Grand Master

Desarrollo de Aplicaciones Móviles, Experto en Android



tech universidad tecnológica



tech universidad tecnológica

Grand Master

Desarrollo de Aplicaciones Móviles, Experto en Android

Modalidad: **Online**

Duración: **2 años**

Titulación: **TECH Universidad Tecnológica**

Horas lectivas: **3.000 h.**

Acceso web: **www.techtitute.com/informatica/grand-master/grand-master-desarrollo-aplicaciones-moviles-experto-android**

Índice

01 Presentación — pág. 4

02 Objetivos — pág. 8

03 Competencias — pág. 18

04 Dirección del curso — pág. 22

05 Estructura y contenido — pág. 28

06 Metodología — pág. 48

07 Titulación — pág. 56

01

Presentación

Hoy en día no se puede concebir un mundo sin apps ni empresas que aprovechen las nuevas tecnologías. Las aplicaciones móviles son un paso más allá para cualquier empresa, constituyendo un signo de madurez de las mismas y un esfuerzo por adaptarse a los nuevos hábitos de los usuarios, lo que las hacen imprescindibles en prácticamente cualquier ámbito. Actualmente, la necesidad de las aplicaciones móviles está vinculada a las funcionalidades que pueden ofrecer más allá de la información que pueda poseer una página web. Es por ello que, hoy en día, existen más de 3.000 millones de dispositivos Android en el mundo y, en consecuencia, el sector del Desarrollo de aplicaciones móviles ha experimentado un crecimiento vertiginoso. Es ahora, más que nunca, cuando la industria presenta la necesidad imperiosa de profesionales especialistas en el Desarrollo de Aplicaciones, cuestión que ha impulsado la creación de este programa.





Observa, de la mano de expertos, todos los aspectos clave del Customer Experience para desarrollar aplicaciones en Android con gran entrada en el mercado"

Los informes del mercado laboral confirman la demanda creciente de perfiles expertos en el diseño de aplicaciones móviles, valorando la experiencia completa en el ciclo de vida del desarrollo, despliegue y monetización de las aplicaciones. Para dotar al informático en todos los conocimientos sobre el lenguaje de programación de aplicaciones, con especial enfoque en Android, las arquitecturas e interfaces de usuario. Este programa se ha diseñado de la mano de los mejores expertos de la materia, los cuales conforman el claustro docente.

Así, el informático adquirirá las capacidades para entender los diferentes lenguajes de programación para cada tipo de dispositivo, profundizando en el diseño *responsive*. Esto le permitirá desarrollar aplicaciones adaptables a las características distintas de cada uno de los aparatos. A su vez, el profesional profundizará en áreas como la Informática Web, Software, desarrollo de negocios, Marketing y ventas, consultorías, tecnología móvil multiplataforma, empresas, mundo del automóvil, domótica, Internet de las Cosas (IoT), banca y drones.

Asimismo, TECH dispone de la mejor metodología de estudio 100% online, lo que elimina la necesidad de asistir a clases presenciales o de seguir un horario predeterminado. De esta manera, en 24 meses los informáticos adquirirán la base precisa para dimensionar su propio negocio a partir del desarrollo de aplicaciones o bien, acometer tareas de consultoría en todos aquellos aspectos vinculados con la tecnología móvil sobre Android.

Este **Grand Master en Desarrollo de Aplicaciones Móviles, Experto en Android** contiene el programa educativo más completo y actualizado del mercado. Sus características más destacadas son:

- El desarrollo de casos prácticos presentados por expertos en Informática
- Los contenidos gráficos, esquemáticos y eminentemente prácticos con los que están concebidos recogen una información científica y práctica sobre aquellas disciplinas indispensables para el ejercicio profesional
- Los ejercicios prácticos donde realizar el proceso de autoevaluación para mejorar el aprendizaje
- Su especial hincapié en metodologías innovadoras para el desarrollo de aplicaciones móviles
- Las lecciones teóricas, preguntas al experto, foros de discusión de temas controvertidos y trabajos de reflexión individual
- La disponibilidad de acceso a los contenidos desde cualquier dispositivo fijo o portátil con conexión a internet

En este Grand Master estarás al día y conocerás a fondo los diferentes lenguajes de programación para cada tipo de dispositivo móvil"

A medida que domines todo el contenido sobre el Desarrollo de Aplicaciones para Dispositivos Móviles, comprenderás cómo monetizarlas y profundizarás en el Marketing Móvil"

Brinda soluciones eficientes a tu empresa, con el manejo de los fundamentos relacionados al Desarrollo de Aplicaciones en Sistemas Android.

TECH te da la oportunidad para aprender a desarrollar aplicaciones móviles de forma autónoma y profesional, en dispositivos multiplataforma.

Incluye en su cuadro docente a profesionales pertenecientes al ámbito de la informática, que vierten en este programa la experiencia de su trabajo, además de reconocidos especialistas de sociedades de referencia y universidades de prestigio.

Su contenido multimedia, elaborado con la última tecnología educativa, permitirá al profesional un aprendizaje situado y contextual, es decir, un entorno simulado que proporcionará un estudio inmersivo programado para entrenarse ante situaciones reales.

El diseño de este programa se centra en el Aprendizaje Basado en Problemas, mediante el cual el alumno deberá tratar de resolver las distintas situaciones de práctica profesional que se le planteen a lo largo del curso académico. Para ello, el profesional contará con la ayuda de un novedoso sistema de vídeo interactivo realizado por reconocidos expertos.

