

TÉCNICO SUPERIOR UNIVERSITARIO EN TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN: ÁREA SISTEMAS INFORMÁTICOS

HOJA DE ASIGNATURA CON DESGLOSE DE UNIDADES TEMÁTICAS

1. Nombre de la asignatura	Ingeniería de software II
2. Competencias	Implementar sistemas de información de calidad, a través de técnicas avanzadas de desarrollo de software para eficientar los procesos de las organizaciones.
3. Cuatrimestre	Quinto
4. Horas Prácticas	64
5. Horas Teóricas	26
6. Horas Totales	90
7. Horas Totales por Semana Cuatrimestre	6
8. Objetivo de la Asignatura	El alumno desarrollará un sistema de información empleando las técnicas, metodologías y herramientas de diseño, pruebas y liberación necesarias para garantizar la implementación, de acuerdo a los requerimientos del cliente.

Unidades Temáticas	Horas		
	Prácticas	Teóricas	Totales
I. Diseño de interfaz de usuario	8	4	12
II. Documentación en la etapa de codificación	12	6	18
III. Pruebas de software	30	12	42
IV. Liberación de software	14	4	18
Totales	64	26	90

ELABORÓ: COMITÉ DE DIRECTORES DE LA CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

APROBÓ: C. G. U. T.

REVISÓ: COMISIÓN ACADÉMICA Y DE VINCULACIÓN DEL ÁREA

FECHA DE ENTRADA EN VIGOR: SEPTIEMBRE 2009

F-CAD-SPE-23-PE-XXX

INGENIERÍA DE SOFTWARE II

UNIDADES TEMÁTICAS

1. Unidad Temática	I. Diseño de interfaz de usuario
2. Horas Prácticas	8
3. Horas Teóricas	4
4. Horas Totales	12
5. Objetivo	El alumno diseñará un sistema de información aplicando los principios de usabilidad, estándares y guías de estilo para mejorar la facilidad de uso.

Temas	Saber	Saber hacer	Ser
Principios de Usabilidad	Identificar los principios de Usabilidad: - Interacción - Consistencia - Facilidad de Aprendizaje - Retroalimentación - Ayuda - Estandarización - Accesibilidad	Diseñar un sistema de información empleando los principios de usabilidad.	Analítico Sistemático Hábil para el Trabajo en Equipo Ordenado Ético Disciplinado Líder
Estándares y Guías de Estilo	Identificar los tipos de estándares: - iure - facto Identificar las guías de estilo.	Diseñar un sistema de información empleando los estándares de iure y facto, así como las guías de estilo.	Analítico Sistemático Hábil para el Trabajo en Equipo Ordenado Disciplinado Ética profesional Líder

ELABORÓ: COMITÉ DE DIRECTORES DE LA CARRERA DE
TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

APROBÓ: C. G. U. T.

REVISÓ: COMISIÓN ACADÉMICA Y DE VINCULACIÓN DEL ÁREA

FECHA DE ENTRADA EN VIGOR: SEPTIEMBRE 2009

F-CAD-SPE-23-PE-XXX

INGENIERÍA DE SOFTWARE II

Proceso de evaluación		
Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
Elaborará con base en un proyecto un documento que contenga: • Diseño de la interfaz de usuario aplicando principios de usabilidad, los estándares y las guías de estilo. • Justificación del uso de la guía de estilo.	1. Comprender los principios de usabilidad. 2. Identificar los estándares y guías de estilo. 3. Comprender los estándares y guías de estilo. 4. Diseñar las estrategias para aplicar los principios de usabilidad en el sistema de información.	Proyecto Lista de cotejo

ELABORÓ: COMITÉ DE DIRECTORES DE LA CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

APROBÓ: C. G. U. T.

REVISÓ: COMISIÓN ACADÉMICA Y DE VINCULACIÓN DEL ÁREA

FECHA DE ENTRADA EN VIGOR: SEPTIEMBRE 2009

F-CAD-SPE-23-PE-XXX

INGENIERÍA DE SOFTWARE II

Proceso enseñanza aprendizaje	
Métodos y técnicas de enseñanza	Medios y materiales didácticos
Aprendizaje basado en proyectos Ejercicios prácticos Equipos colaborativos	Pintarrón Computadora Cañón Software para diseño (Lenguaje de Programación, Visio)

Espacio Formativo		
Aula	Laboratorio / Taller	Empresa
	X	

ELABORÓ: COMITÉ DE DIRECTORES DE LA CARRERA DE
TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

APROBÓ: C. G. U. T.

