

1.- DATOS DE LA ASIGNATURA

Nombre de la asignatura:	Introducción a la Metodología CMMI-DEV
Carrera:	Ingeniería en Tecnologías de la Información y Comunicaciones
Clave de la asignatura:	SPD-1601
(Créditos) SATCA ¹	2-8-10

2.- PRESENTACIÓN

Caracterización de la asignatura.

Esta asignatura aporta al perfil del Ingeniero en Tecnologías de la Información y Comunicaciones las competencias necesarias para desarrollar soluciones de software empelando la metodología CMMI-DEV, en este curso el estudiante adquirirá las competencias esenciales para llevar a cabo las actividades de cada una de las fases del proceso así como elaborar los documentos requeridos en cada uno de ellos.

Intención didáctica.

El temario está organizado en 5 unidades:

Gestión de Requerimientos, en esta unidad se abordaran las especificaciones formales, las técnicas de identificación de requisitos, el establecimiento del alcance funcional del proyecto, el establecimiento formal de aceptación del usuario

Análisis Y Diseño, es esta unidad se estudian técnicas para diseñar prototipos; casos de uso, fichas técnicas y matrices de rastreabilidad para diseñar tanto bases de datos como modelos formatos de arquitectura tecnológica

Implementación, es esta unidad se trabaja con frameworks y plataformas de desarrollo, en los cuales se utilizan repositorios de código fuente y las versiones del código fuente

Pruebas, en esta unidad se diseña una matriz de pruebas y casos de pruebas, se aplican check list de pruebas unitarias y se realiza la gestión de defectos, inconformidades y evidencias

Puesta En Producción, en esta última unidad se elabora el manual de usuario, el manual técnico de la solución además de materiales formales para la capacitación del usuario y las carta de entrega y cierre de proyecto

¹ Sistema de asignación y transferencia de créditos académicos



3.- COMPETENCIAS A DESARROLLAR

Competencias específicas	Competencias genéricas
<p>Conocer cada una de las fases de la metodología CMMI-DEV y ponerlos en práctica así como elaborar los documentos requeridos de cada fase</p>	<p><u>Competencias instrumentales</u></p> <ul style="list-style-type: none">• Capacidad de análisis y síntesis.• Capacidad de organizar y planificar.• Conocimientos básicos de la carrera.• Comunicación oral y escrita.• Habilidades del manejo de la computadora.• Habilidad para buscar, analizar, clasificar y sintetizar información proveniente de fuentes diversas.• Solución de problemas.• Toma de decisiones. <p><u>Competencias interpersonales</u></p> <ul style="list-style-type: none">• Capacidad crítica y autocrítica• Trabajo en equipo• Habilidades interpersonales <p><u>Competencias sistémicas</u></p> <ul style="list-style-type: none">• Capacidad de aplicar los conocimientos en la práctica.• Habilidades de investigación.• Capacidad de aprender.• Capacidad de generar nuevas ideas (creatividad).• Habilidad para trabajar en forma autónoma.• Búsqueda del logro.

4.- HISTORIA DEL PROGRAMA

Lugar y fecha de elaboración o revisión	Participantes	Observaciones (cambios y justificación)
Instituto Tecnológico de Gustavo a Madero, agosto-septiembre de 2016.	M.C. Rivalcoba Rivas Jorge Iván. LI. Rodrigo Gris Suarez	

5.- OBJETIVO(S) GENERAL(ES) DEL CURSO

- Estudiar e implementar el modelo de desarrollo CMMI-DEV en los procesos de desarrollo de software, observando cada una de las mejores prácticas en la metodología.

6.- COMPETENCIAS PREVIAS

- Aplica tecnologías y herramientas actuales y emergentes de programación para desarrollar sistemas de información que ofrezcan soluciones a problemas del entorno.
- Desarrolla aplicaciones web dinámicas del lado cliente y del servidor, considerando la conectividad a orígenes de datos, la interconectividad entre aplicaciones y cómputo en la nube.

7.- TEMARIO

Unidad	Temas	Subtemas
1	GESTION DE REQUERIMIENTOS	<ol style="list-style-type: none"> 1. Elaborar Especificaciones Formales de Requerimientos de Software 2. Aplicar técnicas para la identificación de requerimientos, como pueden ser la Entrevista, Design Thinking, Mapas Mentales, Brainstorming, entre otras. 3. Establecer el alcance funcional de un proyecto conforme a las necesidades del usuario 4. Formalizar la validación y aceptación del usuario
2	ANALISIS Y DISEÑO	<ol style="list-style-type: none"> 1. Aplicar técnicas para diseñar prototipos 2. Especificar de Casos de Uso y fichas técnicas 3. Diseñar Matriz de Rastreabilidad 4. Diseñar y modelar bases de datos