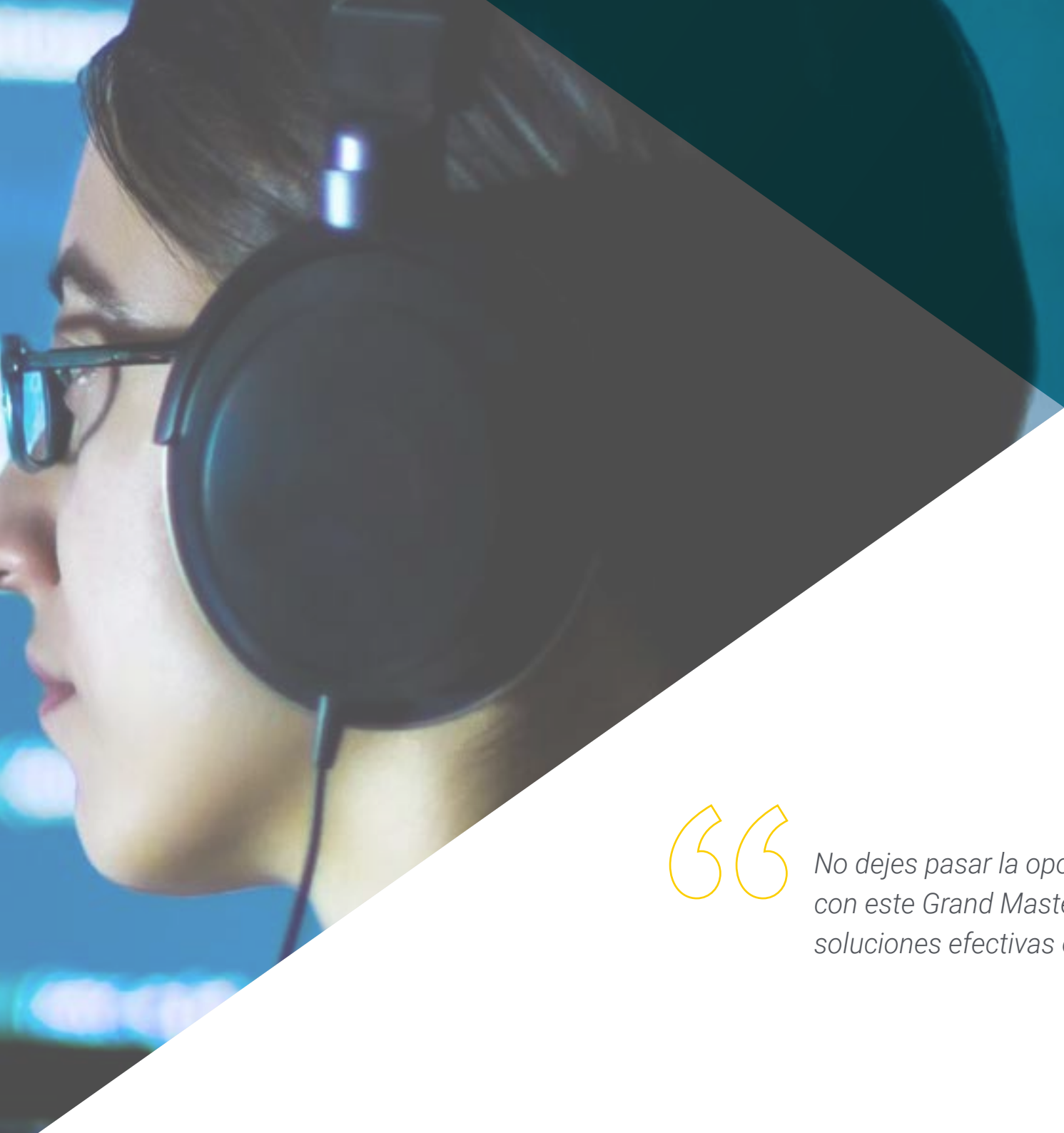


02

Objetivos

Debido al notorio crecimiento y consolidación que trae consigo el mundo de las tecnologías y dispositivos móviles, el informático está obligado a la continua actualización para adquirir todo el conocimiento necesario. Por ello, este programa le aporta todas las herramientas y fundamentos necesarios para profundizar en el desarrollo de aplicaciones móviles, con especial enfoque en el sistema Android, y brindar soluciones efectivas en el entorno profesional. El egresado que curse esta titulación estará al día en las tendencias y sabrá desempeñarse en todos los aspectos desde su diseño, programación hasta la experiencia del usuario final.