REVISÓ: COMISIÓN ACADÉMICA Y DE VINCULACIÓN DEL ÁREA

FECHA DE ENTRADA EN VIGOR: SEPTIEMBRE 2009

INGENIERÍA DE SOFTWARE II

UNIDADES TEMÁTICAS

1. Unidad Temática	II. Documentación en la etapa de codificación
2. Horas Prácticas	12
3. Horas Teóricas	6
4. Horas Totales	18
5. Objetivo	El alumno elaborará los diagramas UML y la documentación para guiar la etapa de codificación en el desarrollo de software.

Temas	Saber	Saber hacer	Ser
Diagramas UML en la codificación	Identificar los elementos de los diagramas de componentes, implementación, despliegue y actividad.	Elaborar los diagramas de componentes, implementación, despliegue y actividad.	Analítico Sistemático Hábil para el Trabajo en Equipo Ordenado Disciplinado
Documentación en la codificación	Identificar las convenciones del lenguaje de programación para documentar el código tales como: - Nombrado de variables, métodos y constantes - Documentación interna (Identificación de la clase: Nombre, Descripción de la funcionalidad, Fecha de creación, versión). - Formato (Identación).	Elaborar el código empleando las convenciones de documentación propia del lenguaje.	Analítico Sistemático Hábil para el Trabajo en Equipo Ordenado Disciplinado

ELABORÓ: COMITÉ DE DIRECTORES DE LA CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

APROBÓ: C. G. U. T.

REVISÓ: COMISIÓN ACADÉMICA Y DE VINCULACIÓN DEL ÁREA

FECHA DE ENTRADA EN VIGOR: SEPTIEMBRE 2009

INGENIERÍA DE SOFTWARE II

Proceso de evaluación		
Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
Elaborará con base en un proyecto un documento que incluya: • Diagrama de Componentes • Diagrama de implementación • Diagrama de despliegue • Diagrama de actividad • Código documentado	1. Identificar los elementos de los diagramas que apoyan la etapa de codificación. 2. Comprender las convenciones en la documentación de código. 3. Analizar el uso de convenciones en la documentación de código.	Proyecto Lista de cotejo

ELABORÓ: COMITÉ DE DIRECTORES DE LA CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

APROBÓ: C. G. U. T.

REVISÓ: COMISIÓN ACADÉMICA Y DE VINCULACIÓN DEL ÁREA

FECHA DE ENTRADA EN VIGOR: SEPTIEMBRE 2009

INGENIERÍA DE SOFTWARE II

Proceso enseñanza aprendizaje	
Métodos y técnicas de enseñanza	Medios y materiales didácticos
Aprendizaje basado en proyectos Equipos colaborativos Práctica de laboratorio	Pintarrón Computadora Cañón Herramienta UML (StarUML, Rational Rose, Umlet, ArgoUML, Visual Paradigm)

Espacio Formativo		
Aula	Laboratorio / Taller	Empresa
	X	

ELABORÓ: COMITÉ DE DIRECTORES DE LA CARRERA DE
TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

APROBÓ: C. G. U. T.

REVISÓ: COMISIÓN ACADÉMICA Y DE VINCULACIÓN DEL ÁREA

FECHA DE ENTRADA EN VIGOR: SEPTIEMBRE 2009

INGENIERÍA DE SOFTWARE II

UNIDADES TEMÁTICAS

1. Unidad Temática	III. Pruebas de software
2. Horas Prácticas	30
3. Horas Teóricas	12
4. Horas Totales	42
5. Objetivo	El alumno realizará pruebas de software empleando metodologías y herramientas para detectar e interpretar errores.

Temas	Saber	Saber hacer	Ser
Tipos de pruebas de software	Identificar el concepto de prueba de software y el objetivo de las mismas.		Analítico Sistemático Hábil para el Trabajo en Equipo Ordenado Disciplinado
Casos de Prueba	Identificar el alcance del procedimiento de prueba. Identificar la estructura para documentar un caso de prueba.	Diseñar casos de prueba de software.	Analítico Sistemático Hábil para el Trabajo en Equipo Ordenado Disciplinado Creativo
Herramientas para pruebas de software	Identificar las herramientas para ejecutar o simular los casos de prueba.	Interpretar los resultados obtenidos de la ejecución de un caso de prueba utilizando herramientas para pruebas de software.	Analítico Sistemático Hábil para el Trabajo en Equipo Ordenado Disciplinado

ELABORÓ: COMITÉ DE DIRECTORES DE LA CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

APROBÓ: C. G. U. T.

