

PROGRAMA SINÓPTICO POR COMPETENCIAS

I. DATOS DE IDENTIFICACIÓN

PROGRAMA ACADÉMICO:	INGENIERÍA EN Sistemas Computacionales		
NOMBRE:	Tópicos Avanzados de Programación		CLAVE:SCD-1027
TIPO DE CURSO:	Obligatorio/Opcional		
HORAS: (T.P.C.)	TEÓRICAS: 2	PRÁCTICAS: 3	CRÉDITOS ACADÉMICOS:5
SEMESTRE:	Cuarto (4º)		
FECHA DE ELABORACIÓN:	13 / 02 /2014		
ELABORADO POR:	SNIT		

II. COMPETENCIAS A DESARROLLAR:

Desarrolla soluciones de software para resolver problemas en diversos contextos utilizando programación concurrente, acceso a datos, que soporten interfaz gráfica de usuario y consideren dispositivos móviles.

III. CONTENIDOS:

UNIDAD I: Interfaz gráfica de usuario.	
COMPETENCIA ESPECÍFICA DE LA UNIDAD: Específica(s): • Desarrolla programas para interactuar con el usuario de una manera amigable, tilizando GUI (Interfaz Gráfica de Usuario) manipuladas a través de eventos. Genéricas: • Capacidad de análisis y síntesis. • Capacidad de organizar y planificar. • Comunicación oral y escrita. • habilidad para buscar y analizar información proveniente de fuentes diversas. • Solución de problemas. • Toma de decisiones. • Capacidad crítica y autocrítica. • Capacidad de trabajar en equipo. • Capacidad de comunicar sus ideas. • Capacidad de liderazgo. • Capacidad de aplicar los conocimientos en la práctica. • Habilidades de investigación. • Capacidad de aprender. • Capacidad de adaptarse a nuevas situaciones. • Capacidad de generar nuevas ideas (creatividad). • Habilidad para trabajar en forma autónoma. • Preocupación por la calidad. • Búsqueda del logro.	CONTENIDO: 1.1 Creación de interfaz gráfica para usuarios. 1.2 Tipos de eventos. 1.3 Manejo de eventos. 1.4 Manejo de componentes gráficos de control

UNIDAD II: Componentes y librerías.	
COMPETENCIA ESPECÍFICA DE LA UNIDAD: Específica(s): • Diseña e implementa componentes y librerías para lograr la reutilización de código. Genéricas: • Capacidad de análisis y síntesis. • Capacidad de organizar y planificar. • Comunicación oral y escrita. • habilidad para buscar y analizar información proveniente de fuentes diversas. • Solución de problemas. • Toma de decisiones. • Capacidad crítica y autocrítica. • Capacidad de trabajar en equipo. • Capacidad de comunicar sus ideas. • Capacidad de liderazgo. • Capacidad de aplicar los conocimientos en la práctica. • Habilidades de investigación. • Capacidad de aprender. • Capacidad de adaptarse a nuevas situaciones. • Capacidad de generar nuevas ideas (creatividad). • Habilidad para trabajar en forma autónoma. • Preocupación por la calidad. • Búsqueda del logro.	CONTENIDO: 2.1 Definición conceptual de componentes, paquetes / ibrerías. 2.2 Uso de librerías proporcionadas por el lenguaje. 2.3 Creación de componentes (visuales y no visuales) definidos por el usuario. 2.4 Creación y uso de paquetes/librerías definidas por el usuario.

UNIDAD III: Programación concurrente (MultiHilos).	
COMPETENCIA ESPECÍFICA DE LA UNIDAD: Específica(s): • Crea subprogramas para resolver problemas concurrentes utilizando Multihilos. Genéricas: • Capacidad de análisis y síntesis. • Capacidad de organizar y planificar. • Comunicación oral y escrita • habilidad para buscar y analizar información proveniente de fuentes diversas. • Solución de problemas. • Toma de decisiones. • Capacidad crítica y autocrítica. • Capacidad de trabajar en equipo. • Capacidad de comunicar sus ideas. • Capacidad de liderazgo. • Capacidad de aplicar los conocimientos en la	CONTENIDO: 3.1 Concepto de hilo. 3.2 Comparación de un programa de flujo único contra uno de flujo múltiple. 3.3 Creación y control de hilos. 3.4 Sincronización de hilos computación.