		5. Diseñar modelos formales de arquitectura tecnológica
3	IMPLEMENTACION	<ol style="list-style-type: none"> 1. Usar diferentes frameworks y plataformas de desarrollo de software Web y móviles 2. Integrar línea base de código 3. Usar repositorios de código fuente 4. Versionar código fuente
4	PRUEBAS	<ol style="list-style-type: none"> 1. Diseñar matriz de pruebas y casos de pruebas 2. Aplicar Check List de Pruebas Unitarias 3. Gestionar defectos e inconformidades y evidencias
5	PUESTA EN PRODUCCION	<ol style="list-style-type: none"> 1. Elaborar Manual de Usuario 2. Elaborar manual técnico de la solución 3. Elaborar materiales formales para la capacitación del usuario 4. Elaborar Carta de Entrega 5. Elaborar Carta de Cierre de Proyecto

8.- SUGERENCIAS DIDÁCTICAS (desarrollo de competencias genéricas)

El docente debe:

Ser conocedor de la disciplina que está bajo su responsabilidad, conocer su origen y desarrollo histórico para considerar este conocimiento al abordar los temas. Desarrollar la capacidad para coordinar y trabajar en equipo; orientar el trabajo del estudiante y potenciar 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.

- Propiciar actividades de metacognición. Ante la ejecución de una actividad, señalar o identificar el tipo de proceso intelectual que se realizó: una identificación de patrones, un análisis, una síntesis, la creación de un heurístico, etc. Al principio lo hará el profesor, luego será el estudiante quien lo identifique.
- 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. Ejemplo: realizar prácticas en equipo que permitan obtener un resultado a partir del trabajo de todos.
- 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.
- Ejemplos: resolver un problema real aplicando a un negocio.

- Propiciar el desarrollo de capacidades intelectuales relacionadas con la lectura, la escritura y la expresión oral. Ejemplos: trabajar las actividades prácticas a través de guías escritas, redactar reportes e informes de las actividades de experimentación, exponer al grupo las conclusiones obtenidas durante las observaciones.
- Facilitar el contacto directo con problemas de su entorno para que plantee la solución mediante el modelado orientado a objetos y programe la solución utilizando el lenguaje de programación en ambiente Web.
- Propiciar el desarrollo de actividades intelectuales de inducción-deducción y análisis síntesis, que encaminen hacia una posición crítica del estudiante.
- 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.
- Relacionar los contenidos de la asignatura con el cuidado del medio ambiente; así como con las prácticas.
- Cuando los temas lo requieran, utilizar medios audiovisuales para una mejor comprensión del estudiante.
- Propiciar el uso de software libre.
- Propiciar el uso de ambientes GUI.

9.- SUGERENCIAS DE EVALUACIÓN

- Todas las actividades (sugeridas y propuestas por el docente) que se realizan en esta materia deben enfocarse a evaluar de manera permanente las competencias específica y, genéricas (instrumentales, interpersonales y sistémicas) que se proponen en este programa. Esto implica por parte del docente una planeación del curso detallada que motive al estudiante al desarrollo de las mismas. Por parte del alumno se requiere un compromiso y apertura al conocimiento y experiencias que sobre el tema se generen, así mismo se visualice el Desarrollo humano empresarial como una materia integradora y de áreas de oportunidades para su desarrollo personal y profesional.
- Se consideran los esquemas anteriores, la participación en el análisis de los temas a través de los foros de discusión, el manejo y aplicación de conceptos que realice el estudiante en las investigaciones encargadas, así como en la elaboración de una propuesta final del curso.
- En todo momento, es factible evaluar por escrito la interpretación de experiencias, apropiación de conocimientos y mejora del criterio, entre otros. Sin embargo, es recomendable contar con una ponderación de las competencias adquiridas, sobre todo en actividades como la discusión, análisis, exposición en público, capacidades de trabajo en equipo, entre otras actividades de aprendizaje

incluidas en la asignatura. Es decir, priorizar las actividades integrales más que exámenes escritos u orales y trabajos realizados por volumen.

- Pruebas objetivas de los temas vistos en clase: Prueba escrita o examen
- Método de casos: solución a una situación del área.
- Análisis de situaciones: Toma de decisiones y consecuencias
- Rúbricas de evaluación: Especificación de la matriz de calificación para los trabajos entregados.

10.- UNIDADES DE APRENDIZAJE

Unidad 1: GESTION DE REQUERIMIENTOS.