REVISÓ: COMISIÓN ACADÉMICA Y DE VINCULACIÓN DEL ÁREA

FECHA DE ENTRADA EN VIGOR: SEPTIEMBRE 2009

INGENIERÍA DE SOFTWARE II

Proceso de evaluación		
Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
Elaborará con base en un proyecto un documento que incluya: • Tabla comparativa con descripción de los tipos de prueba. • Casos de prueba. • Interpretación de los resultados obtenidos al ejecutarlos.	1. Identificar el concepto y los tipos de prueba existentes. 2. Comprender la estructura de un caso de prueba. 3. Analizar las herramientas que permiten ejecutar casos de prueba. 4. Comprender el proceso para usar una herramienta que ejecute los casos de prueba. 5. Interpretar los resultados de la ejecución de un caso de prueba.	Proyecto Lista de cotejo

ELABORÓ: COMITÉ DE DIRECTORES DE LA CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

APROBÓ: C. G. U. T.

REVISÓ: COMISIÓN ACADÉMICA Y DE VINCULACIÓN DEL ÁREA

FECHA DE ENTRADA EN VIGOR: SEPTIEMBRE 2009

INGENIERÍA DE SOFTWARE II

Proceso enseñanza aprendizaje	
Métodos y técnicas de enseñanza	Medios y materiales didácticos
Aprendizaje basado en proyectos Equipos colaborativos Práctica de laboratorio	Pintarrón Computadora Cañón Herramienta para ejecutar pruebas de software

Espacio Formativo		
Aula	Laboratorio / Taller	Empresa
	X	

ELABORÓ: COMITÉ DE DIRECTORES DE LA CARRERA DE
TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

APROBÓ: C. G. U. T.

REVISÓ: COMISIÓN ACADÉMICA Y DE VINCULACIÓN DEL ÁREA

FECHA DE ENTRADA EN VIGOR: SEPTIEMBRE 2009

INGENIERÍA DE SOFTWARE II

UNIDADES TEMÁTICAS

1. Unidad Temática	IV. Liberación de software
2. Horas Prácticas	14
3. Horas Teóricas	4
4. Horas Totales	18
5. Objetivo	El alumno elaborará la documentación técnica y el paquete de instalación para la liberación del software.

Temas	Saber	Saber hacer	Ser
Manual Técnico.	Identificar los elementos de un manual técnico de un sistema de información.	Elaborar un manual técnico de un sistema de información.	Analítico Sistemático Hábil para el Trabajo en Equipo Ordenado Disciplinado
Proceso de Instalación.	Identificar el plan y mecanismo de instalación.de instalación. Identificar los elementos del manual de instalación de un sistema de Información.	Elaborar el paquete de instalación.	Analítico Sistemático Hábil para el Trabajo en Equipo Ordenado Disciplinado
Manual de Usuario.	Identificar estándares para la elaboración de un manual de usuario.	Elaborar un manual de usuario basado en un estándar.	Analítico Sistemático Hábil para el Trabajo en Equipo Ordenado Disciplinado

ELABORÓ: COMITÉ DE DIRECTORES DE LA CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

APROBÓ: C. G. U. T.

REVISÓ: COMISIÓN ACADÉMICA Y DE VINCULACIÓN DEL ÁREA

FECHA DE ENTRADA EN VIGOR: SEPTIEMBRE 2009

INGENIERÍA DE SOFTWARE II

Proceso de evaluación		
Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
Elaborará con base en un proyecto: • Manual de usuario • Manual técnico • Paquete de Instalación	1. Identificar la estructura de un manual técnico, de instalación y de usuario. 2. Comprender el proceso para generar un paquete de instalación. 3. Estructurar un paquete de instalación.	Proyecto Lista de cotejo

ELABORÓ: COMITÉ DE DIRECTORES DE LA CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

APROBÓ: C. G. U. T.

REVISÓ: COMISIÓN ACADÉMICA Y DE VINCULACIÓN DEL ÁREA

FECHA DE ENTRADA EN VIGOR: SEPTIEMBRE 2009

INGENIERÍA DE SOFTWARE II

Proceso enseñanza aprendizaje	
Métodos y técnicas de enseñanza	Medios y materiales didácticos
Aprendizaje basado en proyectos Equipos colaborativos Práctica de laboratorio	Pintarrón Computadora Cañón Herramientas de apoyo para la generación de discos de instalación

Espacio Formativo		
Aula	Laboratorio / Taller	Empresa
	X	

ELABORÓ: COMITÉ DE DIRECTORES DE LA CARRERA DE
TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

APROBÓ: C. G. U. T.

