

Swift, desarrollo de aplicaciones para iOS - iPhone/iPad

Curso práctico de 5 días - 35h
Ref.: DAH - Precio 2025: 2 120€ sin IVA

Al finalizar el curso, los alumnos serán capaces de desarrollar aplicaciones iOS en Swift, el lenguaje de programación de Apple, utilizando las distintas funciones del iPhone y el iPad.

OBJETIVOS PEDAGÓGICOS

Al término de la formación, el alumno podrá:

Comprender el ecosistema de desarrollo de los dispositivos móviles de Apple

Uso del lenguaje Swift y de las distintas API

Dominar la arquitectura de una aplicación móvil escrita en Swift

Optimización, creación e implantación de aplicaciones de alto rendimiento

Diseñar una interfaz gráfica

Intercambio de datos con un servidor

Los ejercicios se realizarán en Swift utilizando XCode e Interface Builder en Mac.

PROGRAMA

última actualización: 03/2024

1) Conceptos básicos del lenguaje Swift

- Variables, constantes, tuplas, cadenas.
- El facultativo y su gestión, al descubierto.
- Operadores y estructuras de control.
- Cierres y funciones.
- Clases, estructuras, enumeraciones, propiedades.
- Herencia y protocolos.
- La arquitectura de una aplicación móvil escrita en Swift.

Trabajo práctico : Aplicación de conceptos Swift.

2) El ecosistema de desarrollo de los dispositivos móviles de Apple

- XCode, la interfaz Builder y las distintas herramientas disponibles.
- El depurador.
- Herramientas de medición: instrumentos.
- El portal para desarrolladores de Apple.
- Optimización e implantación de una aplicación de alto rendimiento.

Trabajo práctico : Implementación con Swift e Interface Builder (UI).

3) Diseñar y construir una interfaz gráfica

- Arquitectura, memoria, consumo de energía y modos (Background, Foreground).
- AppDelegate: ciclo de vida y funcionamiento.
- Interface Builder: guiones gráficos y segmentos.
- Controlador de navegación, Controlador TableView, Controlador collectionView.
- Controlador tabBar: creación e implementación.

PARTICIPANTES

Promotores, jefes de proyecto, arquitectos.

REQUISITOS PREVIOS

Conocimientos prácticos de programación y de un lenguaje orientado a objetos (Objective-C, Java, C++, C#).

COMPETENCIAS DEL FORMADOR

Los expertos que imparten la formación son especialistas en las materias tratadas. Han sido validados por nuestros equipos pedagógicos, tanto en el plano de los conocimientos profesionales como en el de la pedagogía, para cada curso que imparten. Cuentan al menos con entre cinco y diez años de experiencia en su área y ocupan o han ocupado puestos de responsabilidad en empresas.

MODALIDADES DE EVALUACIÓN

El formador evalúa los progresos pedagógicos del participante a lo largo de toda la formación mediante preguntas de opción múltiple, escenificaciones de situaciones, trabajos prácticos, etc. El participante también completará una prueba de posicionamiento previo y posterior para validar las competencias adquiridas.

MEDIOS PEDAGÓGICOS Y TÉCNICOS

- Los medios pedagógicos y los métodos de enseñanza utilizados son principalmente: ayudas audiovisuales, documentación y soporte de cursos, ejercicios prácticos de aplicación y ejercicios corregidos para los cursillos prácticos, estudios de casos o presentación de casos reales para los seminarios de formación.
- Al final de cada cursillo o seminario, ORSYS facilita a los participantes un cuestionario de evaluación del curso que analizarán luego nuestros equipos pedagógicos.
- Al final de la formación se entrega una hoja de presencia por cada media jornada de presencia, así como un certificado de fin de formación si el alumno ha asistido a la totalidad de la sesión.

MODALIDADES Y PLAZOS DE ACCESO

La inscripción debe estar finalizada 24 horas antes del inicio de la formación.

ACCESIBILIDAD DE LAS PERSONAS CON DISCAPACIDAD

¿Tiene alguna necesidad específica de accesibilidad? Póngase en contacto con la Sra. FOSSE, interlocutora sobre discapacidad, en la siguiente dirección psh-accueil@orsys.fr para estudiar de la mejor forma posible su solicitud y su viabilidad.

- Gestión de vistas: UIView, widgets disponibles.
 - Dibujo en vistas y animaciones.
- Trabajo práctico : Implementación de conceptos sobre segues, TableView Controller (listas).*

4) Gestos

- Gestos en la vista.
- Gestos en el simulador: uso y limitaciones.
- Añadir gestos mediante Interface Builder, por código.

Trabajo práctico : Implementación de varios gestos en un controlador.

5) La red y los Servicios Web, intercambiando datos con un servidor

- Gestión de notificaciones push.
- Asíncrono con Grand Central Dispatch.
- Gestión de llamadas a Servicios Web en JSON.
- Serialización y deserialización en formato JSON.
- Organización de los datos.
- Gestión de preferencias y archivos en el sandbox.
- ORM: creación de modelos de datos, implementación.

Trabajo práctico : Llamada a Servicios Web en JSON y recursos en Internet. Gestión de datos con CoreData.

6) API de geolocalización y geocodificación

- Geolocalización mediante GPS y red.
- Arquitectura básica: CLLocationManager y seguridad.
- Aplicación de la geolocalización.
- Gestión de errores.

Trabajo práctico : Aplicación de la geolocalización.

7) Interacción entre Swift y Objective-C

- Herencia, sintaxis y tipos de datos: comparación.
- Puntos especiales y limitaciones.

8) Introducción a SwiftUI

- SwiftUI y XCode.
- Creación de vistas.
- Gestión de informes y datos compartidos.
- Gestión de listas y navegación.
- Gestos, animaciones y transiciones.
- UIViewControllers y SwiftUI.

Trabajo práctico : Cree un proyecto con vistas personalizadas, vistas múltiples y vistas de lista.

FECHAS

Contacto