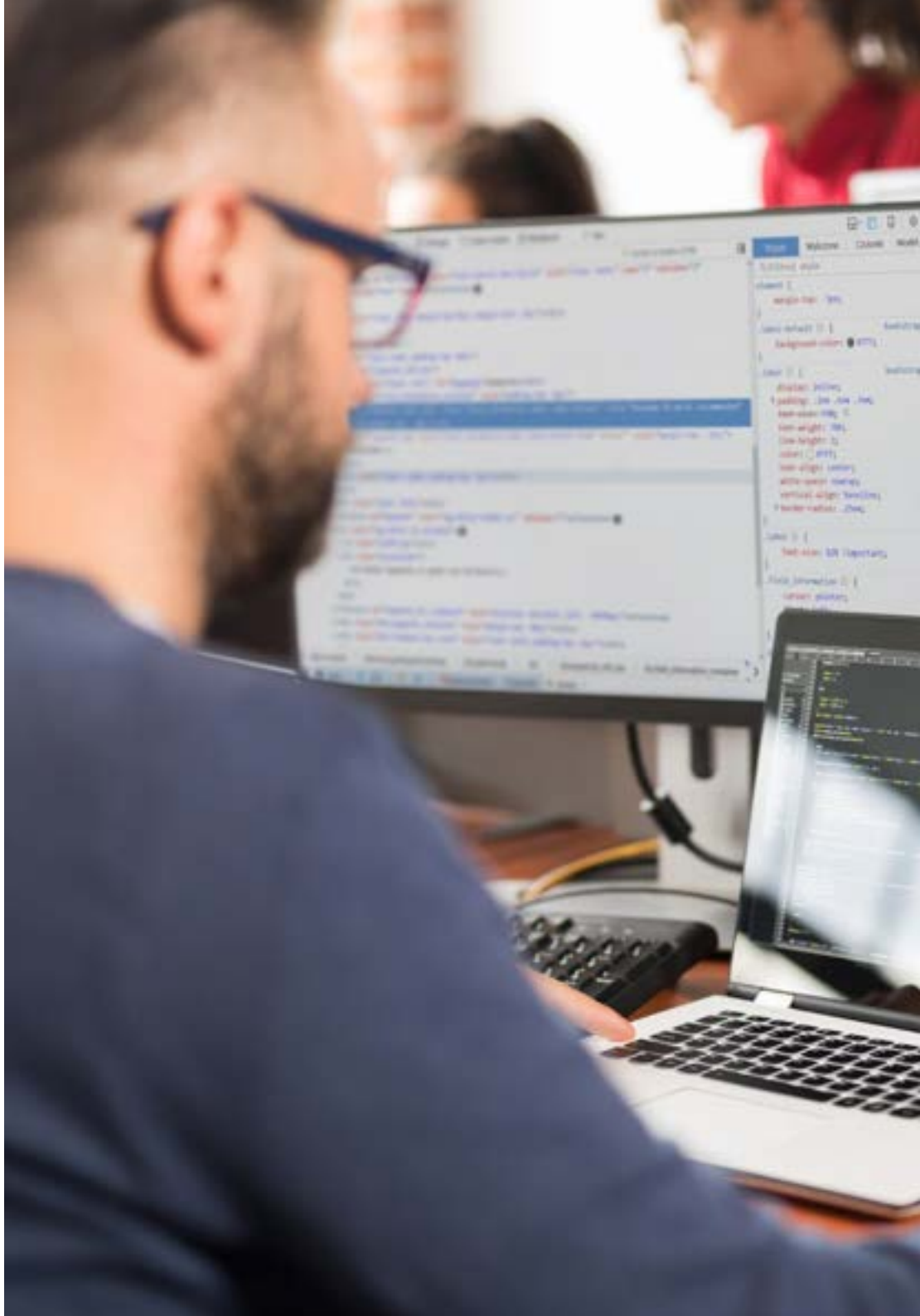


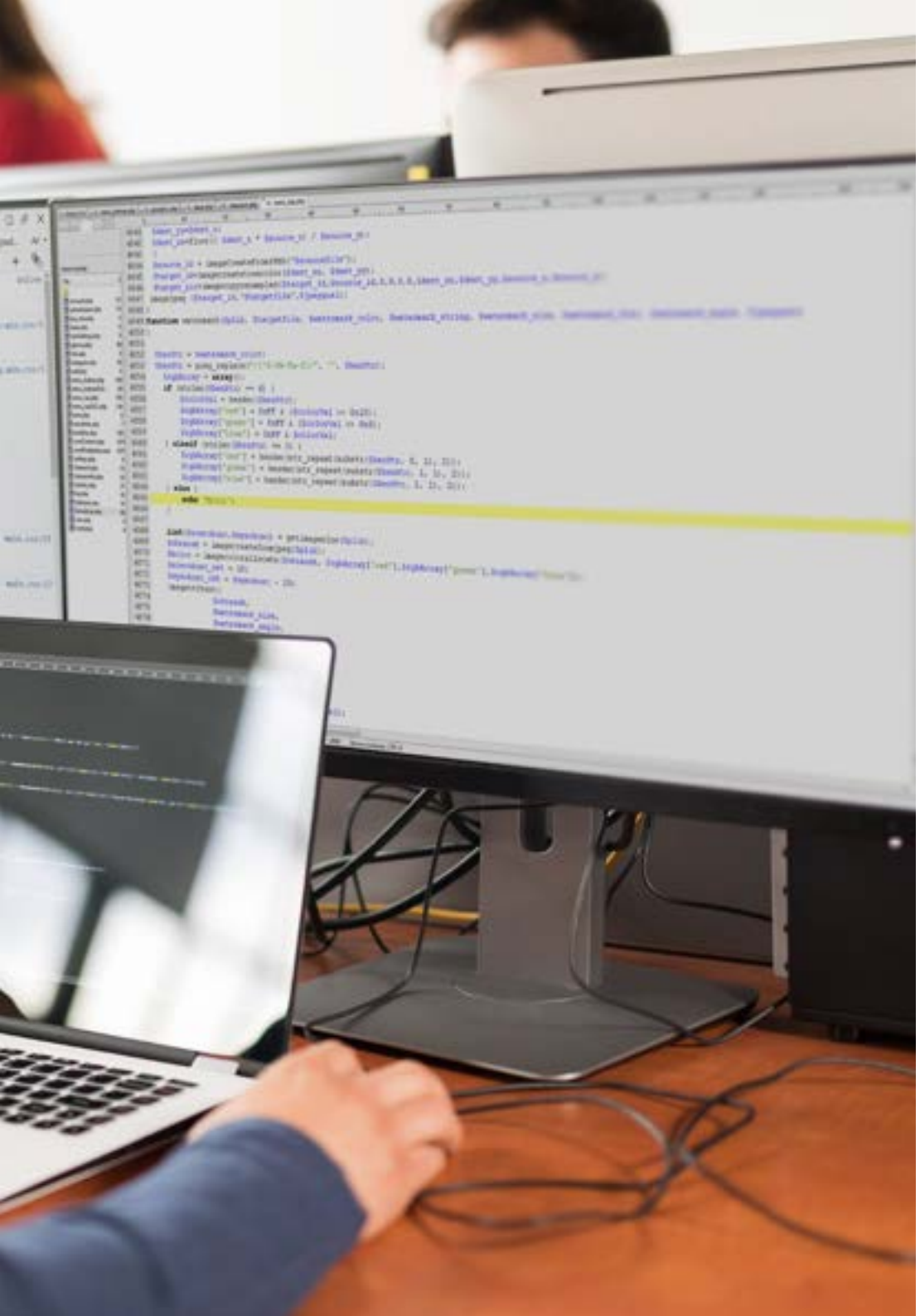


No dejes pasar la oportunidad de actualizarte con este Grand Master y poder brindar soluciones efectivas en el entorno profesional”

Objetivos generales

- Analizar las necesidades y el comportamiento de los usuarios en relación a los dispositivos móviles y sus aplicaciones
- Ejecutar el diseño de las arquitecturas, iteraciones e interfaces de usuario a través de los lenguajes de programación de las plataformas móviles más representativas del mercado (Web, iOS y Android)
- Aplicar los mecanismos de control, prueba y depuración de errores en el Desarrollo de Aplicaciones móviles
- Abordar diferentes casos prácticos y de negocio para publicar, distribuir y difundir aplicaciones móviles en los principales mercados de aplicaciones
- Dominar el conocimiento práctico para planificar y dirigir proyectos tecnológicos relacionados con tecnologías móviles
- Desarrollar las capacidades, aptitudes y herramientas necesarias para aprender a desarrollar aplicaciones móviles de forma autónoma y profesional, en dispositivos multiplataforma





- Explorar los contenidos relacionados con la monetización de aplicaciones y el Marketing Móvil
- Determinar los elementos estructurales de un Sistema Android
- Analizar las diferencias entre los distintos *Frameworks* de Desarrollo, puntos fuertes y debilidades
- Desarrollar capacidades avanzadas y mejores prácticas en el Desarrollo de Aplicaciones en el Lenguaje de Programación Kotlin
- Desarrollar una metodología para la gestión óptima de datos en el dispositivo
- Analizar los casos de uso para dispositivos Android en el mercado
- Dominar los elementos de un diseño responsive y superar retos derivados del mismo
- Compilar las diferentes etapas de un ciclo de integración continua enfocado al desarrollo en Android



Objetivos específicos

Módulo 1. Metodologías de programación en desarrollo de aplicaciones para dispositivos móviles

- Explorar procesos de desarrollo de software tradicional
- Analizar los procesos de desarrollo ágiles
- Promover prácticas de desarrollo
- Examinar las diferentes técnicas de representación y diagramación
- Profundizar en los diferentes patrones de diseño presente en la industria de software
- Explorar diferentes técnicas de prueba de software
- Reconocer las normas y estándares de calidad referencia en el desarrollo

Módulo 2. Tecnologías en desarrollo de aplicaciones para dispositivos móviles

- Establecer los conceptos para dispositivo móvil
- Compilar las principales plataformas
- Examinar sus componentes comunes
- Identificar componentes diferenciadores, sus capacidades y limitaciones
- Delimitar los diferentes escenarios en los que pueden operar. Ventajas
- Analizar las diferentes interacciones que estos dispositivos pueden mediar
- Concienciar sobre los diferentes abusos que se pueden cometer

Módulo 3. Herramientas de trabajo en desarrollo de aplicaciones para dispositivos móviles

- Preparar el entorno de Desarrollo
- Adquirir destrezas en la terminal de comandos
- Utilizar eficientemente el sistema de control de versiones
- Abordar el uso de sistemas de versionado de código remoto
- Asentar las nociones claves de funcionamiento del Internet
- Desarrollar los conceptos relevantes de la programación de software
- Examinar las estructuras de datos
- Revisar las técnicas de diseño e interpretación de algoritmos

Módulo 4. Desarrollo web multiplataforma orientado a dispositivos móviles

- Determinar las ventajas y limitaciones del modelo de desarrollo nativo y de desarrollo de Apps hibridas
- Examinar las características y limitaciones de las Progressive Web Apps (PWA)
- Analizar los principales *Frameworks* para el desarrollo de aplicaciones web: Angular, React, Vue
- Compilar las principales tecnologías para el desarrollo de aplicaciones móviles multiplataforma: *Ionic* y *Flutter*
- Analizar las capacidades para desplegar estas apps híbridas como Webs o Aplicaciones de Escritorio en PC
- Examinar un modelo para elegir la alternativa que mejor se adapte para el desarrollo de una aplicación específica

Módulo 5. Base de datos para el desarrollo de aplicaciones para dispositivos móviles

- Identificar el mejor modelo de base de datos en relación con las características de la aplicación móvil
- Establecer las capacidades de cada uno de los sistemas de base de datos
- Determinar las diferencias entre las distintas bases de datos
- Examinar cómo conectarse y cargar/extraer datos desde los diferentes tipos de bases de datos
- Analizar las capacidades básicas de entornos de desarrollo con capacidades de base de datos incluidas

Módulo 6. Desarrollo de aplicaciones para sistemas iOS

- Desarrollar una aplicación en Swift
- Utilizar los Cocoa Pods para gestionar las librerías
- Hacer uso de Alamofire para conectar nuestra aplicación con una RESTful API
- Concretar los requisitos básicos para la elección de una librería
- Desarrollar un sistema de monetización con ADMOB
- Diseñar vistas desde código
- Publicar una aplicación en App Store

Módulo 7. Despliegues en integración continua para dispositivos móviles

- Determinar el peor escenario que da pie a la necesidad de esta metodología
- Concretar los requisitos que el Software a integrar debe cumplir
- Establecer qué es la integración continua, la entrega continua y el despliegue continuo
- Analizar DevSecOps
- Examinar el monitoreo continuo
- Desarrollar las implementaciones de las diferentes etapas

Módulo 8. *User experience* en dispositivos móviles

- Analizar al nuevo tipo de usuario, sus interacciones y su recorrido en las aplicaciones y webs móviles
- Determinar las herramientas fundamentales de analítica web, movilidad y accesibilidad
- Concretar las técnicas de evaluación de micro-interacciones y el diseño de experiencias personalizadas
- Establecer cómo las nuevas tecnologías disruptivas como la IA o el IoT han llevado la experiencia de cliente a nuevos estándares
- Mostrar cómo la analítica del comportamiento genera una cantidad y calidad de datos nunca vistos en la analítica tradicional
- Desarrollar las nuevas metodologías como *Design Thinking*, centradas en el usuario
- Proponer herramientas básicas y avanzadas de prototipado y *wireframing*

Módulo 9. Seguridad en dispositivos móviles

- Determinar los elementos y niveles de seguridad existentes en un dispositivo móvil
- Establecer las técnicas a usar para un dispositivo
- Analizar los errores comunes en seguridad
- Examinar los mecanismos habilitados en la programación para evitar fallos de seguridad
- Concretar las recomendaciones que dan los Organismos de Seguridad
- Compilar las diferentes soluciones existentes en el mercado para la gestión de la seguridad de los dispositivos móviles dentro de las empresas
- Analizar los procesos criptográficos que se aplican a la seguridad de móviles

Módulo 10. Lenguaje de programación Android

- Examinar el núcleo Linux y la máquina virtual en la base de Android
- Analizar las librerías nativas del sistema
- Establecer las bondades de Android frente a otras plataformas
- Determinar los elementos de una aplicación Android
- Presentar las versiones Android y sus mejoras
- Evaluar el mercado de aplicaciones de Android
- Fundamentar la evolución futura de Android

