



Universidad Pontificia de Salamanca

GUÍA DOCENTE 2025-2026

Máster universitario en
Informática Móvil

DESARROLLO DE APLICACIONES iOS

Modalidad virtual



DATOS BÁSICOS

Carácter	Obligatoria
Créditos	10 ECTS
Trimestre	1º y 2º
Calendario	De enero a marzo de 2026
Sesiones síncronas	miércoles y jueves de 19:00-21:30
Idioma	Español
Profesor responsable	Mª Encarnación Beato Gutiérrez
E-mail	ebeatogu@upsa.es
Otros profesores	Rodrigo Extremo Tabares (rextremota@upsa.es) Sergio Padrino Recio (spadrinore@upsa.es)
Tutorías	Los horarios podrán consultarse en la sección «Tutorías» del Aula Virtual de la asignatura

BREVE DESCRIPCIÓN DE LA ASIGNATURA

En esta asignatura se aprenderá cómo desarrollar aplicaciones nativas para iOS, una de las plataformas más utilizadas en el panorama de la programación de aplicaciones móviles actual. Se aprenderá el nuevo lenguaje de programación Swift, el manejo del entorno de desarrollo Xcode, el patrón de diseño Modelo Vista Controlador (MVC) utilizado en este tipo de aplicaciones, los elementos fundamentales de la interfaz de usuario, cómo utilizar el manejador de gestos, cómo construir aplicaciones con múltiples MVC, cómo gestionar notificaciones PUSH, mapas, sensores, animaciones, persistencia de datos así como la posibilidad de adaptar las aplicaciones a distintos tipos y tamaños de terminales.

REQUISITOS PREVIOS

No se establecen requisitos previos.

OBJETIVOS

- Conocer y aplicar los principios de la construcción del software basada en MVC (Modelo Vista Controlador)
- Dominar los aspectos fundamentales del desarrollo de una aplicación móvil nativa con tecnología iOS.
- Saber aplicar bibliotecas y frameworks al desarrollo de aplicaciones iOS
- Diseñar interfaces de usuario iOS, tanto para iPhone como para iPad.
- Conocer y saber realizar los pasos a seguir para la publicación de una aplicación en el App Storey.

RESULTADOS DE APRENDIZAJE

- CC03 Domina el desarrollo de aplicaciones móviles avanzadas en diferentes plataformas de dispositivos móviles
- HD04 Identifica y conoce frameworks y patrones de diseño en el desarrollo de aplicaciones móviles
- C01 Capacidad de análisis y síntesis
- C02 Habilidad para gestionar el tiempo, con capacidad de organización y temporalización de las tareas



- C03 Capacidad de gestión de la información
- C04 Capacidad para la toma de decisiones: capacidad de identificar, analizar y definir los elementos significativos que permitan tomar decisiones con criterio y de forma efectiva
- C05 Trabajo en Equipo
- C06 Adaptación a nuevas situaciones
- C07 Creatividad
- C08 Iniciativa y espíritu emprendedor
- C09 Capacidad de relacionarse con las personas y con el entorno sin perder autonomía, conservando su propia identidad y valores
- C11 Poseer y comprender conocimientos que aporten una base u oportunidad de ser originales en el desarrollo y/o aplicación de ideas, a menudo en un contexto de investigación
- C12 Aplicar los conocimientos adquiridos y su capacidad de resolución de problemas en entornos nuevos o poco conocidos, dentro de contextos más amplios (o multidisciplinares) relacionados con su área de estudio
- C13 Integrar conocimientos y enfrentarse a la complejidad de formular juicios a partir de una información que, siendo incompleta o limitada, incluya reflexiones sobre las responsabilidades sociales y éticas vinculadas a la aplicación de sus conocimientos y juicios
- C14 Comunicar sus conclusiones –y los conocimientos y razones últimas que las sustentan- a públicos especializados y no especializados de un modo claro y sin ambigüedades
- C15 Poseer habilidades de aprendizaje que les permitan continuar estudiando de un modo que habrá de ser en gran medida autodirigido o autónomo.

CONTENIDOS

1. Introducción al Desarrollo iOS

- Arquitectura iOS. Modelo Vista Controlador (MVC)
- Descripción del entorno de desarrollo. Xcode
- El lenguaje de programación. Swift

2. Interfaces Gráficas con UIKit

- Elementos básicos del UIKit: UIButton, UIImage, UITextField, UILabel, UISlider, UISwitch, UISegmentedControl
- Vistas. Jerarquía de vistas. Geometría de la vista
- Autolayout

3. Gestión de Controladores y Navegación

- Controladores. Ciclo de vida de un controlador



- Controladores de controladores. UINavigationController, UITabBarController

4. Componentes Avanzados de Interfaz

- Otros elementos del UIKit: UIAlertController, UIPickerView
- Reconocimiento de gestos
- Multimedia, animaciones, bloques

5. Gestión de Datos y Persistencia

- Configuración. UserDefaults
- Persistencia con JSON. Codable
- Persistencia en ficheros
- CoreData

6. Colecciones y Representación de Datos

- Colecciones de vistas y tablas: UICollectionView y UITableView

7. Desarrollo con SwiftUI y Combine

- SwiftUI
- Combine

8. Comunicación en Red y Servicios Web

- Servicios web. URLSession y Alamofire

9. Funcionalidades Avanzadas del Sistema

- Sensores
- Notificaciones locales, notificaciones push
- Aplicaciones universales

10. Monetización y Almacenamiento en la Nube

- Compras desde la aplicación (In-App). StoreKit
- iCloud

METODOLOGÍA

	HORAS	SÍNCRONAS	ASÍNCRONAS
CLASES VIRTUALES	27	34 (13,6%)	
TUTORÍAS VIRTUALES GRUPALES	7		
TRABAJO AUTÓNOMO DEL ESTUDIANTE	66		216 (86,4%)
RETOS ACADÉMICOS Y TRABAJO FINAL	150		
TOTAL	250	34	216

