

## 1. Datos Generales de la asignatura

<b>Nombre de la asignatura:</b>	Proceso Personal para el Desarrollo de Software.
<b>Clave de la asignatura:</b>	ISH-1303
<b>Créditos (Ht-Hp_ créditos):</b>	1 - 3 - 4
<b>Carrera:</b>	Ingeniería en Sistemas Computacionales

## 2. Presentación

### Caracterización de la asignatura

Esta asignatura esta basada en una metodología proveniente del Instituto de Ingeniería del Software (SEI). PSP es una alternativa dirigida a los ingenieros en sistemas, que les permitirá mejorar la forma en la que construyen software. Considerando aspectos como la planeación, calidad, estimación de costos y productividad, se aplica para aumentar la calidad de los productos de software que se desarrollan.

La metodología PSP se relaciona con las materias precedentes como Fundamentos de Ingeniería de Software e Ingeniería de Software, y posteriores como Modelo de Desarrollo Integral.

Requiere de competencias previas como: Manejo del modelado orientado a objetos, un lenguaje para realizar el modelado, conocer y manejar los conceptos de estructuras de datos, dominio en el uso de herramientas CASE, dominio de algún lenguaje de programación orientado a objetos, identificación de las etapas del ciclo de desarrollo de sistemas.

### Intención didáctica

La asignatura debe ser teórico- práctico para que desarrolle en el alumno la habilidad de prevenir errores al estimar los costos del software y obtener productos de calidad.

En el bloque uno, identifica el proceso de TSP y PSP.

En el bloque dos, implementa los métodos de estimación populares, estimación de costos y estimación de tamaño del software.

En el bloque tres, aplica los principios básicos del método de estimación PROBE.  
 En el bloque cuatro, implementa los principios básicos de planeación de tiempo y calendario.  
 En el bloque cinco, integra el enfoque de calidad de PSP, los costos de la calidad, el proceso de comparación y estrategias de prevención y remoción.  
 En el bloque seis, integra los principios de diseño de software, revisiones y verificación.

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

Lugar y fecha de elaboración o revisión	Participantes	Observaciones
<p>Dirección General de Institutos Descentralizados, México D.F.</p> <p>Fecha: 12 al 14 de Septiembre de 2012.</p>	<p>Representantes:</p> <p>Tecnológico de Estudios Superiores de Coacalco.</p> <p>Instituto Tecnológico Superior de Atlixco, Comalcalco, Fresnillo, Santiago Papasquiaro, Tepexi de Rodríguez, Zapopan.</p>	<p>Análisis y adecuación por competencias del módulo de la especialidad "Ingeniería de Software" de la carrera de Ingeniería en Sistemas Computacionales.</p>

### 4. Competencias a desarrollar

Competencia general de la asignatura
Implementar una metodología bajo estándares para generar una disciplina de trabajo personal.
Competencias específicas
<ul style="list-style-type: none"> <li>Implementar una metodología bajo estándares.</li> <li>Generar una disciplina de trabajo personal.</li> </ul>
Competencias genéricas
<p><b>1. Competencias Instrumentales:</b></p> <ul style="list-style-type: none"> <li>Capacidad de abstracción análisis y síntesis.</li> <li>Capacidad de aplicar los conocimientos en la práctica.</li> </ul>

- Capacidad de investigación.
- Capacidad de aprender y actualizarse permanentemente.
- Capacidad para trabajar en equipo.

## **2. Competencias Interpersonales:**

- Capacidad crítica y autocrítica.
- Trabajo en equipo.
- Habilidades interpersonales.
- Capacidad de trabajar en equipo interdisciplinario.
- Capacidad de comunicarse con profesionales de otras áreas.
- Habilidad para trabajar en un ambiente laboral.
- Compromiso ético.

## **3. Competencias Sistémicas:**

- Capacidad de aplicar los conocimientos en la práctica.
- Habilidades de investigación.
- Capacidad de aprender.
- Capacidad de generar nuevas ideas (creatividad).
- Capacidad de diseñar y gestionar proyectos.
- Preocupación por la calidad.
- Búsqueda del logro.

## **5. Competencias previas de otras asignaturas**

<b>Competencias previas</b>
Conocimiento en el desarrollo de software. Conocimiento del proceso de la ingeniería de requerimientos

## **6. Temario**

<b>Temas</b>		<b>Subtemas</b>
<b>No.</b>	<b>Nombre</b>	
1.	Introducción a TSP y PSP	1.1 Introducción a PSP y TSP. 1.1.1 Principios del PSP. 1.1.2 Flujo de Proceso PSP. 1.1.3 Proceso Personal de Software. 1.1.4 TSP. 1.1.5 CMM y PSP.

