IFCD78 DESARROLLO DE APPS PARA DISPOSITIVOS MÓVILES II

SKU: EVOL-6656-iNO-B-1-1 | **Categorías:** INFORMÁTICA Y COMUNICACIONES, Programación y Desarrollo

INFORMACIÓN DEL CURSO

Horas	60
Acreditado por Universidad	NO
Créditos ECTS	0
Más información	CONTENIDO ADAPTADO A CERTIFICADO DE PROFESIONALIDAD

CURSO DE ESPECIALIZACIÓN

Objetivos

Este Curso IFCD78 DESARROLLO DE APPS PARA DISPOSITIVOS MÓVILES II le ofrece una formación especializada en la materia dentro de la Familia Profesional de Informática y comunicaciones. Con este CURSO IFCD78 DESARROLLO DE APPS PARA DISPOSITIVOS MÓVILES II el alumno será capaz de Capacitar para la generación de aplicaciones personales y/o profesionales para móviles en lenguaje Kotlin, utilizando el entorno de desarrollo Android Studio y la suite de librería Android JetPack, atendiendo a patrones de diseño modernos, como MVVM, y dentro de un plan formativo eminentemente práctico para el desarrollo y publicación de aplicaciones nativas para dispositivos móviles Android.

Contenidos

UNIDAD DIDÁCTICA 1. INTRODUCCIÓN

1. Comprensión de cómo escribir una aplicación de Android compleja para que sea escalable, legible y modular para proyectos personales o profesionales.
2. Comprensión de cómo funcionan las vistas en Android.
3. Comprensión de los conceptos de programación de la industria, como la inyección de dependencias y la programación impulsada por eventos.
4. Comprensión de los nuevos componentes de la arquitectura de Google, AndroidX y JetPack.
5. Profundización en los patrones de diseño modernos, como la arquitectura Model-View-ViewModel (MVVM).
6. Más información sobre Kotlin y sus potentes funciones para el desarrollo de Android.

UNIDAD DIDÁCTICA 2. PUESTA EN MARCHA DE UN NUEVO PROYECTO DE KOTLIN

1. Instalación de Android Studio con Kotlin.
2. Introducción y configuración de Android Jetpack.
3. Incorporación y registro de la actividad de navegación.
4. Creación de modelos de datos en Kotlin.
5. Conversión de navegación a Kotlin (Migración al componente Navigation).
6. Fragmentos en Kotlin y su creación en el componente Navigation.
7. RecyclerView en Kotlin y creación de lista dinámicas.
8. RecyclerAdapter en Kotlin. Refactorización.
9. Ejercicio guionizado para consolidar formación del alumno.

UNIDAD DIDÁCTICA 3. VISTAS PERSONALIZADAS

1. Introducción a la gestión de vistas.
2. Creación de una vista personalizada en Java.
3. Conexión de vista personalizada a Kotlin.
4. Anidación y jerarquía en vistas.
5. Ejercicio guionizado para consolidar formación del alumno.

UNIDAD DIDÁCTICA 4. MODELO - VISTA - MODELO DE ARQUITECTURA VIEWMODEL (MVVM)

1. Introducción al modelo de arquitectura MVVM.
2. Creación de un modelo de vista utilizando componentes Arch (Architecture Components).
3. Trabajo con el componente LiveData Arch.
4. Gestión de eventos de usuario con contratos (Contracts).
5. Uso de interfaces (Interface) para crear contratos.
6. Vinculación de la vista y el modelo.
7. Creación de modelos "Barebone" en Kotlin.
8. Actividad de evaluación de los conocimientos adquiridos por el alumno.

UNIDAD DIDÁCTICA 5. MEJORA DEL CÓDIGO - DIFFUTIL

1. Uso de la utilidad DiffUtil para optimizar listas y vistas complejas.
2. Actividad de evaluación de los conocimientos adquiridos por el alumno.
3. Empleo de DiffUtil con clases selladas.
4. Actividad de evaluación de los conocimientos adquiridos por el alumno.

UNIDAD DIDÁCTICA 6. INYECCIÓN DE DEPENDENCIAS Y TAREAS

1. Introducción y gestión de inyección de dependencias en Kotlin.
2. Trabajo con funciones de alcance (Scopes).
3. Uso de enlaces y módulos.
4. Ejecución de tareas en segundo plano.
5. Codificación y estados.

6. Actividad de evaluación de los conocimientos adquiridos por el alumno.

UNIDAD DIDÁCTICA 7. GOOGLE ROOM, SQLITE Y EL ALMACENAMIENTO PERSISTENTE

1. Configuración de la librería Google Room.
2. Creación de un cliente de base de datos.
3. Conversión de modelos en entidades.
4. Creación de un Data Access Object (DAO) y repaso de sintaxis SQL.
5. Vinculación de base de datos a la app creada.
6. Actividad de evaluación de los conocimientos adquiridos por el alumno.

UNIDAD DIDÁCTICA 8. CORRUTINAS Y PROGRAMACIÓN ASINCRÓNICA

1. Introducción a las corrutinas.
2. Refactorización en tareas para usar corrutinas.
3. Refactorización en funciones de suspensión (Suspend).
4. Ejercicio guionizado para consolidar formación del alumno.