CRITERIOS DE EVALUACIÓN

Convocatoria ordinaria y extraordinaria

La nota que obtendrán los alumnos se calculará a través de la siguiente fórmula:

$$(0,80 \times TRB) + (0,20 \times RET)$$

donde:

TRB: Trabajo final

RET: Retos académicos

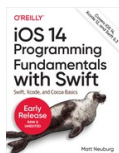
Para aprobar la asignatura será preciso desarrollar una aplicación móvil iOS. Aunque la temática de la aplicación, funcionalidades e interfaz será tarea del alumno, se deberá cumplir unos requisitos mínimos publicados en el aula virtual correspondientes a las distintas unidades didácticas.

El reto académico consistirá en la elaboración de una documentación en donde se realice una descripción funcional y una descripción técnica de la aplicación desarrollada. Se indicará cómo se ha realizado la aplicación, su arquitectura, los componentes y elementos utilizados. Además, será necesario indicar las razones que han motivado el desarrollo de la aplicación en la forma elegida, así como la justificación del uso de librerías de terceros indicando el motivo de su utilización y por qué se ha elegido esa y no otra librería parecida.

Tanto los retos académicos como en el trabajo final deben contener evidencias suficientes que garanticen la autoría del estudiante que los entrega. Estos deberán incluir acciones que únicamente pueda realizar el alumno y por tanto garanticen su autoría, por ello se deberá elaborar un breve vídeo en el que el estudiante argumente la cámara los pasos que ha dado para llevar a cabo el trabajo y explicar la razón de las decisiones clave de su desarrollo.

RECURSOS DE APRENDIZAJE Y APOYO TUTORIAL

Referencias bibliográficas



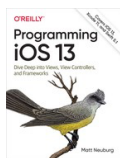
iOS14 Programming Fundamentals with Swift / Matt Neuburg. O'Reilly. 2020.
<http://shop.oreilly.com/>



iOS11 Swift Programming Cookbook / Vandad Nahavandipoor. O'Reilly.
2017. ISBN: 978-1491992470
<http://shop.oreilly.com/>



Universidad
Pontificia
de Salamanca



Programming iOS13 / Matt Neuburg. O'Really. 2019.
<http://shop.oreilly.com/>



iOS Apprentice: Beginning IOS development with Swift and UIKit.
Raywnderlich Tutorial Team. Razeware llc

https://koha.upsa.es/cgi-bin/koha/opac-detail.pl?biblionumber=656516&query_desc=kw%2Cwrdl%3A%20ios



Combine. Asynchronous Programming with Swift. Scott Gardner, Shai Mishali, Florent Pillet, Marin Todorov. Razeware llc

https://koha.upsa.es/cgi-bin/koha/opac-detail.pl?biblionumber=657738&query_desc=kw%2Cwrdl%3A%20ios



Design Patterns by Tutorials: Learning design patterns in Swift. Joshua Greene y Jay Strawn. Razeware llc

https://koha.upsa.es/cgi-bin/koha/opac-detail.pl?biblionumber=663246&query_desc=kw%2Cwrdl%3A%20ios



Xcode
<http://www.appStore.com>



Swift Programming Language
https://developer.apple.com/library/ios/documentation/Swift/Conceptual/Swift_Programming_Language/



Portal desarrolladores iOS
<http://developer.apple.com/ios>



Human Interface Guidelines
<https://developer.apple.com/ios/human-interface-guidelines/overview/design-principles/>

BREVE CV DEL PROFESOR RESPONSABLE

M^a Encarnación Beato Gutiérrez es Ingeniera en Informática y Doctora por la Universidad de Valladolid. En la actualidad es Catedrática de Lenguajes de Programación de la UPSA en la Facultad de Informática donde imparte la mayor parte de su docencia relacionada con esta materia. Posee las acreditaciones de profesor de universidad privada y profesor Contratado Doctor tanto por la ACSUCYL como por la ANECA y tiene dos sexenios de Investigación reconocidos por la CNEAI. Es autora de numerosas publicaciones científicas indexadas en los principales rankings de referencia (JCR y SCOPUS) y ha participado y/o dirigido más de 25 proyectos de investigación competitivos y posee más de 20 registros de la propiedad intelectual resultados de proyectos de investigación.



BREVE CV DEL RESTO DE PROFESORES

Rodrigo Extremo Tabarés es Ingeniero en Informática, Professional Scrum Master I por Scrum.org con más de 12 años de experiencia laboral. A lo largo de su trayectoria profesional, Rodrigo ha trabajado en FLAG Solutions y Deloitte como Senior Mobile Developer y iOS Delivery Lead. Actualmente trabaja en Plexus como Senior iOS Developer. Sus áreas de experiencia incluyen desarrollo de aplicaciones en iOS con amplia experiencia en entornos empresariales, enfocado en la seguridad, así como la aplicación de tecnologías punteras (iBeacons, CoreML) a soluciones móviles y desarrollo de soluciones basadas en SwiftUI.

Sergio Padrino Recio es Staff Software Engineer en GitHub (www.github.com), con más de 16 años de experiencia laboral. Actualmente forma parte del equipo de desarrollo de la aplicación GitHub Desktop (desktop.github.com). Anteriormente ha desempeñado el puesto de iOS Team Lead en Plex (www.plex.tv) y en Fever (www.feverup.com), y de Mobile App Engineer en Tuenti (www.tuenti.com) y en FLIR Systems (<http://www.flir.es>). Además, ha sido profesor de desarrollo de videojuegos en la Universidad de Alcalá de Henares.