



Programa de asignatura por competencias de educación superior

Sección I. Identificación del Curso

Tabla 1. Identificación de la Planificación del Curso.

Actualización:	Junio 01, 2022				
Carrera:	Ingeniería en Desarrollo de Software	Asignatura:	Programación orientada a eventos		
Academia:	Ciencias Computacionales y Programación /	Clave:	19SDS09		
Módulo formativo:	Programación aplicada	Seriación:	19SDS18 - Computación paralela		
Tipo de curso:	Presencial	Prerrequisito:	19SDS06 - Programación orientada a objetos		
Semestre:	Tercero	Créditos:	8.44	Horas semestre:	135 horas
Teoría:	2 horas	Práctica:	3 horas	Trabajo indpt.:	3 horas
				Total x semana:	8 horas

Sección II. Objetivos educacionales

Tabla 2. Objetivos educacionales

Objetivos educacionales		Criterios de desempeño	Indicadores
1	Los egresados gestionarán recursos relacionados con el desarrollo de software en alguna organización.	Los egresados podrán aplicar metodologías en el desarrollo de proyectos en el contexto laboral.	20% de los egresados aplicarán metodologías en el desarrollo de software en su contexto laboral.
2	Los egresados diseñarán e implementarán soluciones innovadoras mediante el uso de tecnologías de la información.	Los egresados participarán activamente en el ciclo de desarrollo e integración continuos	25% de los egresados desempeñarán labores de desarrollo e integración continuos.
3	Los egresados desarrollarán conocimiento especializado que les permite enfocarse en un área del conocimiento específico del desarrollo de software.	Los egresados desempeñarán actividades orientadas al aseguramiento de los activos de información de manera resiliente, la gestión de la infraestructura de redes y comunicaciones, o integrando hardware y software para crear soluciones IoT; así como el uso de inteligencia artificial para gestionar datos y reconocer patrones que determinen oportunidades de negocio en las organizaciones.	5% de los egresados desempeñarán labores en desarrollo de soluciones IoT.
4	Los egresados serán capaces de emprender un negocio basado en el desarrollo de un producto o servicio de tecnologías de la información, aportando valor a la generación de empleos e incrementar el bienestar económico y social, de forma ecológica y sustentable.	Los egresados serán capaces de emprender un negocio basado en el desarrollo propio de un producto o servicio de tecnologías de la información.	2% de los egresados tendrán participación en el acta constitutiva de una empresa creada a partir del desarrollo de software para ofrecer un producto o servicio.



Atributos de egreso de plan de estudios		Criterios de desempeño	Componentes
1	Aplicar los conocimientos de ciencias básicas como física y matemáticas, así como las ciencias de la ingeniería para generar nuevos productos o servicios basándose en la innovación tecnológica.	<ul style="list-style-type: none"> - Conocerá la terminología y la utilización de las características generales de un programa para Windows para el diseño de una ventana de Windows. - Reconocerá el funcionamiento, las operaciones y la implementación que tienen los controles disponibles para las GUI, utilizando el paradigma de la programación orientada a eventos. 	<ul style="list-style-type: none"> 1.a Aspectos básicos. <ul style="list-style-type: none"> 1.a.i Funciones básicas del sistema operativo. 1.a.ii Interrupciones. 1.a.iii Procesos. 1.a.iv Cola de mensajes. 1.a.v Funciones Call Back. 1.a.vi Mensajes. <ul style="list-style-type: none"> 1. Estructura genérica de un mensaje. 2. Envío Post. 3. Envío Send. 1.b. Ventanas. <ul style="list-style-type: none"> 1.b.i Clase ventana. <ul style="list-style-type: none"> 1. Características. 2. Tipos. 3. Componentes. 4. Estilos. 1.b.ii Creación de una ventana. 1.b.iii Procedimiento ventana. 1.b.iv Bomba de mensajes del proceso. 1.c. Eventos. <ul style="list-style-type: none"> 1.c.i Mouse. 1.c.ii Teclado. 1.c.iii Controles de videojuego. 1.c.iv Eventos temporizado. 3.1 componentes de la interface gráfica. <ul style="list-style-type: none"> 3.1.a Botones.



