



IFCD043PO: Programación de aplicaciones orientadas a objetos

Objetivos

□ **Objetivo general**

- Programar aplicaciones en un contexto orientado a objetos utilizando JAVA.

□ **Objetivos específicos**

- Comprender los diferentes paradigmas de programación y su aplicación en el desarrollo de software.
- Identificar y utilizar correctamente los elementos básicos de programación como variables, constantes y estructuras de control.
- Reconocer la importancia de la arquitectura de software y aplicar conceptos de modularidad y desacoplamiento.
- Utilizar diagramas UML y patrones de diseño para representar visualmente el software y mejorar su estructuración.
- Implementar técnicas de gestión de excepciones y procesamiento de documentos XML para asegurar la robustez del software.
- Aplicar estrategias de Refactorización, versionado y pruebas automatizadas para optimizar la calidad y mantenibilidad del código.
- Comprender los principios de la programación estructurada para aplicarlos eficazmente en la escritura de programas.
- Identificar y corregir errores comunes de sintaxis en lenguajes estructurados.
- Manejar variables, operadores y estructuras de control para realizar operaciones básicas y gestionar el flujo de ejecución.
- Utilizar adecuadamente arrays y cadenas (strings) en tareas de búsqueda, modificación y procesamiento de datos.

- Comprender los principios básicos que sustentan la Programación Orientada a Objetos para aplicarlos adecuadamente en proyectos de desarrollo.
- Identificar y utilizar correctamente los conceptos de encapsulación y abstracción en la construcción de software eficiente y bien estructurado.
- Aplicar eficazmente los mecanismos de herencia y polimorfismo para promover la reutilización y extensión del código en diferentes escenarios técnicos.
- Desarrollar aplicaciones prácticas en contextos web y cliente-servidor utilizando técnicas avanzadas orientadas a objetos, garantizando soluciones robustas y escalables.
- Comprender el funcionamiento de las bases de datos relacionales y su papel en el desarrollo de aplicaciones orientadas a objetos.
- Analizar los distintos componentes estructurales y lógicos que intervienen en el diseño y la gestión de datos relacionales.
- Aplicar los lenguajes de definición y manipulación de datos para la creación, consulta y actualización de bases de datos.
- Evaluar las características y funcionalidades de diversos sistemas gestores de bases de datos en función de sus requerimientos técnicos y operativos.
- Programar aplicaciones móviles en Java utilizando el entorno Android Studio de manera eficiente.
- Gestionar adecuadamente los recursos en aplicaciones Android para garantizar compatibilidad entre distintos dispositivos.
- Implementar técnicas avanzadas de integración multimedia, incluyendo cámaras y reproducción de vídeos.
- Aplicar estrategias avanzadas de internacionalización, localización y gestión de roles para aplicaciones móviles seguras y adaptables.
- Instalar y configurar adecuadamente gestores de contenidos para diversos tipos de proyectos web.
- Personalizar plataformas digitales mediante el uso efectivo de plantillas, módulos, componentes y plugins comunes.

- Aplicar técnicas avanzadas de optimización SEO para mejorar la visibilidad y el posicionamiento web.
- Realizar actualizaciones periódicas y tareas de mantenimiento necesarias para garantizar la seguridad y eficiencia de los gestores de contenidos.

• **Contenidos**

IFCD043PO. Programación de aplicaciones orientadas a objetos	Tiempo estimado
<p>Unidad 1: Conceptos básicos.</p> <ol style="list-style-type: none"> 1. Conceptos básicos. <ol style="list-style-type: none"> 1.1. Paradigmas de programación: estructurada, orientada a objetos, funcional. 2. Fundamentos. Expresiones. Entornos de desarrollo. <ol style="list-style-type: none"> 2.1. Variables y constantes. 2.2. Clasificación de datos. 2.3. Estructuras de control. 3. Plataformas soportadas. Patrones. 4. Diagramas UML. Traducción de UML a código y viceversa. <ol style="list-style-type: none"> 4.1. Diagramas de clases, secuencia y casos de uso. 4.2. Herramientas para la creación de UML. 5. Arquitectura. Uso de interfaces. <ol style="list-style-type: none"> 5.1. Arquitectura de software en capas. 5.2. Diseño modular y desacoplamiento. 6. Paquetes y manejo de excepciones. <ol style="list-style-type: none"> 6.1. Importación de paquetes y organización de clases. 6.2. Gestión de excepciones: try, catch, finally. 7. Manipulación de documentos XML. <ol style="list-style-type: none"> 7.1. Estructura de un documento XML. 7.2. Parsers XML y APIs de manejo (DOM, SAX). 8. Refactorización de código. 9. Versionado de software. <ol style="list-style-type: none"> 9.1. Sistemas de control de versiones: Git, SVN. 10. El testing. <ol style="list-style-type: none"> 10.1. Herramientas de testing: JUnit, TestNG. 11. Las sentencias. <ol style="list-style-type: none"> 11.1. Entornos de desarrollo. 11.2. De software libre. 11.3. Privativos. 	
Cuestionario de Autoevaluación UA 01	30 minutos
Actividad de Evaluación UA 01	90 minutos
Tiempo total de la unidad	19 horas
<p>Unidad 2: Programación estructurada.</p> <ol style="list-style-type: none"> 1. Programación estructurada. 2. Sintaxis. 	