Módulo 11. *Frameworks* utilizados en desarrollo de aplicaciones Android

- Analizar el *framework* de Android Core
- Desarrollar otros *frameworks* usados para el desarrollo de aplicaciones en Android
- Implementar librerias con gradle
- Establecer los *frameworks* para conectar con una API
- Generar Conocimiento especializado sobre Arquitecturas como MVP
- Concretar pros y contras sobre MVP y MVVM

Módulo 12. Interfaces y *layouts* en desarrollo de aplicaciones Android

- Presentar el modelo de ciclo de vida de las vistas en Android
- Examinar los atributos más importantes de un diseño visual (*layout*)
- Analizar los diseños (*layout*) disponibles
- Generar un diseño (*layout*) reusable
- Determinar cómo usar recursos alternativos
- Identificar las diferencias de uso de estos componentes frente a otros sistemas de programación
- Establecer la potencialidad y uso del fichero *AndroidManifest.xml*

Módulo 13. Lenguaje de programación en aplicaciones Android. Kotlin

- Desarrollar el lenguaje de programación Kotlin
- Compilar las características y capacidades diferenciales del Lenguaje Kotlin
- Examinar el modelo de ejecución básico de un programa en Kotlin
- Analizar la sintaxis del lenguaje y la estructura de un programa en Kotlin
- Concretar el modelo de tipos y variables en Kotlin
- Establecer las distintas formas de gestión de flujo del código en Kotlin
- Determinar el modelo de clases, colecciones y objetos en Kotlin
- Generar conocimiento especializado sobre el modelo de herencia en Kotlin
- Concretar el modelo de gestión de excepciones y tipos nulos en Kotlin

Módulo 14. Lenguaje de programación en aplicaciones Android. Kotlin avanzado. Genericidad, programación funcional y paralelismo

- Examinar el modelo de Genericidad Covariante y Contravariante en Kotlin
- Analizar el modelo de programación funcional con *Lambdas* de Kotlin
- Concretar las funciones de orden superior de Kotlin
- Desarrollar las extensiones y los objetos acompañantes en Kotlin
- Examinar el uso del patrón *Delegation* en Kotlin
- Compilar las anotaciones y la reflexión en Kotlin
- Profundizar en el modelo de *Testing* en Kotlin
- Establecer los distintos modelos de programación asíncrona disponibles en Kotlin
- Determinar el modelo de corrutinas de Kotlin
- Compilar las distintas librerías y herramientas de utilidad del ecosistema Kotlin

Módulo 15. Gestión de datos en dispositivos Android

- Analizar las diferentes técnicas para la gestión de datos en Android
- Proponer métodos de uso óptimo de los datos en el dispositivo
- Identificar las herramientas necesarias para la optimización de los datos
- Examinar las características de JSON y XML para la gestión de datos en Android
- Evaluar las problemáticas generales de Sistemas Distribuidas aplicables al mundo de las aplicaciones de los dispositivos móviles
- Determinar el uso de la librería *Room* como abstracción para el uso de SQLite en Android y sus ventajas y desventajas
- Establecer los permisos de seguridad necesarios en la Gestión de Datos en cualquier de las técnicas que se usan en Android

Módulo 16. Herramientas de dispositivos Android

- Compilar las herramientas más avanzadas en la gestión diaria
- Evaluar herramientas de control de dispositivos Android
- Demostrar las ventajas de utilizar Android en los drones
- Concretar las funcionalidades de herramientas CRM
- Demostrar las bondades de dispositivos Android en plataformas Bancarias
- Analizar las posibilidades de plataforma IoT y Android
- Examinar herramientas de eficiencia de procesos en Android

Módulo 17. Diseño *responsive* en Android

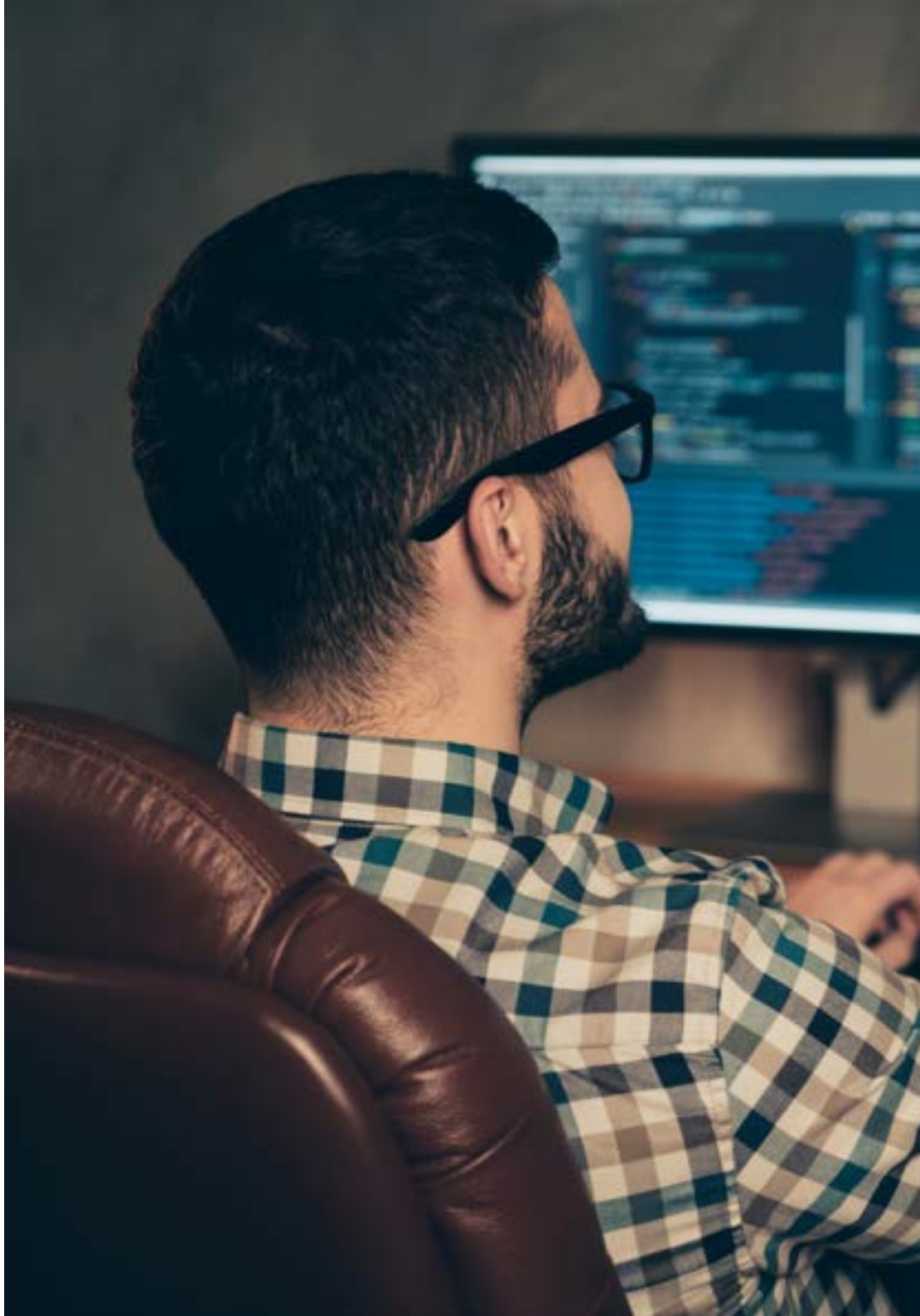
- Analizar los elementos principales de un diseño
- Definir una metodología de diseño de elementos visuales y una metodología de diseño de pantallas
- Solucionar los diversos problemas que se encuentran en el desarrollo multidispositivo
- Disponer de herramientas para generar mejores y más recursos para el desarrollo multidispositivo
- Examinar *frameworks* alternativos al desarrollo nativo responsive
- Generar una metodología propia para desarrollar aplicaciones empleando buenas prácticas para la visualización multidispositivo desde el inicio del proyecto