Continuación: Tabla 2. Objetivos educacionales (continuación)

No.	Atributos de egreso de plan de estudios	Criterios de desempeño	Componentes
			3.1.b Cajas de texto. 3.1.c ScrollBars. 3.1.d Menús. 3.1.e Cuadros de diálogo.
2	Desarrollar una experimentación adecuada para recopilar, almacenar y analizar grandes cantidades de información basándose en el juicio ingenieril para crear productos o servicios innovadores mediados por software.	- Conocerá la terminología y la utilización de las interfaces gráficas para el diseño de un programa.	2.1 interface gráfica. 2.1.a Definición de GUI. 2.1.b Importancia. 2.1.c Estilos de interacción. 2.1.d Principios de diseño. 2.1.e Estándares de diseño y guías de estilo. 2.1.f Usabilidad. 2.1.g Velocidades de interacción humana.
3	Reconocer la mejora continua como parte de su desarrollo profesional para mantener un perfil actualizado en desarrollo de software para el diseño e implementación de productos y servicios basados en tecnologías con las tendencias emergentes.	- Reconocerá el funcionamiento, las operaciones y la implementación que tienen los recursos disponibles para las GUI, utilizando el paradigma de la programación orientada a eventos. - Reconocerá el funcionamiento, la sintaxis y la implementación que tienen los distintos modelos de arquitectura de software, utilizando el paradigma de la programación orientada a eventos.	4.1 Manejo de recursos. 4.1 Iconos. 4.2 Cursores. 4.3 Menús. 4.4 Cuadros de diálogo. 4.5 Imágenes. 4.6 Tablas de cadenas. 4.7 Versiones. 5.1 Tipos de modelos. 5.1.a Vista controlador. 5.1.b Vista presentador.

Sección III. Atributos de la asignatura

Tabla 3. Atributos de la asignatura

Problema a resolver		
El alumno desarrolla e implementa aplicaciones de software en las que incorpora los conceptos del paradigma de la Programación Orientada a Eventos, principalmente para el control de flujo de la interacción del usuario con el sistema de software.		
Atributos (competencia específica) de la asignatura		
Aplicar tecnologías y herramientas actuales y emergentes de programación para desarrollar sistemas de información que ofrezcan soluciones a problemas del entorno.		
Aportación a la competencia específica		Aportación a las competencias transversales
Saber	Saber hacer	Saber Ser
- Conocer los conceptos básicos relativos a la programación orientada a eventos, así como su relación con los elementos del sistema operativo del pc.	- Aplicar la programación visual orientada a eventos en el diseño de interfaces visuales.	Actitudinales: - Trabajo en equipo. - Iniciativa. - Juicio. Valores: - Ético. - Honesto. - Responsable.
Producto integrador de la asignatura, considerando los avances por unidad		
Diseño e implementación de un software editor básico de formas geométricas, el cual debe permitir manipular dichas formas a través de la interfaz de mouse, editarlas y guardarlas (serializarlas), todo lo anterior usando una arquitectura Documento-Vista.		

Sección IV. Desglose específico por cada unidad formativa

Tabla 4.1. Desglose específico de la unidad "Programación."

Número y nombre de la unidad: 1. Programación.							
Tiempo y porcentaje para esta unidad:		Teoría:	8 horas	Práctica:	15 horas	Porcentaje del programa:	25.56%
Aprendizajes esperados:		- Identificar y aplicar la sintaxis de estructuras básicas del sistema operativo para familiarizarse con el entorno del lenguaje.			- Manipular los controles y componentes estándar definidos en el lenguaje para el desarrollo de aplicaciones.		
Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad (Evidencia de aprendizaje de la unidad)			
1.a Aspectos básicos. 1.a.i Funciones básicas del sistema operativo. 1.a.ii Interrupciones. 1.a.iii Procesos. 1.a.iv Cola de mensajes. 1.a.v Funciones Call Back. 1.a.vi Mensajes. 1. Estructura genérica de un mensaje. 2. Envío Post. 3. Envío Send. 1.b. Ventanas. 1.b.i Clase ventana. 1. Características. 2. Tipos. 3. Componentes. 4. Estilos. 1.b.ii Creación de una ventana. 1.b.iii Procedimiento ventana.	Saber: - Apropiar conceptos básicos de la asignatura. - Conocer los componentes de una interface gráfica (ventana o Windows) así como el trasfondo del manejo de su interacción con el sistema operativo. - Analizar la función de los eventos en una interfaz gráfica. Saber hacer: - Implementar ventanas y el manejo de sus recursos.	- Exposición por parte del profesor mediante material audiovisual. - Informe de lectura mediante una línea de tiempo. - Desarrollo de prácticas acorde al tema de ventanas y envío Post y Send.	Evaluación formativa: - Reporte de ejercicios prácticos con rúbrica. - Reporte de producto integrador (Clase ventanas) con rúbrica. Evaluación sumativa: - Examen escrito.	Componente de software "Clase Ventana" operativo.			



Continuación: Tabla 4.1. Desglose específico de la unidad "Programación."

Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad
1.b.iv Bomba de mensajes del proceso. 1.c. Eventos. 1.c.i Mouse. 1.c.ii Teclado. 1.c.iii Controles de videojuego. 1.c.iv Eventos temporizados.	Ser: Es responsable y disciplinado para elaborar sistemas de software que cumplan con los requisitos solicitados.			
Bibliografía				
- Lozano, A. (2015). Iniciación a la programación utilizando lenguajes visuales orientados a eventos. España: Bellisco. - Yeager, D. (2014). Object-Oriented Programming Languages and Event-Driven Programming. EE.UU.: Mercury Learning & Information.				