<ul style="list-style-type: none"> 2.1. Reglas básicas de sintaxis en lenguajes estructurados. 2.2. Identificación de errores comunes de sintaxis. 3. Datos básicos. 4. Declaración de variables. Operadores. <ul style="list-style-type: none"> 4.1. Tipos de operadores: aritméticos, lógicos y relacionales. 5. Estructuras. Funciones y parámetros. <ul style="list-style-type: none"> 5.1. Estructuras de control: condicionales y bucles. 5.2. Funciones: creación, parámetros, valor de retorno. 6. Arrays. Strings. <ul style="list-style-type: none"> 6.1. Declaración y uso de arrays unidimensionales y multidimensionales. 6.2. Manejo de cadenas (strings): concatenación, modificación. 7. Buscar. Reemplazar. Extraer. Copiar. 8. Métodos recursivos. 9. Documentación con JavaDoc. 	
Cuestionario de Autoevaluación UA 02	30 minutos
Actividad de Evaluación UA 02	60 minutos
Tiempo total de la unidad	11 horas
<p>Unidad 3: Programación orientada a objetos.</p> <ul style="list-style-type: none"> 1. Programación orientada a objetos. 2. Filosofía. 3. Definición. Uso. Encapsulación. Abstracción. <ul style="list-style-type: none"> 3.1. Encapsulación: protección de datos y métodos. 3.2. Abstracción: simplificación de la complejidad. 4. Herencia. Polimorfismo. Diferencias. <ul style="list-style-type: none"> 4.1. Tipos de herencia: simple y múltiple. 5. Aplicaciones WEB. Aplicaciones autónomas. Applets. Servlets. Aplicaciones con ventanas. <ul style="list-style-type: none"> 5.1. Funcionalidad de applets y servlets. 6. Páginas dinámicas frente a páginas estáticas. <ul style="list-style-type: none"> 6.1. Herramientas para el desarrollo de páginas dinámicas (PHP, JavaScript). 7. Modelo cliente-servidor. Controles básicos. Conversión. <ul style="list-style-type: none"> 7.1. Arquitectura cliente-servidor: funcionamiento y aplicaciones. 7.2. Control de flujo y manejo de datos en aplicaciones cliente-servidor. 	
Cuestionario de Autoevaluación UA 03	30 minutos
Actividad de Evaluación UA 03	60 minutos
Tiempo total de la unidad	10 horas

<p>Unidad 4: Industria relacionada: bases de datos relacionales.</p> <ol style="list-style-type: none"> 1. Industria relacionada: bases de datos relacionales. 2. Conceptos fundamentales. <ol style="list-style-type: none"> 2.1. Claves primarias y foráneas. 3. Modelos de datos. El lenguaje de definición de datos (DDL): El lenguaje de manipulación de datos (DML). <ol style="list-style-type: none"> 3.1. Consultas SQL básicas: SELECT, INSERT, UPDATE, DELETE. 4. Estrategias para el control de las transacciones y de la concurrencia. Características y comparación de los SGBD más habituales. <ol style="list-style-type: none"> 4.1. Comparación de SGBD: MySQL, PostgreSQL, Oracle, SQL Server. 5. Herramientas. Importación y exportación de datos. <ol style="list-style-type: none"> 5.1. Estrategias de backup y recuperación de bases de datos. 	
<p>Cuestionario de Autoevaluación UA 04</p>	<p>30 minutos</p>
<p>Actividad de Evaluación UA 04</p>	<p>60 minutos</p>
<p>Tiempo total de la unidad</p>	<p>6 horas</p>
<p>Unidad 5: Programación Java para dispositivos móviles.</p> <ol style="list-style-type: none"> 1. Programación Java para dispositivos móviles. 2. Android. <ol style="list-style-type: none"> 2.1. Introducción al entorno de desarrollo Android Studio. 2.2. Gestión de recursos y compatibilidad entre dispositivos. 3. Librerías multimedia integradas. <ol style="list-style-type: none"> 3.1. Uso de cámaras y reproducción de videos. 4. Resolución de problemas comunes. Principales Frameworks. 5. Modelado de las relaciones de las tablas: Mapeo-objeto-relacional. 6. Creación de una plantilla. Desarrollo de la aplicación utilizando el patrón MVC. Internalización y localización. Autenticación de usuarios - Gestión de accesos por roles: autorización. <ol style="list-style-type: none"> 6.1. Estrategias para la internacionalización y localización de aplicaciones. 6.2. Implementación del sistema de roles y autorización. 7. Integración de componentes ya desarrollados: gráficos estadísticos. <ol style="list-style-type: none"> 7.1. Integración de AJAX para tareas habituales. 7.2. Uso de gráficos interactivos y su integración en la aplicación. 	

Cuestionario de Autoevaluación UA 05	30 minutos
Actividad de Evaluación UA 05	45 minutos
Tiempo total de la unidad	10 horas
Unidad 6: Gestores de contenidos.	
<ol style="list-style-type: none"> 1. Gestores de contenidos. 2. Instalación, configuración, personalización. Plantillas. Módulos, componentes y plugings más utilizados. 3. Técnicas de SEO. Nociones de seguridad. 4. Actualización y mantenimiento de los gestores de contenidos. 	
Cuestionario de Autoevaluación UA 06	30 minutos
Actividad de Evaluación UA 06	60 minutos
Tiempo total de la unidad	3 horas
Examen final	1 hora
6 unidades	60 horas