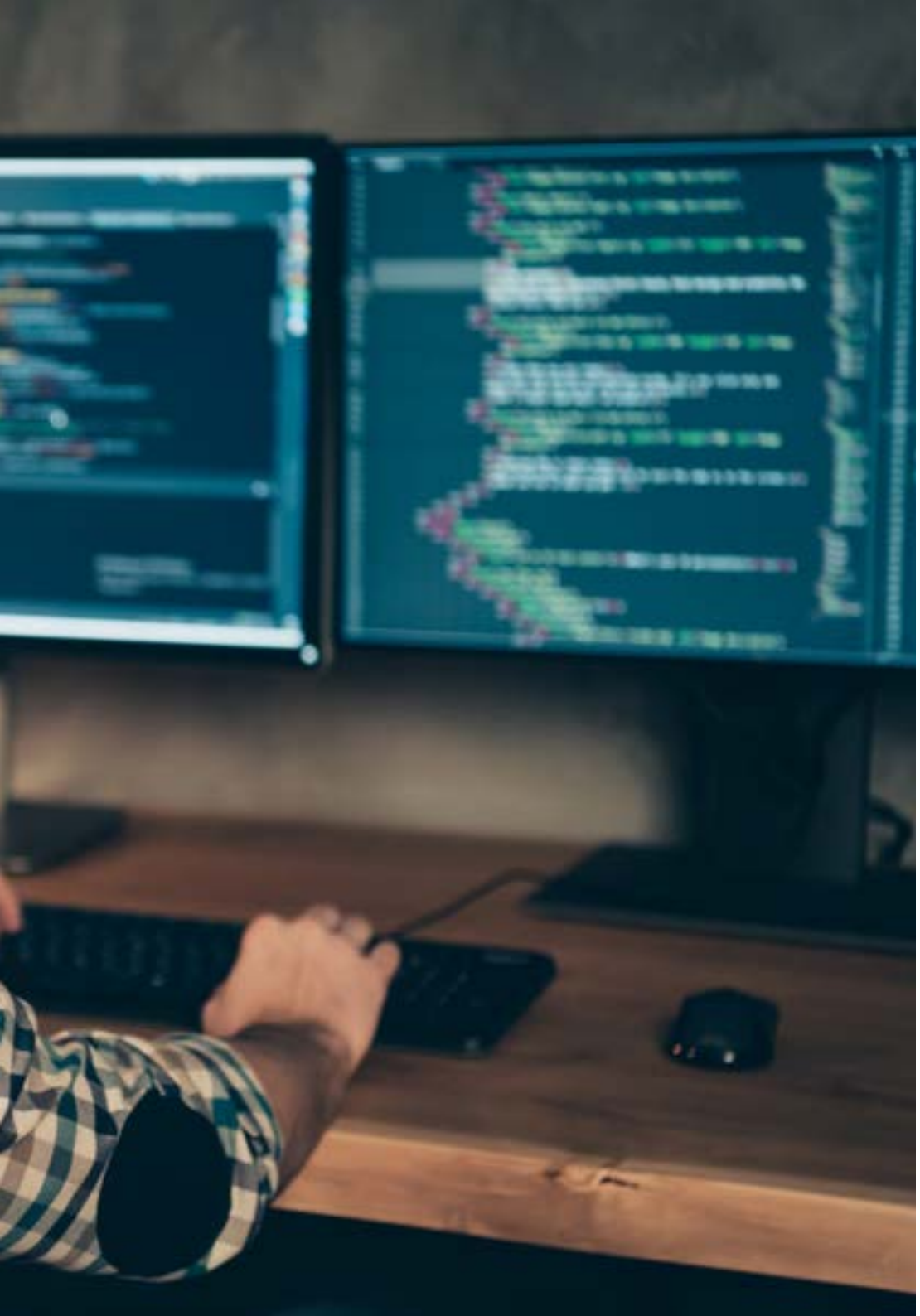
Módulo 18. Marketing en aplicaciones Android

- Analizar las nuevas metodologías centradas en el usuario
- Determinar cómo la Inteligencia Artificial ha llevado la CX al siguiente nivel
- Establecer la importancia de la accesibilidad y movilidad
- Desarrollar las técnicas más avanzadas de analítica de sesión y comportamiento
- Concretar objetivos de micropersonalización durante el *journey* de usuario
- Compilar las nuevas metodologías propias de un entorno cambiante y vivo
- Proponer técnicas para la creación de prototipos

Módulo 19. Ciclo de vida en aplicaciones Android. Cloud, playstore y versionado

- Concretar los beneficios de adoptar un modelo automatizado de despliegue de versiones
- Establecer las diferencias entre integración continua, entrega y despliegue continuos
- Definir las principales características de DevOps
- Evaluar algunas de las herramientas fundamentales para implementar pipelines de CI/CD
- Concretar los factores esenciales para desarrollar aplicaciones preparadas para soportar los procesos CI/CD
- Examinar las Tecnologías de Contenedores como pilar fundamental del CI/CD
- Identificar prácticas, casos de uso, tecnologías y herramientas del ecosistema de CI/CD





Adquirirás conocimiento especializado sobre el ciclo de vida de las aplicaciones Android con el fin de trabajar de manera más eficiente como desarrollador"

03

Competencias

La industria tecnológica demanda constantemente perfiles profesionales multitareas, lo que implica la capacitación constante para desempeñarse en un mercado con una vertiginosa evolución. Este Grand Master capacita a los informáticos con conocimientos especializados, puestos al día sobre los diferentes componentes de tecnología de los dispositivos móviles para desarrollar aplicaciones. El egresado conocerá a fondo los mejores *Hardwares* y *Softwares* imprescindibles para cualquier profesional dedicado a este sector. De esta manera, el informático estará al día en los conocimientos necesarios para ser autónomo a la hora de desarrollar una aplicación para dispositivos Android desde el diseño hasta su puesta en producción.



