PROGRAMA SINÓPTICO POR COMPETENCIAS

I. DATOS DE IDENTIFICACIÓN

PROGRAMA ACADÉMICO:	INGENIERÍA EN Sistemas Computacionales		
NOMBRE:	Ingeniería de Software	CLAVE: SCD-1011	
TIPO DE CURSO:	Obligatorio/Opcional		
HORAS: (T.P.C.)	TEÓRICAS: 2	PRÁCTICAS: 3	CRÉDITOS ACADÉMICOS: 5
SEMESTRE:	Sexto (6º)		
FECHA DE ELABORACIÓN:	13 / 02 /2014		
ELABORADO POR:	SNIT		

II. COMPETENCIAS A DESARROLLAR:

Desarrolla soluciones de software, considerando la metodología y herramientas para la elaboración de un proyecto aplicativo en diferentes escenarios.

III. CONTENIDOS:

UNIDAD I: Análisis.	
<p>COMPETENCIA ESPECÍFICA DE LA UNIDAD:</p> <p>Específica(s):</p> <ul style="list-style-type: none"> • Abstrae información del usuario final para elaborar el análisis de requerimientos del software a desarrollar. <p>Genéricas:</p> <ul style="list-style-type: none"> • Trabajo en equipo • Capacidad de analizar información provenientes de diversas fuentes • Habilidad de comunicación oral y escrita. • Propuesta de solución de problemas • Toma de decisiones. • Habilidades interpersonales • Capacidad de aplicar los conocimientos en la práctica. 	<p>CONTENIDO:</p> <ul style="list-style-type: none"> 1.1 Revisión de especificación de requisitos. <ul style="list-style-type: none"> 1.1.1 Norma IEEE830 1.1.2 Trazabilidad de requisitos 1.2 Descripción de procesos actuales 1.3 Diagramas UML 1.4 Estudio de Factibilidad 1.5 Análisis Costo-Beneficio

UNIDAD II: Diseño.	
<p>COMPETENCIA ESPECÍFICA DE LA UNIDAD:</p> <p>Específica(s):</p> <ul style="list-style-type: none"> • Aplica modelos, técnicas y herramientas para la etapa de diseño del software. <p>Genéricas:</p> <ul style="list-style-type: none"> • Trabajo en equipo • Capacidad de analizar información provenientes de diversas fuentes • Habilidad de comunicación oral y escrita. • Propuesta de solución de problemas • Toma de decisiones. 	<p>CONTENIDO:</p> <ul style="list-style-type: none"> 2.1 Diseño de procesos propuestos <ul style="list-style-type: none"> 2.1.1 Herramientas CASE para diseño 2.2 Diseño arquitectónico 2.3 Diseño de datos 2.4 Diseño de interfaz de usuario

- Habilidades interpersonales
- Capacidad de aplicar los conocimientos en la práctica.

UNIDAD III: Desarrollo.

<p>COMPETENCIA ESPECÍFICA DE LA UNIDAD:</p> <p>Específica(s):</p> <ul style="list-style-type: none"> • Construye un software derivado de la problemática planteada en el análisis y diseño de un sistema. <p>Genéricas:</p> <ul style="list-style-type: none"> • Capacidad para usar nuevas tecnologías. • Trabajo en equipo • Habilidad de comunicación oral y escrita. • Propuesta de solución de problemas • Toma de decisiones. • Habilidades interpersonales • Capacidad de aplicar los conocimientos en la práctica. 	<p>CONTENIDO:</p> <ul style="list-style-type: none"> 3.1 Lenguajes para el desarrollo de software 3.2 Manejadores de bases de datos 3.3 Construcción del sistema
---	--

UNIDAD IV: Pruebas e Implantación.

<p>COMPETENCIA ESPECÍFICA DE LA UNIDAD:</p> <p>Específica(s):</p> <ul style="list-style-type: none"> • Evalúa el software construido a partir de las pruebas e implementación realizada. <p>Genéricas:</p> <ul style="list-style-type: none"> • Trabajo en equipo • Habilidad de comunicación oral y escrita. • Propuesta de solución de problemas • Toma de decisiones. • Habilidades interpersonales • Capacidad de aplicar los conocimientos en la práctica. 	<p>CONTENIDO:</p> <ul style="list-style-type: none"> 4.1 Diseño de caso de prueba 4.2 Pruebas de componentes 4.3 Pruebas del sistema 4.4 Documentación de resultados de las pruebas. 4.5 Entrega del sistema y Capacitación a usuarios 4.6 Entrega de documentación técnica y de usuario del sistema.
---	--

IV. FORMA DE EVALUACIÓN:

Para evaluar las actividades de aprendizaje se recomienda solicitar: resúmenes, reportes de prácticas, estudios de casos, exposiciones en clase, reportes de visitas, desarrollo de proyecto, portafolio de evidencias.

Para verificar el nivel del logro de las competencias del estudiante se recomienda utilizar: listas de cotejo, matrices de valoración, rúbricas, guías de observación, coevaluación y autoevaluación.

V. REFERENCIA BIBLIOGRÁFICA:

Impresas:

1. Booch G. (2006). *El lenguaje Unificado de Modelado*, UML 2.0, Guía de Usuario. 2ª. Edición. España: Pearson ADDISON-WESLEY.
2. Braude, E. (2003). *Ingeniería de Software una perspectiva orientada a objetos*. México: ALFAOMEGA.
3. Fontela, C. (2011) *UML Modelado de Software para Profesionales*. 1ª. Edición, Buenos Aires: Alfa-Omega Grupo Editor.
4. Fowler M. (2000). *UML Gota a Gota*. 1ª. Edición. México: PEARSON.
5. Hansen G.W., Hansen J.V. 2002 *Diseño y administración de Bases de datos*: Prentice Hall.
6. Kendall E. K. (2011). *Análisis y Diseño de sistemas*. 8ª. Edición. México: PEARSON..
7. Pressman, R.S. (2008). *Ingeniería del Software un enfoque práctico*. 6ª. Edición México: MC GRAW HILL.
8. Senn J.A. (1996). *Análisis y Diseño de Sistemas*. 2ª Edición. México: MC GRAW HILL.
9. Silberschatz, A.; Korth, H. F., Sudarshan, S. 2002 *Fundamentos de Bases de Datos*. 5ª ed. Ed. McGraw Hill.
10. Sommerville, Ian (2001). *Ingeniería de Software*. México: PRENTICE HALL.

Electrónicas:

11. Unam. (Sin fecha). Panorama de los diferentes modelos de ciclo de vida de los sistemas. Consultado en Octubre 2012. Disponible en:
http://uxmcc1.iimas.unam.mx/~cursos/Objetos/clases3_4.html.