		1.1.6 Aprendiendo PSP.
2.	Planeación, cálculo y estimación de tamaño	2.1 Planeación de Proyectos. 2.2 Medición de tamaño. 2.2.1 Conteo de programas. 2.2.2 Estándar de conteo. 2.3 Estándar de Codificación. 2.4 Antecedentes de estimación. 2.5 Principios de estimación. 2.6 Métodos de estimación populares. 2.6.1 El Método FuzzyLogic. 2.6.2 Método de estimación por puntos de función. 2.6.3 Método del componente estándar. 2.6.4 Método Delphi. 2.6.5 Método de puntos de casos de uso. 2.6.6 Estimación basada en Proxies.
3.	El método de estimación PROBE	3.1 El método de estimación PROBE. 3.3 Estimación del tamaño del programa. 3.4 Cálculo del intervalo de predicción. 3.3.1 Distribución normal con rangos de tamaño. 3.3.2 Distribución Log-Normal. 3.5 Método A. Regresión con LOC de objeto estimadas. 3.6 Método B. Regresión con LOC nuevas y cambiadas estimadas. 3.7 Método C. Método de promedios. 3.8 Método D: A juicio del Ingeniero. 3.9 Tutorial: Utilizando PROBE con el StudentWorkbook .
4.	Planeación de tiempo y calendario	4.1 Necesidad de planes de tiempo y calendario. 4.2 El proceso de planeación de tiempo. 4.3 Estimando el calendario. 4.4 Valor ganado. 4.5 Seguimiento del proyecto. 4.6 Calculando la terminación del trabajo. 4.7 Cambios al plan.
5.	Calidad y Diseño de software	5.1 El enfoque de calidad de PSP. 5.2 El costo de la calidad. 5.3 La estrategia de calidad. 5.4 Proceso de comparación. 5.5 Estrategias de remoción de defectos. 5.6 Estrategias de prevención de defectos. 5.7 El proceso de diseño. 5.8 Niveles de diseño . 5.9 Métodos y notaciones de diseño. 5.10 El proceso de diseño de PSP. 5.10.1 Plantilla de escenario operacional. 5.10.2 Plantilla de especificación funcional. 5.10.3 Plantilla de especificación de

		estados. 5.10.4 Plantilla de especificación lógica. 5.11 Jerarquía del diseño. 5.12 Usando UML.
6.	Revisiones de diseño y código, y verificación de diseño	6.1 Inspecciones, guías y revisiones. 6.1.1 Estrategia de revisión de PSP. 6.1.2 Control de proceso. 6.1.3 Listas de verificación. 6.2 Revisiones de diseño y código. 6.3 Métodos para evaluar y mejorar la calidad de las revisiones. 6.4 Necesidades y beneficios de las revisiones de diseño. 6.5 Tópicos de verificación de diseño.

## 7. Actividades de aprendizaje

Competencia específica y genéricas (a desarrollar y fortalecer por tema)	
<b>Competencia Específica</b> Identifica los principios de TSP y PSP.  <b>Competencias genéricas:</b> <ul style="list-style-type: none"> <li>• Capacidad de aplicar los conocimientos en la práctica.</li> <li>• Capacidad de investigación.</li> <li>• Trabajo en equipo.</li> <li>• Habilidades interpersonales.</li> <li>• Capacidad de aplicar los conocimientos en la práctica.</li> <li>• Habilidades de investigación.</li> <li>• Capacidad de aprender.</li> <li>• Capacidad de generar nuevas ideas (creatividad).</li> <li>• Capacidad de diseñar y gestionar proyectos.</li> </ul>	
Tema	Actividades de aprendizaje
<b>1.- Introducción a TSP y PSP.</b>	<ul style="list-style-type: none"> <li>• Investigación los conceptos generales de TSP y PSP.</li> <li>• Desarrollar un cuadro sinóptico de las características de TSP y PSP.</li> </ul>
Competencia específica y genéricas (a desarrollar y fortalecer por tema)	
<b>Competencia Específica</b> Analiza los modelos de estimación.	

**Competencias genéricas:**

- Capacidad de aplicar los conocimientos en la práctica.
- Capacidad para trabajar en equipo.
- Trabajo en equipo.
- Habilidades interpersonales.
- Capacidad de comunicarse con profesionales de otras áreas.
- Habilidad para trabajar en un ambiente laboral.
- Capacidad de aplicar los conocimientos en la práctica.
- Habilidades de investigación.
- Capacidad de aprender.
- Capacidad de generar nuevas ideas (creatividad).

Tema	Actividades de aprendizaje
<b>2.- Planeación, cálculo y estimación de tamaño.</b>	<ul style="list-style-type: none"><li>• Desarrollar un cuadro comparativo sobre los modelos de estimación.</li><li>• Aplicar modelo de estimación a un caso práctico.</li></ul>

Competencia específica y genéricas (a desarrollar y fortalecer por tema)

**Competencia Específica**

Aplica el método de estimación PROBE

**Competencias genéricas:**

- Trabajo en equipo.
- Habilidades interpersonales.
- Capacidad de trabajar en equipo interdisciplinario.
- Capacidad de comunicarse con profesionales de otras áreas.
- Capacidad de aplicar los conocimientos en la práctica.
- Habilidades de investigación.

Tema	Actividades de aprendizaje
<b>3.- El método de estimación PROBE.</b>	<ul style="list-style-type: none"><li>• Exposición de las fases del método de estimación PROBE.</li><li>• Aplicar modelo de estimación PROBE a un caso práctico.</li></ul>

Competencia específica y genéricas (a desarrollar y fortalecer por tema)

**Competencia Especifica**

Desarrolla plan de trabajo.