Sección IV. Desglose específico por cada unidad formativa

Tabla 4.2. Desglose específico de la unidad "La interfaz de usuario gráfica."

Número y nombre de la unidad: 2. La interfaz de usuario gráfica.							
Tiempo y porcentaje para esta unidad:		Teoría:	4 horas	Práctica:	15 horas	Porcentaje del programa:	21.11%
Aprendizajes esperados: - Investigar e implementar los principales estándares para el desarrollo de interfaz gráfica de usuario en las aplicaciones de software.							
Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad (Evidencia de aprendizaje de la unidad)			
2.1 interface gráfica. 2.1.a Definición de GUI. 2.1.b Importancia. 2.1.c Estilos de interacción. 2.1.d Principios de diseño. 2.1.e Estándares de diseño y guías de estilo. 2.1.f Usabilidad. 2.1.g Velocidades de interacción humana. 2.1.h Consideraciones humanas en el diseño. 2.1.i Consideraciones de la tecnológicas. 2.1.j Algoritmos de dibujo. 2.1.k Hittest.	Saber: - Conocer los diferentes tipos de interacción con las GUI y sus estándares. - Analizar la función de las GUI y su interacción con el usuario. Saber hacer: - Implementar aplicaciones GUI basadas en estándares. Ser: Establece un compromiso de responsabilidad y ético al desarrollar interfaces gráficas de usuario acorde a	- Exposición por parte del profesor mediante material audiovisual. - Desarrollo de prácticas acorde a estándares, del tema de algoritmos de flujo.	Evaluación formativa: - Reporte de ejercicios prácticos con rúbrica. - Reporte de producto integrador (Clase figuras) con rúbrica (algoritmos de dibujo). Evaluación sumativa: - Examen escrito.	Componente de software "Clase Figuras" operativo.			



Continuación: Tabla 4.2. Desglose específico de la unidad "La interfaz de usuario gráfica."

Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad
	estándares tecnológicos en el desarrollo de software.			
Bibliografía				
<p>- Lozano, A. (2015). Iniciación a la programación utilizando lenguajes visuales orientados a eventos. España: Bellisco.</p> <p>- Yeager, D. (2014). Object-Oriented Programming Languages and Event-Driven Programming. EE.UU.: Mercury Learning & Information</p>				

Sección IV. Desglose específico por cada unidad formativa

Tabla 4.3. Desglose específico de la unidad "Controles."

Número y nombre de la unidad: 3. Controles.							
Tiempo y porcentaje para esta unidad:		Teoría:	6 horas	Práctica:	9 horas	Porcentaje del programa:	16.67%
Aprendizajes esperados: - Utilizar los controles básicos (Botones, Cajas de texto, ScrollBar, Menús y Cuadros de diálogo) para desarrollar una interfaz gráfica de usuario.							
Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad (Evidencia de aprendizaje de la unidad)			
3.1 Componentes de la interface gráfica. 3.1.a Botones. 3.1.b Cajas de texto. 3.1.c ScrollBars. 3.1.d Menús. 3.1.e Cuadros de diálogo. 3.1.4 Recursos. a Iconos. b Cursores. c Menús. d Cuadros de diálogo. e Imágenes. f Tablas de cadenas. g Versiones.	Saber: - Conocer los distintos tipos de controles disponibles para las GUI. - Analizar la mejor función de los controles para su mejor aprovechamiento. Saber hacer: - Implementar rutinas para usar los distintos controles. - Usar los distintos tipos de controles disponibles para las GUI. Ser: Establece en trabajo de equipo y colaborativo la responsabilidad de utilizar controles básicos para desarrollar interfaces	- Exposición por parte del profesor mediante material audiovisual. - Desarrollo de prácticas acorde a estándares, del tema de interfaz gráfica.	Evaluación formativa: - Reporte de ejercicios prácticos con rúbrica. - Reporte de producto integrador (Clase Controles) con rúbrica. Evaluación sumativa: - Examen escrito.	Componente de software "Clase Controles" operativo.			



Continuación: Tabla 4.3. Desglose específico de la unidad "Controles."

Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad
	gráficas de usuario.			
Bibliografía				
- Lozano, A. (2015). Iniciación a la programación utilizando lenguajes visuales orientados a eventos. España: Bellisco.				
- Yeager, D. (2014). Object-Oriented Programming Languages and Event-Driven Programming. EE.UU.: Mercury Learning & Information.				

Sección IV. Desglose específico por cada unidad formativa

Tabla 4.4. Desglose específico de la unidad "Recursos."