práctica. • Habilidades de investigación. • Capacidad de aprender. • Capacidad de adaptarse a nuevas situaciones. • Capacidad de generar nuevas ideas (creatividad). • Habilidad para trabajar en forma autónoma. • Preocupación por la calidad. • Búsqueda del logro.	
--	--

UNIDAD IV: Acceso a datos.	
COMPETENCIA ESPECÍFICA DE LA UNIDAD: Específica(s): • Establece conexiones a diferentes orígenes de datos para su manipulación y visualización de información. Genéricas: • Capacidad de análisis y síntesis. • Capacidad de organizar y planificar. • Comunicación oral y escrita. • habilidad para buscar y analizar información proveniente de fuentes diversas. • Solución de problemas. • Toma de decisiones. • Capacidad crítica y autocrítica. • Capacidad de trabajar en equipo. • Capacidad de comunicar sus ideas. • Capacidad de liderazgo • Capacidad de aplicar los conocimientos en la práctica. • Habilidades de investigación. • Capacidad de aprender. • Capacidad de adaptarse a nuevas situaciones. • Capacidad de generar nuevas ideas (creatividad). • Habilidad para trabajar en forma autónoma. • Preocupación por la calidad. • Búsqueda del logro	CONTENIDO: 4.1 Introducción. 4.2 Conexión a origen de datos. 4.3 Manipulación de datos. 4.4 Visualización de datos.

UNIDAD V: Programación de dispositivos móviles.
--

COMPETENCIA ESPECÍFICA DE LA UNIDAD: Específica(s): • Desarrollar aplicaciones básicas para dispositivos móviles, considerando su entorno operativo. Genéricas: • Capacidad de análisis y síntesis. • Capacidad de organizar y planificar. • Comunicación oral y escrita. • habilidad para buscar y analizar información proveniente de fuentes diversas. • Solución de problemas. • Toma de decisiones. • Capacidad crítica y autocrítica. • Capacidad de trabajar en equipo. • Capacidad de comunicar sus ideas. • Capacidad de liderazgo. • Capacidad de aplicar los conocimientos en la práctica. • Habilidades de investigación. • Capacidad de aprender. • Capacidad de adaptarse a nuevas situaciones. • Capacidad de generar nuevas ideas (creatividad). • Habilidad para trabajar en forma autónoma. • Preocupación por la calidad. • Búsqueda del logro.	CONTENIDO: 5.1. Introducción a las tecnologías y herramientas móviles. 5.2 Clasificación y aplicaciones de los dispositivos móviles. 5.3 Entorno operativo de las aplicaciones móviles. 5.4 Desarrollo de aplicaciones móviles. 5.5. Aspectos de seguridad.
---	--

IV. FORMA DE EVALUACIÓN:

La evaluación debe ser permanente y continua. Se debe hacer una evaluación diagnóstica, formativa y sumativa. Se debe aplicar autoevaluación, coevaluación y heteroevaluación.

Algunos de los instrumentos que se pueden utilizar:

- Mapa conceptual.
- Mapa mental.
- Guía de problemas de programación.
- Examen (teóricos y prácticos).
- Reportes de prácticas.
- Resúmenes.
- Cuadro sinóptico.
- Preguntas guiadas.
- Plenaria.
- Cuadro comparativo.

Herramientas:

- Rúbrica.
- Lista de cotejo.
- Matriz de valoración.
- Guía de observación.

Se debe generar un portafolio de evidencias, de preferencia en formato digital.

REFERENCIA BIBLIOGRÁFICA:

1. Aguilar, L. J. (2010). Programacion en c/c++ java y UML. México: McGraw Hill.
2. Bell, D. (2011). Java para estudiantes. México: Pearson.
3. Ceballos, F. J. (2010). JAVA 2: Curso de programación. Madrid: RA-MA.
4. Dean, J. (2009). Introducción a la programación con Java. México: McGraw Hill.
5. Deitel, D. y. (2010). Java Cómo Programar. México: Prentice Hall.
6. Friesen, J. (2011). Java para desarrollo android. España: Anaya Multimedia.
7. Huddleston, R. (2011). Android para todos. España: Anaya Multimedia.
8. Lauren Darcey, S. C. (2012). Android 4. Madrid: Anaya Multimedia.
9. Soriano, J. E. (2011). Android: Programación de dispositivos móviles a través de ejemplos. México: Marcombo, S.A.
10. Raynal, Michel. (2012). Concurrent Programming: Algorithms, Principles, and Foundations. Springer.