



1. Datos Generales de la asignatura

Nombre de la asignatura:	Proceso Personal para el Desarrollo de Software.
Clave de la asignatura:	ISH-2103
SATCA¹:	1 - 3 - 4
Carrera:	Ingeniería en Sistemas Computacionales

2. Presentación

Caracterización de la asignatura
<p>Se encuentra ubicada en el octavo semestre de la especialidad en ingeniería de software, junto con las asignaturas de: Programación de Móviles; Ingeniería de Requerimientos; Arquitectura y Diseño de Software; Modelo de Desarrollo Integral (CMMI) y Verificación y Validación de software.</p> <p>Esta asignatura está 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.</p> <p>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.</p> <p>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.</p>
Intención didáctica
<p>El papel que el profesor debe desempeñar en la impartición de esta materia es de guía o facilitador del aprendizaje, ubicándose en este nuevo rol en el grupo. Considerando que trabaja con seres humanos, que tienen aspiraciones, sentimientos y que además tienen talentos que esperan ser desarrollados para adquirir los conocimientos, habilidades y actitudes en su formación profesional.</p> <p>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.</p> <p>En la unidad uno se busca que el estudiante identifique el proceso de TSP y PSP.</p> <p>En la unidad dos, implementa los métodos de estimación populares, estimación de costos y estimación de tamaño del software.</p> <p>En la unidad tres se aplican los principios básicos del método de estimación PROBE.</p>

¹ Sistema de Asignación y Transferencia de Créditos Académicos



En la unidad cuatro se implementan los principios básicos de planeación de tiempo y calendario.

En la unidad cinco se integra el enfoque de calidad de PSP, los costos de la calidad, el proceso de comparación y estrategias de prevención y remoción.

En la unidad seis se integra los principios de diseño de software, revisiones y verificación. El rol que estudiante debe llevar a cabo para desarrollar conocimientos, habilidades y actitudes son: comprometerse, dentro del grupo, para el logro de un aprendizaje efectivo; demostrar apertura para aprender de los demás y para compartir los conocimientos; buscar, con responsabilidad, la información que se considere necesaria; aprovechando los recursos disponibles en línea evitando el plagio; trabajar, en forma colaborativa, practicando las habilidades de comunicación y aplicar las habilidades de análisis y de síntesis de la información, con visión crítica.

3. Participantes en el diseño y seguimiento curricular del programa

Lugar y fecha de elaboración o revisión	Participantes	Observaciones
Tecnológico Nacional de México, Campus Comalcalco Tabasco, julio 2021.	Tecnológico Nacional de México, Campus Comalcalco Tabasco	Reunión de Academia de Sistemas Computacionales, según consta minuta con fecha 13 de julio de 2021.

4. Competencia(s) a desarrollar

Competencia(s) específica(s) de la asignatura
Implementa una metodología bajo estándares para generar una disciplina de trabajo personal para el desarrollo del software.

5. Competencias previas

Competencias específicas:

- Realiza el análisis de un proyecto de software a partir de la identificación del modelo de negocios de la organización que permita alcanzar estándares y métricas de calidad.
- Desarrolla soluciones de software, considerando la metodología y herramientas para la elaboración de un proyecto aplicativo en diferentes escenarios.

Competencias genéricas:

- Habilidades de gestión de información (habilidad para buscar y analizar información proveniente de fuentes diversas).
- Capacidad de análisis y síntesis.
- Capacidad de comunicación oral y escrita.
- Capacidad de aplicar los conocimientos en la práctica.

Habilidades en el uso de las tecnologías de la información y de la comunicación.



6. Temario

No	Temas	Subtemas
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.



		<p>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.</p>
6	Revisiones de diseño y código, y verificación de diseño.	<p>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.</p>

7. Actividades de aprendizaje de los temas

1. Introducción a TSP y PSP.	
Competencias	Actividades de aprendizaje
<p><i>Específica:</i> Identifica los principios de TSP y PSP para caracterizar ambos principios en un cuadro sinóptico.</p> <p><i>Genéricas:</i></p> <ul style="list-style-type: none"> ● Capacidad de abstracción análisis y síntesis. ● Habilidades de investigación ● Trabajo en equipo. ● Habilidades interpersonales. 	<ul style="list-style-type: none"> ● Organizar al grupo en equipos para realizar una investigación los conceptos generales de TSP y PSP y realizar una infografía de ambos conceptos. Subirla en formato pdf. ● Realizar una investigación documental de los siguiente subtemas: principios del PSP; flujo de Proceso PSP; proceso Personal de Software; TSP y CMM y PSP. Entregar un reporte de investigación en formato pdf. Evaluar y retroalimentar el reporte de investigación por parte del facilitador y los compañeros en una coevaluación