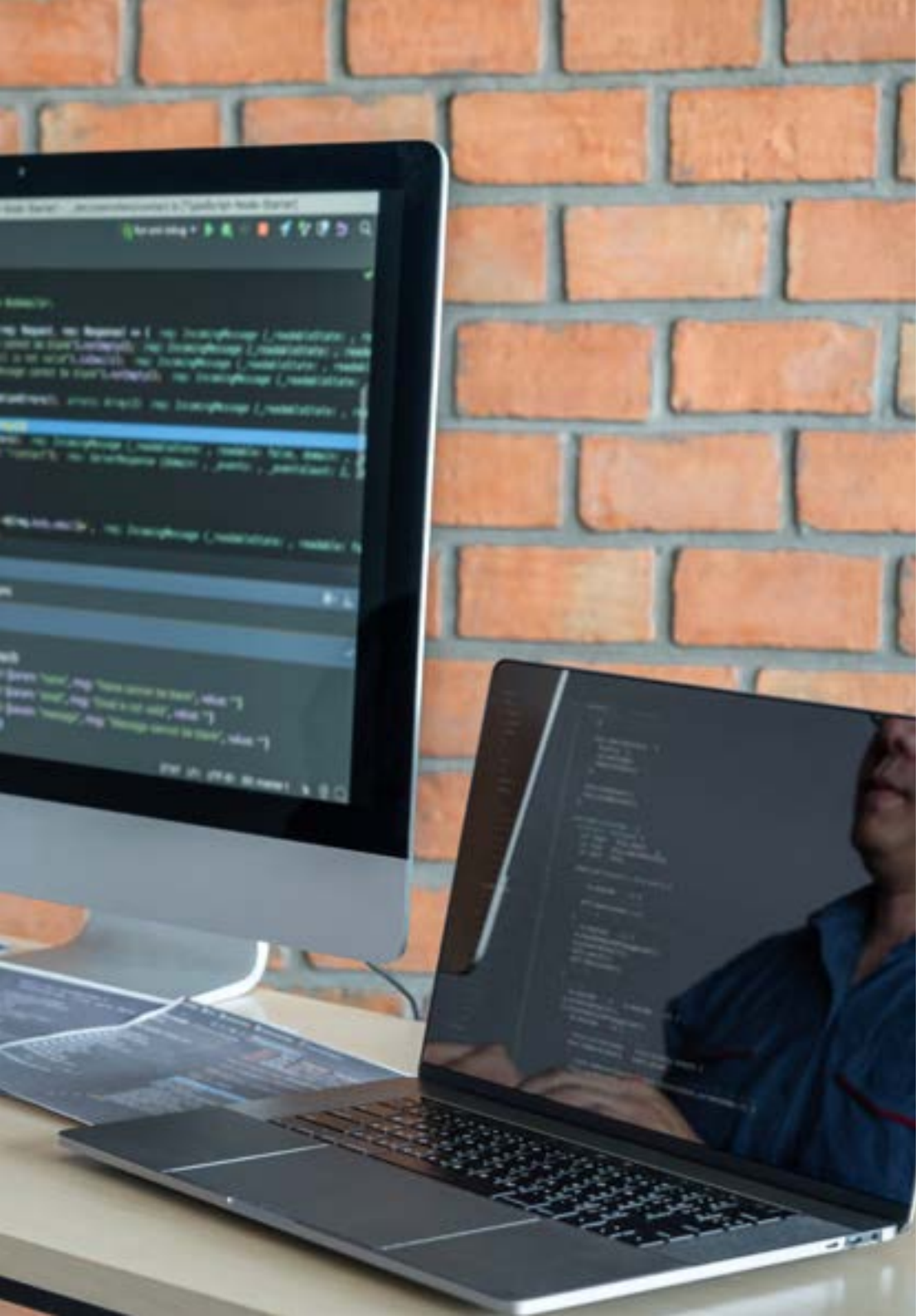


Con esta titulación impulsarás tu carrera al obtener el conocimiento práctico para planificar y dirigir proyectos con tecnologías móviles"

Competencias generales

- Desarrollar aplicaciones móviles para dispositivos móviles tanto Android como iOS
- Potenciar los *skills* necesarios en todos los puntos del ciclo de vida de Desarrollo del Software
- Analizar las diferentes metodologías de Programación para los diferentes dispositivos y casos de uso
- Dominar conocimientos especializados sobre los diferentes componentes de tecnología de los dispositivos móviles
- Comprender los entornos de trabajo para los diferentes lenguajes y estilos de programación de aplicaciones móviles
- Realizar de forma eficiente las aplicaciones desde el punto de vista de la experiencia de usuario para que sea realmente funcional y atractiva
- Dominar los elementos estructurales de un Sistema Android
- Analizar los diversos *Frameworks* para ser usados por múltiples Arquitecturas
- Definir las propiedades de una interfaz desde un punto de vista de usabilidad
- Desarrollar conocimiento especializado sobre el lenguaje de programación Kotlin y el contexto de su aparición
- Gestionar el *Content Provider* para en el intercambio de datos y su seguridad
- Abordar las herramientas empresariales en Android: su eficiencia, utilidad
- Analizar los elementos de un diseño responsive
- Explorar las nuevas metodologías centradas en el usuario
- Ajustar el Sistema de Pipelines Automatizados al Ecosistema Android





Competencias específicas

- Analizar los Procesos del Desarrollo de Software desde el punto de vista tradicional y ágil
- Desarrollar las consideraciones generales sobre los dispositivos móviles
- Dominar los conceptos clave de la Programación y el Internet, la Web y su funcionamiento
- Comparar el modelo de desarrollo nativo y el modelo de desarrollo web multiplataforma basado en apps híbridas
- Determinar cómo usar las bases de datos en las aplicaciones móviles
- Publicar una aplicación en *Play store*
- Determinar las diferentes etapas de un ciclo de integración continua
- Comprender los principios de la cultura centrada en el usuario y cómo esta genera una nueva posición para los profesionales de la experiencia de cliente
- Atender a los problemas de seguridad existentes en los dispositivos móviles
- Utilizar recursos alternativos analizando los diseños (*layout*) disponibles
- Desarrollar las distintas formas de gestión de flujo del código en Kotlin y sus capacidades
- Mejorar la productividad de las aplicaciones a través de las capacidades diferenciales del Lenguaje Klotin
- Desarrollar las extensiones y los objetos acompañantes en Kotlin
- Usar las bases de datos o servicios en red para crear ficheros en diferentes formatos
- Manejar la librería *Room* como abstracción para el uso de SQLite en Android
- Evaluar herramientas de control de dispositivos Android, analizando las posibilidades de plataforma IoT y Android
- Adoptar el diseño *responsive* para brindar experiencias de uso más satisfactorias
- Combinar Técnicas de Prototipado y *Wireframing* con las nuevas tecnologías disruptivas como son la Inteligencia Artificial y el Internet de las Cosas

04

Dirección del curso

Este Grand Master en Desarrollo de Aplicaciones Móviles, Experto en Android está dirigido por docentes de amplio conocimiento y experiencia en las nuevas tecnologías, arquitectura de soluciones e infraestructura digital, expertos en Programación Android y Desarrolladores de Aplicaciones. De esta manera, ofrecen al informático una garantía de calidad del contenido seleccionado para esta titulación. Apostando por la optimización del proceso de aprendizaje de los profesionales que buscan en este espacio la contribución que necesitan para su éxito profesional.





Compilar todas las experiencias de informáticos e ingenieros expertos en Desarrollo de Aplicaciones, añadirá valor a tus propuestas y te harán más atractivo frente el mercado laboral"

Dirección



D. Olalla Bonal, Martín

- *Blockchain Technical Specialist* en IBM SPGI
- Técnico en Electrónica Digital
- Arquitecto *Blockchain*
- Arquitecto de Infraestructura en Banca
- Formación *Hyperledger Fabric* a empresas
- Formación *Blockchain* orientado a negocio en empresas
- Gestión de proyectos y puesta en producción de soluciones
- Más de 25 años de experiencia en el sector IT

Profesores

D. Gozalo Fernández, Juan Luis

- Ingeniero informático
- Profesor Asociado en Expertos Universitarios DevOps y Desarrollo de Aplicación *Blockchain* en la Universidad Internacional de La Rioja
- Exdirector Blockchain DevOps en Alastria
- Director de Tecnología Nivel de Servicio en Santander España
- Director Desarrollo aplicación móvil Tinkerlink en Cronos Telecom
- Director Tecnología Gestión de Servicio IT en Barclays Bank España
- Licenciado en Ingeniería Superior de Informática en la UNED
- Especialización en Deep Learning en DeepLearning.ai

D. Gómez Rodríguez, Antonio

- Ingeniero de soluciones Cloud en Oracle
- Director de Proyectos en importantes grupos empresariales como Sopra Group y Everis
- Jefe de Proyectos en "Empresa pública de Gestión de Programas Culturales en la Consejería de Cultura de Andalucía
- Licenciado en Ingeniería Superior de Telecomunicación por la Universidad Politécnica de Cataluña
- Postgrado en Tecnologías y Sistemas de Información, Instituto Catalán de Tecnología
- E-Business Master, Escuela de Negocios La Salle
- Analista de Sistemas de Información. Sopra Group

D. Guerrero Díaz-Pintado, Arturo

- Ingeniero de redes de I+D en Telefónica
- Ingeniero Técnico de Preventa a través de *Watson Customer Engagement* portfolio (soluciones de Marketing y *Customer Experience*) dentro de España, Portugal, Grecia e Israel en IBM
- Consultor de servicios profesionales trabajando con organizaciones de vanguardia en Europa, Oriente Medio y América Latina desde IBM
- Licenciado en Ingeniería Superior de Telecomunicación por la Universidad de Alcalá y la *Danish Technical University*
- Ha realizado destacas colaboraciones en reconocidas universidades y centros de educación superior en materias relacionadas con la tecnología como Inteligencia Artificial, *Internet of Things, Cloud, Customer Experience y Digital Transformation*

D. Villot Guisán, Pablo

- Arquitecto Cloud, de Soluciones Exponenciales y Subject Mater Expert *Blockchain* en KPMG
- Arquitecto Cloud, de Soluciones Exponenciales y Subject Mater Expert *Blockchain* y de Integración en Everis
- Desarrollador y Responsable Técnico de aplicaciones web y de escritorio pesado para el área de Logística Comercial de Inditex, Connectis
- Certificado Microsoft en MSCA: *Cloud Platform*
- Licenciado en Ingeniería Informática por la Universidad de La Coruña

D. Pérez Rico, Javier

- Líder técnico Android en Nologis
- Líder técnico Android en Seekle
- Programador Android en Gowex-Ideup
- Programador Junior Android en Tecnocom
- Ponente del II Simposio iTest, E@tic2011
- Licenciado en Ingeniería en Técnica Informática de sistemas por la Universidad Complutense de Madrid
- Máster en Investigación por la Universidad Complutense de Madrid

