herramientas para la evaluación de productividad
6	Supervisión y Revisión del proyecto	6.1. Administración de recursos
		6.2. Administración del tiempo
		6.3. Evaluación y ajustes del proyecto

## 8.- SUGERENCIAS DIDÁCTICAS (desarrollo de competencias genéricas)

El profesor debe:

Dominar la disciplina que está bajo su responsabilidad, conocer su origen y desarrollo histórico para abordar los temas. Deberá tener 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.

Además debe

- Armar equipos de trabajo para el desarrollo de un proyecto de software que el alumno elija.

- Trabajar en la definición de puestos dentro del equipo de trabajo, donde el alumno participe dinámicamente en el establecimiento de los perfiles.
- Exponer diferentes herramientas usadas en el mercado para la gestión de proyectos de software.

El alumno debe:

- Leer en al menos cinco fuentes los conceptos básicos de la gestión de proyectos
- Hacer fichas bibliográficas que contienen la información de las fuentes consultadas
- Hacer mapas conceptuales que representen los procesos de la gestión de proyectos de software.
- Crear la documentación necesaria que contenga la planificación de un proyecto de software.
- Usando componentes de las metodologías de investigación, definir el objetivo de un proyecto de software
- Estimar los tiempos, costos y personal involucrado en un proyecto de software que el alumno elija para aplicar los conocimientos al respecto.
- Leer artículos sobre el riesgo en los diferentes proyectos de diferentes áreas
- Investigar en diferentes fuentes acerca de temas de impacto y consecuencias de los riesgos en los proyectos de software
- Hacer un estudio de costo-beneficio de llevar a cabo un proyecto de software
- Exponer los resultados de un análisis de la viabilidad de un proyecto de software
- Crear una propuesta de desarrollo de software
- Identificar las actividades de un proyecto de desarrollo de software elegido por el equipo de trabajo del alumno
- Identificar los hitos del proyecto de software elegido
- Hacer un Calendario de las actividades del proyecto elegido.



- Hacer un documento que contenga las políticas de comunicación y seguimiento del proyecto entre los actores involucrados en el proyecto.
- Proponer formatos para comunicación y seguimiento de un proyecto de software
- Enumerar los componentes mínimos que conforman un contrato de software.
- Crear un contrato de software del proyecto elegido
- Usar una herramienta de productividad en algunos programas que conforman el proyecto elegido.
- Evaluar los resultados del uso de alguna herramienta de productividad y exponer resultados.
- Investigar acerca de los diferentes estándares de calidad para proyectos de software.
- Usar los componentes básicos del Project Management Institute.
- Exponer diferentes herramientas usadas en el mercado para la gestión de proyectos de software.

## **9.- SUGERENCIAS DE EVALUACIÓN**

La evaluación de la asignatura se hará con base en siguiente desempeño:

- Autoevaluación (Participación)
- Co-evaluación (Trabajo en equipo)
- Proyecto
- Reportes de prácticas
- Ejercicios realizados.
- Tareas
- Exposición
- Reportes de investigación
- Exámenes escritos

## **10.- UNIDADES DE APRENDIZAJE**

### **Unidad 1: Introducción a la gestión de proyectos**

<b>Competencia específica a desarrollar</b>	<b>Actividades de Aprendizaje</b>
Conocer los conceptos básicos de la gestión de proyectos así como identificar las fases que la conforman.	<ul style="list-style-type: none"> <li>• Leer en al menos cinco fuentes bibliográficas los conceptos básicos de la gestión de proyectos y hacer un resumen.</li> <li>• Hacer mapas conceptuales que representen los procesos de la gestión de proyectos de software.</li> <li>• Realizar un cuadro sinóptico que permita Identificar los componentes básicos del Project Management Institute.</li> </ul>

## Unidad 2: Calidad de Software

<b>Competencia específica a desarrollar</b>	<b>Actividades de Aprendizaje</b>
Identificar la importancia del uso y selección de estándares y métricas de calidad en un proyecto de software, así como el impacto de no apearse a ellos.	<ul style="list-style-type: none"> <li>• Investigar acerca de los diferentes estándares de calidad para proyectos de software y hacer un cuadro comparativo con los resultados.</li> <li>• Sugerir 3 medidas, 3 métricas y los indicadores que se podrían utilizar para evaluar la calidad de un proyecto de software.</li> <li>• Esquematizar mediante un mapa mental los enfoques PSP y TSP.</li> </ul>

## Unidad 3: Planificación del proyecto

<b>Competencia específica a desarrollar</b>	<b>Actividades de Aprendizaje</b>
Planificar un proyecto de software	<ul style="list-style-type: none"> <li>• Formar equipos de trabajo para seleccionar</li> </ul>