**Competencias genéricas:**

- Capacidad de abstracción análisis y síntesis.
- Capacidad de aplicar los conocimientos en la práctica.
- Capacidad de investigación.
- Habilidades interpersonales.
- Capacidad de trabajar en equipo interdisciplinario.
- Capacidad de comunicarse con profesionales de otras áreas.
- Habilidad para trabajar en un ambiente laboral.
- Capacidad de aplicar los conocimientos en la práctica.
- Habilidades de investigación.
- Capacidad de aprender.
- Capacidad de generar nuevas ideas (creatividad).
- Capacidad de diseñar y gestionar proyectos.

Tema	Actividades de aprendizaje
<b>4.-. Planeación de tiempo y calendario</b>	<ul style="list-style-type: none"><li>• Investigar técnicas de planeación de proyectos.</li><li>• Formular un plan de trabajo donde se estimen tiempos y recursos de un proyecto.</li></ul>
Competencia específica y genéricas (a desarrollar y fortalecer por tema)	

**Competencia Especifica**

Aplica las vistas de diseño de PSP.

**Competencias genéricas:**

- Capacidad de abstracción análisis y síntesis.
- Capacidad de aplicar los conocimientos en la práctica.
- Capacidad de investigación.
- Habilidades interpersonales.
- Capacidad de trabajar en equipo interdisciplinario.
- Capacidad de comunicarse con profesionales de otras áreas.
- Habilidad para trabajar en un ambiente laboral.
- Capacidad de aplicar los conocimientos en la práctica.
- Habilidades de investigación.
- Capacidad de aprender.
- Capacidad de generar nuevas ideas (creatividad).
- Capacidad de diseñar y gestionar proyectos.
- Búsqueda del logro.

Tema	Actividades de aprendizaje
<b>5.- Planeación de tiempo y calendario.</b>	<ul style="list-style-type: none"> <li>• Investigar y debatir los escenarios de PSP.</li> <li>• Aplicar métricas de calidad al proceso de diseño.</li> <li>• Investigar estrategias de prevención y remoción de defectos.</li> </ul>
Competencia específica y genéricas (a desarrollar y fortalecer por tema)	
<b>Competencia Específica</b>  Aplica técnicas de revisión a código y diseño.  <b>Competencias genéricas:</b> <ul style="list-style-type: none"> <li>• Capacidad de abstracción análisis y síntesis.</li> <li>• Capacidad de aplicar los conocimientos en la práctica.</li> <li>• Capacidad de investigación.</li> <li>• Habilidades interpersonales.</li> <li>• Capacidad de trabajar en equipo interdisciplinario.</li> <li>• Capacidad de comunicarse con profesionales de otras áreas.</li> <li>• Habilidad para trabajar en un ambiente laboral.</li> <li>• Capacidad de aplicar los conocimientos en la práctica.</li> <li>• Habilidades de investigación.</li> <li>• Capacidad de aprender.</li> <li>• Capacidad de generar nuevas ideas (creatividad).</li> <li>• Capacidad de diseñar y gestionar proyectos.</li> </ul>	
Tema	Actividades de aprendizaje
<b>6.- Revisiones de diseño y código, y verificación de diseño.</b>	<ul style="list-style-type: none"> <li>• Investigar las diferentes técnicas de revisión de código y diseño.</li> <li>• Implementar técnicas de revisión al código y diseño.</li> </ul>

## 8. Prácticas (para fortalecer las competencias de los temas y de la asignatura)

<ul style="list-style-type: none"> <li>• Implementar la metodología PSP en un proyecto de desarrollo de software.</li> </ul>
--



- Elaborar ejercicios de técnicas de estimación de software.
- Recabar métricas y aplicar estadísticas basadas en formulas de calidad de PSP.
- Recabar métricas y aplicar el método de estimación PROBE.

### **9. Proyecto integrador (Para fortalecer las competencias de la asignatura con otras asignaturas)**

Seleccionar algún proyecto de software, como objeto de estudio, donde el alumno implemente la metodología PSP, que genere técnicas de estimación de software, aplicando estadísticas basadas en fórmulas de calidad y métricas de estimación PROBE.

### **10.. Evaluación por competencias (específicas y genéricas de la asignatura)**

Se recomienda evaluar los siguientes puntos:

- Evaluación Teórica
- Prácticas de Laboratorio
- Evaluación de proyectos

La ponderación de la evaluación debe ser propuesta por la academia correspondiente de cada institución.

### **11. Fuentes de información (actualizadas considerando los lineamientos de la APA\*)**

1. Humphrey, Watts S., A DISCIPLINE FOR SOFTWARE ENGINEERING, ADDISON-WESLEY. 2005.

2. Humphrey, Watts S., TEAM SOFTWARE PROCESS INTRODUCTION, ADDISON-WESLEY.

3. Humphrey, Watts S., INTRODUCTION TO PSP, ADDISON-WESLEY.

\* American Psychological Association (*APA*)