D. Noguera Rodríguez, Pablo

- Desarrollador Experto en Java: JSE, JEE y Android - Ilabora Formación
- Programación de Aplicaciones en Android - EOI – Madrid
- Desarrollador de Apps Nativas (iOS & Android)- Starman Aviation (Aviaze App)
- Desarrollador de Apps Nativas (iOS) - Stef (Mtrack App)
- Desarrollador de Apps Nativas (iOS & Android) - Bitnovo (Bitnovo App)

D. Gutiérrez Blanco, Víctor

- Actual iOS Team Leader en DriveSmart
- iOS Team Leader en Keytree Spain
- iOS Leader en SeekleLabs
- iOS & Python Developer en Web Partners
- Full-Stack Developer en Let's Health
- Front-End Lead Developer en Indra y en Intelygenz
- Licenciado en Ingeniería Informática por la Universidad Pontificia de Salamanca Campus Madrid





D. Frias Favero, Pedro Luis

- CTO - Swearit Technologies
- COO - Key Identification
- Full Stack Developer - Ironhack
- Licenciado en Ingeniería Industrial por la universidad Yacambu
- Experto en *Blockchain* y Aplicaciones Descentralizadas por la Universidad de Alcalá

D. Jiménez Pérez, Carlos

- Desarrollador Android Senior en OnTheSpot - Telefónica Tech
- Profesor Adjunto en Universidad Carlos III de Madrid
- Licenciado en Ingeniería en Automática y Electrónica
- Máster en Ingeniería de Sistemas Electrónicos y Aplicaciones

05

Estructura y contenido

Con el objetivo de que el informático tenga a su alcance el mejor contenido para el desarrollo con éxito de aplicaciones móviles, TECH se ha dispuesto de un equipo de profesionales de la programación, desarrollo de aplicaciones y nuevas tecnologías que han seleccionado un contenido exclusivo y que acompañarán al informático en el proceso de estudio durante los módulos a desarrollarse en 24 meses. Desde una plataforma online, el egresado podrá acceder a contenido interactivo, distribuido en videos, gráficos, ejemplos basados en problemas, material audiovisual y contando con una amplísima comunidad para debatir las cuestiones planteadas.





Contarás con el mejor contenido audiovisual con ejercicios prácticos y formatos interactivos para que tu proceso de aprendizaje sea de forma ágil y eficiente"

Módulo 1. Metodologías de programación en Desarrollo de Aplicaciones para dispositivos móviles

1.1. Procesos del Desarrollo de Software
- 1.1.1. *Waterfall*
- 1.1.2. *Spiral*
- 1.1.3. RUP
- 1.1.4. V-Model

1.2. Procesos del Desarrollo de Software Ágiles
- 1.2.1. *Scrum*
- 1.2.2. XP
- 1.2.3. Kanban

1.3. Lenguaje Unificado de Modelado (UML)
- 1.3.1. UML
- 1.3.2. Tipos de modelado
- 1.3.3. Bloques básicos de UML

1.4. Diagramas UML de Comportamiento
- 1.4.1. *Activity diagram*
- 1.4.2. *Use case diagram*
- 1.4.3. *Interaction overview diagram*
- 1.4.4. *Timing diagram*
- 1.4.5. *State machine diagram*
- 1.4.6. *Communication diagram*
- 1.4.7. *Sequence diagram*

1.5. Diagramas UML Estructurales
- 1.5.1. *Class diagram*
- 1.5.2. *Object diagram*
- 1.5.3. *Component diagram*
- 1.5.4. *Composite structure diagram*
- 1.5.5. *Deployment diagram*

1.6. Patrones de Diseño creacionales
- 1.6.1. *Singleton*
- 1.6.2. *Prototype*
- 1.6.3. *Builder*
- 1.6.4. *Factory*
- 1.6.5. *Abstract factory*

1.7. Patrones de Diseño estructurales
- 1.7.1. *Decorator*
- 1.7.2. *Facade*
- 1.7.3. *Adapter*
- 1.7.4. *Bridge*
- 1.7.5. *Composite*
- 1.7.6. *Flyweight*
- 1.7.7. *Proxy*

1.8. Patrones de comportamiento
- 1.8.1. *Chain of responsibility*
- 1.8.2. *Command*
- 1.8.3. *Iterator*
- 1.8.4. *Mediator*
- 1.8.5. *Memento*
- 1.8.6. *Observer*
- 1.8.7. *State*
- 1.8.8. *Strategy*
- 1.8.9. *Template method*
- 1.8.10. *Visitor*

1.9. *Testing*
- 1.9.1. Pruebas unitarias
- 1.9.2. Pruebas de integración
- 1.9.3. Técnicas de caja blanca
- 1.9.4. Técnicas de caja negra



1.10. Calidad
- 1.10.1. ISO
- 1.10.2. ITIL
- 1.10.3. COBIT
- 1.10.4. PMP

Módulo 2. Tecnologías en Desarrollo de Aplicaciones para dispositivos móviles

2.1. Dispositivos móviles
- 2.1.1. Dispositivo móviles
- 2.1.2. Infraestructura de un dispositivo móvil
- 2.1.3. Fabricantes de hardware
- 2.1.4. Desarrolladores de software
- 2.1.5. Proveedores de servicios
- 2.1.6. Proveedores de plataforma
- 2.1.7. Principales plataformas

2.2. Componentes físicos de los dispositivos móviles
- 2.2.1. Almacenamiento
  - 2.2.1.1. Inmutable
  - 2.2.1.2. Mutable
  - 2.2.1.3. Temporal
  - 2.2.1.4. Externo
- 2.2.2. Presentadores
  - 2.2.2.1. Pantallas, altavoces, respuestas hápticas
- 2.2.3. Métodos de entrada
  - 2.2.3.1. Botones/teclados
  - 2.2.3.2. Pantallas
  - 2.2.3.3. Micrófonos
  - 2.2.3.4. Sensores de movimiento

2.2.4. Fuentes de energía
2.2.4.1. Las fuentes de energía
2.2.4.2. Uso adaptativo de recursos
2.2.4.3. Programación eficiente
2.2.4.4. Desarrollo sostenible
2.3. Procesadores
2.3.1. Procesador central
2.3.2. Otros procesadores abstraídos
2.3.3. Procesadores de inteligencia artificial
2.4. Transmisores de información
2.4.1. Largo alcance
2.4.2. Medio alcance
2.4.3. Corto alcance
2.4.4. Ultra corto alcance
2.5. Sensores
2.5.1. Internos al dispositivo
2.5.2. Ambientales
2.5.3. Médicos
2.6. Componentes lógicos
2.6.1. Inmutables
2.6.2. Mutables por el fabricante
2.6.3. A disposición del usuario
2.7. Categorización
2.7.1. Portátiles
2.7.2. Teléfonos inteligentes
2.7.2.1. Tabletas
2.7.2.2. Dispositivos multimedia
2.7.2.3. Complementos inteligentes
2.7.3. Asistentes robotizados
2.8. Modos de operación
2.8.1. Desconectado
2.8.2. Conectado
2.8.3. Siempre disponible
2.8.4. Punto a punto
2.9. Interacciones
2.9.1. Interacciones mediadas por el usuario
2.9.2. Interacciones mediadas por el proveedor
2.9.3. Interacciones mediadas por el dispositivo
2.9.4. Interacciones mediadas por el entorno
2.10. Seguridad
2.10.1. Medidas implementadas por el fabricante
2.10.2. Medidas implementadas por proveedores
2.10.3. Seguridad aplicada por el usuario
2.10.4. Privacidad

Módulo 3. Herramientas de trabajo en Desarrollo de Aplicaciones para dispositivos móviles

3.1. Entorno y herramientas para el Desarrollo de Aplicaciones para dispositivos móviles
3.1.1. Preparación del Entorno para Mac OS
3.1.2. Preparación del Entorno para Linux
3.1.3. Preparación del Entorno para Windows
3.2. Línea de Comando
3.2.1. Línea de Comando
3.2.2. Emuladores
3.2.3. Intérprete de comandos
3.2.4. Creación de carpetas
3.2.5. Creación de archivos
3.2.6. Navegación

