

	FORMATO PROPUESTA DE DESARROLLO PROGRAMA DE CURSO			
	CODIGO: FO-M-DC-05-01	VERSION: 2	FECHA: 2010-04-19	PAGINA: 1 de 2

1. IDENTIFICACIÓN				
Nombre de la Asignatura PROGRAMACIÓN ORIENTADA A EVENTOS			Código 72020402	Área Ingeniería aplicada
Naturaleza Teórico-Práctica	No de Créditos 3	TP Trabajo Presencial 3	TD Trabajo Dirigido 2	TI Trabajo Independiente 4
Semestre Cuarto	Duración 80 Horas	Habilitable NO	Homologable SI	Validable NO
PRE-REQUISITO: No aplica al plan de estudios				
2. JUSTIFICACIÓN				
<p>El desarrollo de habilidades para diseñar y construir soluciones de forma eficiente, ergonómica y parametrizables en el entorno de la producción de software de calidad. Por tal motivo un ingeniero de sistemas debe conocer los conceptos fundamentales vinculados en el diseño, organización de código, utilización de componentes del lenguaje y demás factores necesarios para presentar una solución a un problema atendiendo las interacciones del usuario en tiempo de ejecución, con altos estándares de calidad en su producto final.</p>				
3. COMPETENCIAS				
3.1 Competencias Generales				
El estudiante emplea los conceptos fundamentales de programación orientada a eventos para la construcción de aplicaciones que den solución a situaciones del contexto.				
3.2 Competencias Especificas				
<ul style="list-style-type: none"> • Comprende el concepto de POE y las aplica en el desarrollo de aplicaciones • Conoce los diferentes componentes gráficos y sus métodos • Estructura formularios utilizando los conceptos de eventos, validaciones y excepciones. • Implementa el manejo de hilos en las aplicaciones. • Aplica los conceptos de programación gráfica y movimiento para elaborar juegos sencillos. 				



FORMATO PROPUESTA DE DESARROLLO PROGRAMA DE CURSO

CODIGO: FO-M-DC-05-01

VERSION: 2

FECHA: 2010-04-19

PAGINA: 2 de 2

4. OBJETIVOS

- Aplicar el concepto de Programación Orientada a Objetos en el desarrollo de aplicaciones
- Conocer los componentes gráficos y sus métodos en la elaboración de aplicaciones
- Crear formularios aplicando los conceptos de eventos, validaciones y excepciones.
- Apropiar el manejo y sincronización de hilos en las aplicaciones
- Elaborar juegos sencillos aplicando movimiento en dos dimensiones, 2D

5. CONTENIDO TEMÁTICO Y ANÁLISIS DE CRÉDITOS

El curso se encuentra constituido por cuatro (4) capítulos:

UNIDAD 1. Programación Gráfica

Componentes visuales - AWT vs SWING
Layout Manager

UNIDAD 2. Validaciones y Excepciones

Manejo de Excepciones
Definición de Eventos y adaptadores
Componentes visuales - Eventos
Manejo de excepciones
Excepciones propias
Validaciones
Implementación formularios

UNIDAD 3. Hilos

Manejo de HILOS
Aplicaciones con Hilos y gráficos

UNIDAD 4. Movimiento 2D

Figuras y movimientos en 2 dimensiones

Análisis de Créditos

TEMAS	TRABAJO PRESENCIAL	TRABAJO DIRIGIDO	TRABAJO INDEPENDIENTE
Componentes visuales - AWT vs SWING	4	2	5
Layout Manager	4	2	5
Manejo de Excepciones	3	2	5
Definición de Eventos y adaptadores	3	2	5
Componentes visuales - Eventos	3	2	5
Manejo de excepciones	3	2	5

	FORMATO PROPUESTA DE DESARROLLO PROGRAMA DE CURSO			
	CODIGO: FO-M-DC-05-01	VERSION: 2	FECHA: 2010-04-19	PAGINA: 3 de 2

Excepciones propias	3	2	5
Validaciones	4	2	5
Implementación formularios	3	2	6
Manejo de HILOS	6	4	6
Aplicaciones con Hilos y graficos	6	4	6
Figuras y movimientos en 2 dimensiones	6	6	6
TOTAL DE HORAS DEL CURSO	48	32	64
TOTAL CRÉDITOS:	3		

6. Estrategias Metodológicas

La materia se desarrollará de forma teórico práctica. Se realizaran clases magistrales y actividades en las que el profesor presentara el tema relacionado y a partir del trabajo en clase los estudiantes expondrán sus interrogantes, los cuales les permitirán desarrollar discusiones en torno a los problemas y métodos abordados en la creación, verificación y evaluación de algoritmos. Los estudiantes a partir de su trabajo autónomo apoyarán las actividades del curso.

Trabajo presencial:

Este se desarrollará con la participación del docente por medio de clases magistrales exposiciones, charlas, talleres/guías, desarrollo de proyectos tutorías, etc. en las cuales se fundamentarán los conceptos teóricos del curso.

Trabajo dirigido:

Este momento de aprendizaje será empleado para el desarrollo de ejercicios prácticos que permitan generar en el estudiante competencias para la solución de problemas reales mediante el uso de algoritmos.

Trabajo independiente:

Se requiere que el estudiante resuelva y profundice los conceptos explorados en el curso, igualmente el trabajo independiente debe permitir generar un espacio en el cual el estudiante resuelve problemas reales de manera independiente permitiendo encontrar posibles dificultades que deberán ser resueltas en los momentos de trabajo dirigido y presencial.

7. RECURSOS

- Sala de cómputo.
- Lenguajes de Programación – JAVA – C++
- Video Beam, tablero digital
- Acceso a la plataforma Moodle donde se dispondrá el contenido del curso.

	FORMATO PROPUESTA DE DESARROLLO PROGRAMA DE CURSO			
	CODIGO: FO-M-DC-05-01	VERSION: 2	FECHA: 2010-04-19	PAGINA: 4 de 2

8. EVALUACIÓN

La valoración final del estudiante; estará comprendida entre cero (0) y cinco cero (5,0) siendo la aprobación igual o superior a tres cero (3,0). Se obtendrá mediante la suma de las evaluaciones parciales (70%) y finales (30%), de la siguiente manera:

70%:

La evaluación se sujeta al acuerdo pedagógico de la asignatura, con el fin de verificar los logros y competencias alcanzados a través de su estudio.

30%:

La evaluación se sujeta al acuerdo pedagógico de la asignatura, con el fin de verificar los logros y competencias alcanzados a través de su estudio.

9. BIBLIOGRAFÍA

- PRESSMAN, Roger. Ingeniería de Software. Un enfoque práctico. Prentice Hall. 1998.
- AHO, Alfred. ULLMAN, Jeffrey. Estructuras de datos y algoritmos. Addison Wesley. Mexico.1988.
- VILLALOBOS, Jorge. Diseño y manejo de estructuras de datos en C. McGraw Hill. Bogotá 1996.
- HELEMAN, Gregory. Estructuras de Datos, Algoritmos, y Programación Orientada a Objetos. McGraw Hill. 1998.
- NIKITIN, Eric. Into the realm of Oberon: An introduction to programming and the Oberon2 programming language. Springer Verlag. Enero 1998.