REVISÓ: COMISIÓN ACADÉMICA Y DE VINCULACIÓN DEL ÁREA

FECHA DE ENTRADA EN VIGOR: SEPTIEMBRE 2009

INGENIERÍA DE SOFTWARE II

CAPACIDADES DERIVADAS DE LAS COMPETENCIAS PROFESIONALES A LAS QUE CONTRIBUYE LA ASIGNATURA

Capacidad	Criterios de Desempeño
Codificar los módulos del sistema recopilando métricas y utilizando lenguajes de programación, para cumplir con su funcionalidad dentro del sistema.	a) Presenta el código fuente de los módulos del sistema: - basado en el diseño establecido. - acorde a los estándares del lenguaje programación elegido. - respetando las buenas prácticas de programación. b) Genera la documentación del código, especificando: - clases (fechas, autores). - métodos (argumentos, objetivo) y - variables (tipos de datos, usabilidad). c) Registra información de su propio proceso de desarrollo: - número de defectos inyectados y removidos. - líneas de código en tiempo determinado.
Integrar los módulos del sistema de acuerdo al diseño y la tecnología establecidos, para el óptimo funcionamiento de la aplicación.	a) Codifica la interface correspondiente que genere las ligas entre módulos, siguiendo el modelo de componentes generado en el diseño. b) Documenta la (s) interfaces generadas, especificando: - Objetivo - Relaciones - Parámetros - Fecha - Autor - entre otros

ELABORÓ: COMITÉ DE DIRECTORES DE LA CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

APROBÓ: C. G. U. T.

REVISÓ: COMISIÓN ACADÉMICA Y DE VINCULACIÓN DEL ÁREA

FECHA DE ENTRADA EN VIGOR: SEPTIEMBRE 2009

F-CAD-SPE-23-PE-XXX

Capacidad	Criterios de Desempeño
Elaborar manuales de usuario y técnico con base en la información generada en las etapas de desarrollo y considerando los estándares de calidad de la organización; para la correcta operación y mantenimiento del sistema.	a) Genera el manual técnico, integrando y organizando la documentación generada: - Requerimientos del sistema. - Análisis de complejidad. - Modelos y documentación de diseño. - Código fuente. - Pruebas. b) Elabora el manual de usuario utilizando los estándares establecidos, redactando el modo de operación del sistema y su alcance.
Diseñar sistemas de información con base a la arquitectura definida; para describir los componentes del sistema y la forma en que interactúan.	a) Genera los modelos usando lenguaje de modelado unificado, y de acuerdo a los requerimientos del sistema y la documentación respectiva.
Liberar el sistema de información acorde a un protocolo de liberación y al plan del proyecto, para su operación y puesta a punto.	a) Establece protocolo de liberación de acuerdo a las características del proyecto y documenta la liberación de la versión del proyecto. b) Genera el paquete de instalación (instaladores, respaldos, ayudas, entre otros.)

ELABORÓ: COMITÉ DE DIRECTORES DE LA CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

APROBÓ: C. G. U. T.

REVISÓ: COMISIÓN ACADÉMICA Y DE VINCULACIÓN DEL ÁREA

FECHA DE ENTRADA EN VIGOR: SEPTIEMBRE 2009

INGENIERÍA DE SOFTWARE II

FUENTES BIBLIOGRÁFICAS

Autor	Año	Título del Documento	Ciudad	País	Editorial
Committee of the IEEE Computer Society.	(1998)	<i>IEEE recommended practice for software requirements specifications (830-1998).</i>	Washington	EE.UU.	IEEE Computer Society
Nielsen, Jakob.	(1993)	<i>Usability Engineering.</i>	California	EE.UU.	Morgan Kaufmann. Academic Press
Pressman, Roger S.	(2008)	<i>Ingeniería de Software</i>	Madrid	España	McGrawHill
Rosson, Mary Beth.	(2002)	<i>Usability Engineering.</i>	Washington	EE.UU.	Academic Press.
S. Adler, Paul. A. Winograd, Terry.	(1992)	<i>Usability: Turning Technologies Into Tools.</i>	New York	EE.UU.	Oxford University Press
Sommerville, Ian.	(2005)	<i>Ingeniería de Software</i>	Madrid	España	Addison Wesley
Committee of the IEEE Computer Society	Enero, 1998.	<i>IEEE recommended practice for software requirements specifications (830-1998)</i>	Consultado: 25 de Junio 2009	http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?tp=&arnumber=720574&isnumber=15571	

ELABORÓ: COMITÉ DE DIRECTORES DE LA CARRERA DE
TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

APROBÓ: C. G. U. T.

REVISÓ: COMISIÓN ACADÉMICA Y DE VINCULACIÓN DEL ÁREA

FECHA DE ENTRADA EN VIGOR: SEPTIEMBRE 2009

F-CAD-SPE-23-PE-XXX