	<ul style="list-style-type: none"> • Desarrollar un cuadro sinóptico de las características de TSP y PSP. Evaluar el cuadro sinóptico y reflexionar en el foro sobre lo aprendido. • Retroalimentar los temas tratados durante la plenaria por parte de los estudiantes y el facilitador.
2. Planeación, cálculo y estimación de tamaño.	
Competencias	Actividades de aprendizaje
<p><i>Específica:</i> Analiza los modelos de estimación.</p> <p><i>Genéricas:</i></p> <ul style="list-style-type: none"> • Capacidad de aprender y actualizarse permanentemente. • Capacidad para trabajar en equipo. • Capacidad de comunicarse con profesionales de otras áreas. • Capacidad de aplicar los conocimientos en la práctica. • Habilidades de investigación. • Capacidad de aprender. 	<ul style="list-style-type: none"> • Organizar al grupo en equipos para desarrollar un cuadro comparativo sobre los modelos de estimación y presentarlo en plenaria al grupo. • Evaluar el cuadro sinóptico en la plataforma sugerir y retroalimentar para la mejora del producto de aprendizaje. • Aplicar modelo de estimación a un caso práctico. • Evaluar la práctica del modelo de estimación del caso práctico y retroalimentar para la mejora del producto de aprendizaje.
3. El método de estimación PROBE.	
Competencias	Actividades de aprendizaje
<p><i>Específica:</i> Aplica el método de estimación PROBE.</p> <p><i>Genéricas:</i></p> <ul style="list-style-type: none"> • Capacidad de aplicar los conocimientos en la práctica. • Capacidad de investigación. • Capacidad de aprender. • Capacidad de generar nuevas ideas (creatividad). 	<ul style="list-style-type: none"> • Organizar al grupo en equipos para la exposición de las fases del método de estimación PROBE. Subir la presentación en la plataforma antes de la presentación grupal para la retroalimentación. • Realizar una investigación documental de los siguiente subtemas: métodos de estimación populares; método FuzzyLogic; método de estimación por puntos de función; método del componente estándar; método Delphi; método de puntos de casos de uso; estimación basada en Proxies. E • Aplicar modelo de estimación PROBE a un caso práctico. • Evaluar la práctica del modelo de estimación del caso práctico y retroalimentar para la mejora del producto de aprendizaje.
4. Planeación de tiempo y calendario.	

Competencias	Actividades de aprendizaje
<p><i>Específica:</i> Desarrolla plan de trabajo para el desarrollo de software implementando la metodología PSP en equipo.</p> <p><i>Genéricas:</i></p> <ul style="list-style-type: none"> ● Capacidad de aprender y actualizarse permanentemente. ● Capacidad para trabajar en equipo. ● Capacidad de comunicarse con profesionales de otras áreas. ● Capacidad de aplicar los conocimientos en la práctica. ● Habilidades de investigación. ● Capacidad de aprender. 	<ul style="list-style-type: none"> ● Organizar al grupo en equipo para investigar técnicas de planeación de proyectos, entregar el reporte de investigación en tiempo y forma, subirlo a la plataforma en formato pdf para su retroalimentación y evaluación. ● Organizar al equipo para formular un plan de trabajo donde se estimen tiempos y recursos de un proyecto. ● Evaluar los avances del proyecto en la fase planeación y retroalimentar para la mejora del producto de aprendizaje.
5. Calidad y Diseño de software.	
Competencias	Actividades de aprendizaje
<p><i>Específica:</i> Aplica las vistas de diseño de PSP en el proyecto de desarrollo de software implementando la metodología PSP en equipo.</p> <p><i>Genéricas:</i></p> <ul style="list-style-type: none"> ● Capacidad de aprender y actualizarse permanentemente. ● Capacidad para trabajar en equipo. ● Capacidad de diseñar y gestionar proyectos. ● Preocupación por la calidad. 	<ul style="list-style-type: none"> ● Organizar al grupo en equipo para investigar y debatir los escenarios de PSP. Participar en el foro y retroalimentar a sus compañeros. ● Realizar prácticas para aplicar métricas de calidad al proceso de diseño. ● Avanzar en el proyecto e investigar estrategias de prevención y remoción de defectos. ● Evaluar los avances del proyecto en la siguiente fase y retroalimentar para la mejora del producto de aprendizaje.
6. Revisiones de diseño y código, y verificación de diseño.	
Competencias	Actividades de aprendizaje
<p><i>Específica:</i> Aplica técnicas de revisión a código y diseño del proyecto de desarrollo de software implementando la metodología PSP en equipo.</p> <p><i>Genéricas:</i></p> <ul style="list-style-type: none"> ● Capacidad de aprender y actualizarse permanentemente. ● Capacidad para trabajar en equipo. ● Capacidad de diseñar y gestionar proyectos. ● Preocupación por la calidad. 	<ul style="list-style-type: none"> ● Organizar al grupo en equipos con diferentes integrantes para investigar las diferentes técnicas de revisión de código y diseño. Realizar el reporte de investigación y subirlo en formato pdf. ● Evaluar el informe de investigación con la rúbrica y proporcionar retroalimentación. ● Presentar los avances del proyecto al implementar técnicas de revisión al código y diseño. ● Integrar todas las fases del proyecto para su evaluación final.