<i>Competencia específica a desarrollar</i>	<i>Actividades de Aprendizaje</i>
<p>Elaborar las especificaciones formales de los requerimientos de Software aplicando técnicas para la identificación de requerimientos, además de establecer el alcance funcional de un proyecto conforme a las necesidades del usuario y formalizar la validación y aceptación del usuario</p>	<p>Aplicación de técnicas para la identificación de requerimientos, como pueden ser la Entrevista, Design Thinking, Mapas Mentales, Brainstorming, entre otras. Entregables: Entrevista, Solicitud de cambios, Mapa Mental.</p> <p>Establecimiento del alcance funcional de un proyecto conforme a las necesidades del usuario. Entregables: Modelo de alcance propuesto, Especificación de requerimientos de software (entrega parcial)</p> <p>Validación, aceptación y formalización de alcance por parte del usuario. Entregables: Especificación de requerimientos de software (entrega final)</p> <p>Gestión del cambio de requerimientos. Entregables: Matriz de control de cambios</p>

Unidad 2: ANALISIS Y DISEÑO.

Competencia específica a desarrollar	Actividades de Aprendizaje
<p>Aplicar técnicas para diseñar prototipos, especificar de casos de uso y fichas técnicas.</p> <p>Diseñar la matriz de rastreabilidad así como diseñar bases de datos y modelos formales de arquitectura tecnológica</p>	<p>Aplicación de técnicas para diseñar prototipos. Entregables: Prototipo de GUI (mockup, sketch, etc.)</p> <p>Especificación de Casos de Uso y fichas técnicas. Entregables: Casos de uso, ficha técnica</p> <p>Diseño de Matriz de Rastreabilidad. Entregables: Matriz de rastreabilidad.</p> <p>Diseño y modelado bases de datos. Entregables: Modelo de base de datos: Entidad - Relación, diccionario de datos (todos los objetos)</p> <p>Diseño de modelos formales de arquitectura tecnológica. Entregables: Selección de arquitectura, Documento de arquitectura (con diagramas)</p>

Unidad 3: IMPLEMENTACION.

Competencia específica a desarrollar	Actividades de Aprendizaje
<p>Usar diferentes frameworks y plataformas de desarrollo de software Web y móviles e integrar línea base de código usando repositorios de código fuente y versiones de código fuente</p>	<p>Uso de diferentes frameworks y plataformas de desarrollo de software Web y móviles. Entregables: Pruebas de concepto de frameworks</p> <p>Integración de línea base de código. Entregables: Proyecto base con arquitectura propuesta, Prototipo.</p> <p>Uso de repositorios de código fuente Entregables: Repositorio de versionado.</p> <p>Versionado de código fuente Entregables: Estructura compuesta de línea de código: trunk, branches,</p>

	tags.
--	-------

Unidad 4: PRUEBAS.

Competencia específica a desarrollar	Actividades de Aprendizaje
Diseñar matriz la de pruebas y casos de pruebas, aplicar Check List de Pruebas Unitarias y gestionar defectos e inconformidades y evidencias	<p>Diseño de casos de prueba unitarios. Entregables: Código de casos de prueba unitarios automatizadas y evidencias de casos de prueba exitosos.</p> <p>Diseño de matriz de pruebas y casos de pruebas. Entregables: Matriz de pruebas funcionales y de integración, Evidencias de matrices de pruebas funcionales y de integración.</p> <p>Gestión de defectos e inconformidades y evidencias. Entregables: Reporte de defectos, Evidencia de incidencias</p>

Unidad 5: PUESTA EN PRODUCCION.

Competencia específica a desarrollar	Actividades de Aprendizaje
Elaborar tanto el manual de usuario, manual técnico así como materiales formales para la capacitación del usuario, la carta de entrega y la carta de cierre de proyecto	<p>Elaboración del plan de despliegue. Entregables: Plan de despliegue (Checklist de integración)</p> <p>Elaboración del Manual de Usuario. Entregables: Manual de usuario</p> <p>Elaboración del manual técnico de la solución. Entregables: Manual técnico</p> <p>Elaboración de materiales formales para la capacitación del usuario. Entregables: Plan de capacitación</p> <p>Elaboración de Carta de Entrega. Entregables: Carta de entrega</p> <p>Elaboración de Carta de Cierre de Proyecto. Entregables: Carta de cierre.</p>

11.- FUENTES DE INFORMACIÓN

- Mejora De Procesos Con Cmmi - Dev Pasta blanda - 2010 por VILLAGRA SERGIO (Autor),
o Tapa blanda.

- Editor: Nueva Libreria (2010).
- Idioma: Español.
- ISBN-10: 9871104898.
- ISBN-13: 978-9871104895
- CMMI: Improving Software and Systems Development Processes Using Capability Maturity Model Integration (CMMI-DEV),
 - Editor: Rocky Nook (1 de diciembre de 2008)
 - Idioma: Inglés
 - ISBN-10: 1933952288
 - ISBN-13: 978-1933952284