<p>desde la definición del objetivo, la estimación de tiempos, costos y personal requerido, identificando la existencia de riesgos y proponiendo acciones para reducir su impacto en el negocio, hasta el análisis de la viabilidad del mismo.</p>	<p>un proyecto de software.</p> <ul style="list-style-type: none"> <li>• Crear la documentación necesaria que contenga la planificación de un proyecto de software.</li> <li>• Usando componentes de las metodologías de investigación, definir el objetivo de un proyecto de software</li> <li>• Estimar los tiempos, costos y personal involucrado en un proyecto de software que el alumno elija para aplicar los conocimientos al respecto.</li> <li>• Trabajar en la definición de puestos dentro del equipo de trabajo, donde el alumno participe dinámicamente en el establecimiento de los perfiles.</li> <li>• Leer artículos sobre el riesgo en los diferentes proyectos de diferentes áreas</li> <li>• Investigar en diferentes fuentes acerca de temas de impacto y consecuencias de los riesgos en los proyectos de software</li> <li>• Hacer un estudio de costo-beneficio de llevar a cabo el proyecto de software</li> <li>• Exponer los resultados de un análisis de la viabilidad de un proyecto de software</li> </ul>
--	---

#### Unidad 4: Presentación de la información

<b>Competencia específica a desarrollar</b>	<b>Actividades de Aprendizaje</b>
<p>Integrar una propuesta y un contrato para el desarrollo de un proyecto de software. Aplicar</p>	<ul style="list-style-type: none"> <li>• Documentar y Presentar la propuesta del proyecto de desarrollo de software cumpliendo con los lineamientos</li> </ul>

políticas de comunicación y seguimiento del proyecto.	<p>establecidos</p> <ul style="list-style-type: none"> <li>• Utilizar los lineamientos de MOPROFOST que apliquen a los temas de la unidad</li> <li>• Utilizar una herramienta CASE para realizar la calendarización del proyecto.</li> </ul>
---	--

### Unidad 5: Selección y Evaluación de personal

Competencia específica a desarrollar	Actividades de Aprendizaje
Establecer los diferentes roles del personal involucrado en un proyecto de software, Aplicando herramientas para evaluación de productividad de un equipo de trabajo.	<ul style="list-style-type: none"> <li>• Asignar las tareas de acuerdo a los roles establecidos al equipo de desarrollo</li> <li>• Hacer un documento que contenga las políticas de comunicación y seguimiento del proyecto entre los actores involucrados en el proyecto.</li> <li>• Utilizar PSP para evaluar la productividad del equipo de trabajo en el desarrollo del proyecto de software.</li> </ul>

### Unidad 6: Supervisión y Revisión del proyecto

Competencia específica a desarrollar	Actividades de Aprendizaje
Aplicar herramientas para la administración del proyecto de software.	<ul style="list-style-type: none"> <li>• Investigar las diferentes herramientas que se utilizan para la administración y seguimiento de un proyecto, documentar ventajas, desventajas y costos.</li> <li>• Aplicar una herramienta para la administración y seguimiento del proyecto.</li> </ul>

## 11.- FUENTES DE INFORMACIÓN

1. Braude, E. Ingeniería de Software una perspectiva orientada a objetos, Alfaomega, México, 2003.
2. Piattini M.G. Calidad de Sistemas Informáticos. Alfaomega, México, 2007
3. Pressman, R. S. Ingeniería del Software un enfoque práctico. México. MC Graw-Hill. Madrid, España. 2008.
4. Watt S. H. PSP A Self Improvement process for software. Addison Wesley. USA, 2005.
5. NYCE. MoProsoft. [www.moprosoft.com.mx](http://www.moprosoft.com.mx)

## **12.- PRÁCTICAS PROPUESTAS**

1. Identificar una problemática real en una empresa y presentarla ante el grupo para su análisis.
2. Elaborar una propuesta de solución a la problemática detectada.
3. Llevar a cabo la gestión de un proyecto de software elegido por los equipos de trabajo, para lo cual es necesario:
  - 3.1. Documentando adecuadamente cada fase
  - 3.2. Integrar y justificar un equipo de desarrollo acorde a la metodología seleccionada para el desarrollo del proyecto de software.
  - 3.3. Presentar durante el semestre avances
  - 3.4. Exponer al final del semestre los resultados
4. Visitar empresas dedicadas al desarrollo de software e identificar de que manera llevan a cabo la gestión de sus proyectos
5. Realizar una investigación en Internet con respecto a las diferentes herramientas automatizadas que existen para cada una de las fases de la gestión de proyectos, cuales son las organizaciones que las utilizan, cuales se utilizan mas y cuanto cuesta dicha herramienta.