- Retroalimentación de todos los temas de la asignatura en plenaria. Evaluación sumativa.

8. Práctica(s)

- Implementar la metodología PSP en un proyecto de desarrollo de software.
- Elaborar ejercicios de técnicas de estimación de software.
- Recabar métricas y aplicar estadísticas basadas en fórmulas de calidad de PSP.
- Recabar métricas y aplicar el método de estimación PROBE.

9. Proyecto de asignatura

Proyecto: Desarrollo de software implementando la metodología PSP en equipo.

Objetivo: Desarrolla un proyecto aplicando los diferentes niveles de PSP bajo los estándares establecidos en el modelo del desarrollo de software que utilice el alumno.

Fundamentación: la aplicación de estándares de calidad en el desarrollo de software ofrece la ventaja de tener resultados con alto nivel de calidad, realizando diferentes actividades en la administración del proyecto hasta llegar a la comprensión de la optimización de los recursos disponibles y apegados a una necesidad específica aplicando los diferentes niveles de PSP.

Planeación: Seleccionar un software para desarrollar la aplicación en las fórmulas de optimización de código, detección de errores, reutilización de código y tiempo de desarrollo aplicando PSP1 y PSP2.

Ejecución: Seleccionar un sistema de información para aplicar los siguientes elementos de calidad de PSP1 y PSP2.

1. Locbase
2. Agregado
3. Código
4. Loc modificado
5. Loc suprimido
6. Nuevo y cambiante
7. Reutilizado
8. Loc total
9. Conteo del tamaño del programa

Evaluación: Presentación de resultados de defectos encontrados en el sistema de información y visión personal de la experiencia del proyecto desarrollado.



10. Evaluación por competencias

Evaluación formativa	
Producto o Evidencia de aprendizaje	Instrumento de evaluación
Rúbrica de mapa conceptual	Rúbrica de mapa conceptual: palabras clave, organización, jerarquía, enlaces.
Cuadro sinóptico de los principios de TSP y PSP	<p>Categorías del cuadro sinóptico de la lista de verificación.</p> <ul style="list-style-type: none"> – El cuadro sinóptico contiene tema o idea central, subtemas o ideas principales, secundarias e información esencial. – El cuadro sinóptico está ordenado, sin faltas de ortografía y con creatividad. – El contenido es claro y está bien distribuido. – Los subtemas están representados llaves. – No hay conceptos repetidos.
Presentación grupal: Niveles de PSP1 y PSP2.	<p>Rúbrica de presentación de temas relacionados a los niveles de PSP1 y PSP2, categorías:</p> <ul style="list-style-type: none"> ● Se identifican claramente las ideas primarias de las ideas secundarias. ● Los archivos Loc organizados. ● Las relaciones entre conceptos presentan jerarquías. ● No hay errores ortográficos en la presentación ● Se observa la preparación del tema, el uso de referencias empleadas y un orden de ideas. ● Se observa seguridad al tratar el tema, buen uso del recurso de apoyo, fluidez de ideas, tono de voz adecuado.
Infografía	<p>Evaluar la infografía respetando los siguientes indicadores de logro:</p> <p>El título es llamativo, está centrado en la lámina y está acompañado de una imagen principal.</p> <p>Formato de texto según las instrucciones dadas por la docente (tamaño de fuente, interlineado, sangría, alineación, entre otros).</p> <p>Utiliza imágenes relacionadas con el contenido.</p> <p>Hace uso de elementos llamativos que ayudan a reforzar la información (flechas, formas, figuras geométricas, entre otras)</p> <p>Organización coherente del contenido. (desde lo más complejo a lo más específico)</p> <p>Se evidencia originalidad en la elaboración de la infografía.</p> <p>Ordena la información de manera que sea comprensible y representativa del tema sugerido.</p>