Número y nombre de la unidad: 4. Recursos.							
Tiempo y porcentaje para esta unidad:		Teoría:	6 horas	Práctica:	9 horas	Porcentaje del programa:	16.67%
Aprendizajes esperados: - Utilizar los Recursos (Iconos, Cursores, Menús, Cuadros de diálogo, Imágenes) para desarrollar una interfaz gráfica de usuario.							
Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad (Evidencia de aprendizaje de la unidad)			
Manejo de recursos. 4.1 Iconos. 4.2 Cursores. 4.3 Menús. 4.4 Cuadros de diálogo. 4.5 Imágenes. 4.6 Tablas de cadenas. 4.7 Versiones.	Saber: - Conocer los recursos disponibles para una GUI y su función. - Analizar la función de los recursos. Saber hacer: - Implementar rutinas para usar los distintos- recursos. Ser: Establece en trabajo de equipo y colaborativo la responsabilidad de utilizar controles avanzados para desarrollar interfaces gráficas de usuario.	- Exposición por parte del profesor mediante material audiovisual. - Desarrollo de prácticas acorde al tema de Recursos.	Evaluación formativa: - Reporte de ejercicios prácticos con rúbrica. Evaluación sumativa: - Reporte de producto integrador (Clase Recursos) con rúbrica.	Componente de software "Clase Recursos" operativo.			
Bibliografía							
- Lozano, A. (2015). Iniciación a la programación utilizando lenguajes visuales orientados a eventos. España: Bellisco. - Yeager, D. (2014). Object-Oriented Programming Languages and Event-Driven Programming. EE.UU.: Mercury Learning & Information.							

Sección IV. Desglose específico por cada unidad formativa

Tabla 4.5. Desglose específico de la unidad "Modelos."

Número y nombre de la unidad: 5. Modelos.							
Tiempo y porcentaje para esta unidad:		Teoría:	7 horas	Práctica:	11 horas	Porcentaje del programa:	20%
Aprendizajes esperados:		- Aplicar un Modelo de arquitectura de software para generar soluciones informáticas implementando el paradigma de la Programación Orientada a Eventos.					
Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad (Evidencia de aprendizaje de la unidad)			
5.1 Tipos de modelos. 5.1.a Vista controlador. 5.1.b Vista presentador. 5.1.c Vista-Adaptador. 5.1.d MVC Jerárquico. 5.1.e Cliente-Servidor.	Saber: - Conocer los distintos modelos de arquitectura de software, sus variaciones y su función. - Analizar las arquitecturas de software para determinar la manera en que se estructura un sistema, misma que impacta sobre la capacidad de éste para satisfacerlo que se conoce como los atributos de calidad del sistema. Saber hacer: - Seleccionar la arquitectura apropiada para	- Exposición por parte del profesor mediante material audiovisual. - Investigación acorde al tema de tipos de modelos.	Evaluación formativa: - Reporte de ejercicios prácticos con rúbrica. Evaluación sumativa: - Reporte de producto integrador final con rúbrica.	Producto integrador final: software editor básico de formas geométricas.			



Continuación: Tabla 4.5. Desglose específico de la unidad "Modelos."

Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad
	<p>el problema que se desea resolver a fin de brindar la mejor solución.</p> <p>Ser: Compromiso ético y responsable para trabajar en equipo para implementar software con base en Modelos de programación.</p>			
Bibliografía				
<p>- Lozano, A. (2015). Iniciación a la programación utilizando lenguajes visuales orientados a eventos. España: Bellisco.</p> <p>- Yeager, D. (2014). Object-Oriented Programming Languages and Event-Driven Programming. EE.UU.: Mercury Learning & Information.</p>				



V. Perfil docente

Tabla 5. Descripción del perfil docente

Perfil deseable docente para impartir la asignatura
<p>Carrera(s): - Ingeniería en Tecnología de la Información y Comunicación.</p> <ul style="list-style-type: none">- Licenciatura en informática.- Ingeniería en Tecnologías de la Información y Comunicación.- Ingeniería en Tecnologías de la Información y Comunicaciones- Ingeniería en Tecnologías de Información y Comunicaciones.- Licenciatura en Comunicación e Información.- Ingeniería en Sistemas de Computación.- Ingeniería en Sistemas Computacionales.- Licenciatura en Ingeniería en Sistemas computacionales.- Licenciatura en Ingeniería en Sistemas de Información.- Ingeniería en Sistemas Estratégicos de Información, Especialización en redes.- Licenciatura en Sistemas Computacionales. o carrera afín<ul style="list-style-type: none">- Con experiencia docente o en el campo deseable de 2 años. Manejo de TIC´s. Con habilidades pedagógicas y uso de metodologías alternativas de enseñanza.- Experiencia mínima de dos años- Licenciatura o superior.