<p>Resolución de problemas de un sistema de información desarrollado aplicando los niveles PSP1, PSP2 y PSP3.</p>	<p>Rúbrica de resolución de problemas:</p> <ul style="list-style-type: none"> • Analiza, reconoce e interpreta perfectamente los datos, identificando con certeza lo que se busca y demostrando una comprensión del problema. • Utiliza estrategias heurísticas efectivas y eficientes desarrollando los niveles PSP1 y PSP2 para la resolución del problema. • Realiza cálculos correctos empleando la fórmula de LOCBASE y LOCMODIFICADO. • Aporta la solución del problema, analiza y discute sobre su fiabilidad, si hay errores revisa el proceso y procede a modificarlo.
<p>Tabla comparativa</p>	<p>Tabla comparativa de defectos encontrados y efectos removidos:</p> <ul style="list-style-type: none"> • Identifica todos los elementos de comparación. • El organizador gráfico presenta los elementos centrales y sus relaciones en forma clara y precisa. • Se observan las diferencias de los códigos.
<p>Reporte de investigación</p>	<p>Indicadores de evaluación de un reporte de investigación:</p> <ul style="list-style-type: none"> – Tiene carátula con datos de la institución, nombre del alumno, materia, nombre del autor, título del libro y capítulo, nombre del docente, grupo y fecha. – El documento contiene índice en el que se han incluido todos los subtemas. – El contenido se encuentra en arial 12, justificados y espacio y medio; los subtítulos en arial 14, centrados y a espacio y medio. – El documento contiene introducción, apartados y conclusión. – El documento emplea citas bibliográficas y referencias ambas emplean las normativas de APA. – El documento se subió a la plataforma en tiene tiempo y formato pdf.

Evaluación sumativa:

Evaluación sumativa:

1. Participación en clases
2. Evidencias de productos y desempeños en la plataforma
3. Portafolio de aprendizaje personal
4. Evaluación del proyecto



11. Fuentes de información

- Bolaños, D., Sierra, A. y Alarcón, I. (2017). *Pruebas de software y JUnit. Un análisis en profundidad y ejemplos prácticos*. Pearson Education.
- Booch, G., Rumbaugh, J. y Jacobson, I. (2016). *El Lenguaje Unificado de Modelado. UML 2.0*. Editorial Addison-Wesley.
- Braude, E. (2017). *Ingeniería de Software. Una perspectiva orientada a objetos*. Alfaomega Grupo Editor S.A.
- Humphrey Watts, S. A. (2005). *Discipline for software engineering*. Editorial Addison-Wesley.
- Humphrey Watts, S. A. (2007). *Team software process introduction*. Editorial Addison-Wesley.
- Humphrey Watts, S. A. (2011). *Introduction to PSP*. Editorial Addison-Wesley.
- Larman, C. (2018). *UML y Patronos*. Prentice Hall.
- Meyer, B. (2017). *Construcción de software Orientado a Objetos*. Segunda Edición. Prentice Hall.
- Pressman, R. (2018). *Ingeniería del Software: Un Enfoque Práctico*. Ed. McGraw Hill.
- Rumbaugh J., Jacobson I. y Booch G. (2016). *El Lenguaje Unificado de Modelado. Manual de Referencia*. Editorial Addison-Wesley.
- Sánchez, S., Sicilia, M. y Rodriguez, D. (2018). *Ingeniería del software. Un enfoque desde la guía SWEBOK*. Alfaomega.
- Sommerville, I. (2017). *Ingeniería del Software*. Pearson 7a. Edición.