3.2.7. Gestión de archivos y carpetas utilizando la interfaz de línea de comandos
3.2.8. Permisos
3.2.9. SSH
3.2.10. Lista de comandos
3.3. Repositorio de Software. Git
3.3.1. Sistema de control de versiones
3.3.2. Git
3.3.3. Configuración
3.3.4. Repositorio
3.3.5. Ramas
3.3.6. Gestión de ramas
3.3.7. Flujos de trabajo
3.3.8. Merge
3.3.9. Comandos
3.4. Servicio Web de control de versiones
3.4.1. Repositorios remotos
3.4.2. Configuración
3.4.3. Autenticación
3.4.4. Bifurcación de Software Fork
3.4.5. Comando de Git Clone
3.4.6. Repositorios
3.4.7. *Github pages*
3.5. Herramientas Avanzadas de Desarrollo para Aplicaciones en dispositivos móviles
3.5.1. *Postman*
3.5.2. *Visual studio code*
3.5.3. GUI para bases de datos
3.5.4. *Hosting*
3.5.5. Herramientas complementarias para el Desarrollo
3.6. Web desde el Prisma de Desarrollo de Aplicaciones para dispositivos móviles
3.6.2. Protocolos
3.6.3. Proveedor de servicios de internet
3.6.4. Direcciones IP
3.6.5. Servicios de Nombres DNS
3.7. Programación en el Desarrollo de Aplicaciones para dispositivos móviles
3.7.1. Programación en el Desarrollo de Aplicaciones para dispositivos móviles
3.7.2. Paradigmas de la programación
3.7.3. Lenguajes de programación
3.8. Componentes de Desarrollo de Aplicaciones para dispositivos móviles
3.8.1. Variables y constantes
3.8.2. Tipos
3.8.3. Operadores
3.8.4. Declaraciones
3.8.5. Bucles
3.8.6. Funciones y objetos
3.9. Estructura de datos
3.9.1. Estructura de datos
3.9.2. Tipos de estructura lineales
3.9.3. Tipos de estructura funcionales
3.9.4. Tipos de estructuras de árbol
3.10. Algoritmos
3.10.1. Algoritmos en Programación. Divide y vencerás
3.10.2. Algoritmos voraces
3.10.3. Programación dinámica

Módulo 4. Desarrollo web multiplataforma orientado a dispositivos móviles

4.1. Desarrollo Web Multiplataforma
- 4.1.1. Desarrollo Web Multiplataforma
- 4.1.2. Apps Híbridas vs. Apps Nativas
- 4.1.3. Tecnologías para crear Apps Híbridas

4.2. *Progressive Web Apps* (PWA)
- 4.2.1. *Progressive Web Apps* (PWA)
- 4.2.2. *Progressive Web Apps* (PWA). Características
- 4.2.3. *Progressive Web Apps* (PWA). Construcción
- 4.2.4. *Progressive Web Apps* (PWA). Limitaciones

4.3. *Framework Ionic*
- 4.3.1. *Framework Ionic*. Análisis
- 4.3.2. *Framework Ionic*. Características
- 4.3.3. Construcción de una Apps con *Ionic*

4.4. *Frameworks* de desarrollo Web
- 4.4.1. Análisis de *Framework* en Desarrollo Web
- 4.4.2. *Frameworks* de Desarrollo Web
- 4.4.3. Comparativa de *Frameworks* web

4.5. *Framework* Angular
- 4.5.1. *Framework* Angular
- 4.5.2. Uso de Angular en el Desarrollo de Aplicaciones Multiplataforma
- 4.5.3. Angular + *Ionic*
- 4.5.4. Construcción de Apps en Angular

4.6. Biblioteca de Desarrollo *React*
- 4.6.1. Biblioteca de *JavaScript React*
- 4.6.2. Biblioteca de *JavaScript React*. Uso
- 4.6.3. *React Native*
- 4.6.4. *React* + *Ionic*
- 4.6.5. Construcción de Apps en *React*

4.7. *Framework* de Desarrollo Vue
- 4.7.1. *Framework* de Desarrollo Vue
- 4.7.2. *Framework* de Desarrollo Vue . Uso
- 4.7.3. Vue + *Ionic*
- 4.7.4. Construcción de Apps en Vue

4.8. *Frameworks* de Desarrollo Electron
- 4.8.1. *Frameworks* de Desarrollo Electron
- 4.8.2. *Frameworks* de Desarrollo Electron. Uso
- 4.8.3. Desplegando nuestras Apps también en escritorio

4.9. Herramienta de Desarrollo de dispositivos móviles *Flutter*
- 4.9.1. Herramienta de Desarrollo de dispositivos móviles *Flutter*
- 4.9.2. Uso de SDK *Flutter*
- 4.9.3. Construcción de Apps en *Flutter*

4.10. Herramientas de Desarrollo para dispositivos móviles. Comparativa
- 4.10.1. Herramientas para el Desarrollo de Aplicaciones en dispositivos móviles
- 4.10.2. *Flutter* vs. Ionic
- 4.10.3. Selección del *Stack* más adecuado para la creación de una App

Módulo 5. Base de datos para el Desarrollo de Aplicaciones para dispositivos móviles

5.1. Bases de Datos en dispositivos móviles
- 5.1.1. Persistencia de datos en el desarrollo de aplicaciones móviles
- 5.1.2. Capacidades de bases de datos para aplicaciones móviles
- 5.1.3. SQL. *Structured Query Language*

5.2. Elección de la base de datos para las aplicaciones móviles
- 5.2.1. Análisis de las Aplicaciones en dispositivos móviles en función de la base de datos
- 5.2.2. Categorías de bases de datos
- 5.2.3. Panorama de base de datos

5.3. Desarrollo con SQLite
- 5.3.1. Base de datos SQLite
- 5.3.2. Despliegue del odelo
- 5.3.3. Conexión a SQLite

5.4. Desarrollo con Oracle Berkeley DB
- 5.4.1. Base de datos Berkeley DB
- 5.4.2. Despliegue del modelo
- 5.4.3. Conexión a Berkeley DB

5.5. Desarrollo con Realm
- 5.5.1. Capacidades de Realm
- 5.5.2. Creación de base de datos en Realm
- 5.5.3. Conexión a Realm

5.6. Desarrollo con CouchDB Lite
- 5.6.1. Base de datos CouchDB Lite
- 5.6.2. Creación de base de datos con CouchDB Lite
- 5.6.3. Conexión con CouchDB Lite

5.7. Desarrollo con base de datos centralizadas MySQL
- 5.7.1. Bases de Datos MySQL
- 5.7.2. Despliegue de modelo relacional con MySQL
- 5.7.3. Conexión a MySQL

5.8. Desarrollos centralizados. Oracle, MS SQL Server, MongoDB
- 5.8.1. Desarrollo con Oracle
- 5.8.2. Desarrollo con MS SQL Server
- 5.8.3. Desarrollo con MongoDB

5.9. Datos de tipo Grafos
- 5.9.1. Base de datos orientada a Grafos
- 5.9.2. Creación de base de datos con Neo4j
- 5.9.3. Conexión con Neo4j desde App Móvil

5.10. Entornos con capacidades de almacenamiento
- 5.10.1. Desarrollos con Firebase
- 5.10.2. Desarrollos con Core Data
- 5.10.3. Desarrollo con Visual Builder Cloud Service

Módulo 6. Desarrollo de Aplicaciones para sistemas iOS

6.1. Entorno de Desarrollo Xcode
- 6.1.1. Creación de un proyecto
- 6.1.2. Configuración de un Emulador para compilar
- 6.1.3. Configuración de un teléfono físico para compilar

6.2. Lenguaje de Programación Swift
- 6.2.1. Swift I: lenguaje de programación
- 6.2.2. Swift II: funciones y loops
- 6.2.3. Swift III: lambdas y structs

6.3. Librerías y Cocoa Pods
- 6.3.1. Pods: instalación
- 6.3.2. Configuración de Cocoa Pods
- 6.3.3. Estructura de Cocoa Pods

6.4. Librerías: Api, Base de datos y R.swift
- 6.4.1. Alamofire
- 6.4.2. Bases de datos SQL con GRDB
- 6.4.3. R.swift

6.5. Diseño de pantallas
- 6.5.1. Diseño con Storyboard
- 6.5.2. Diseño responsive
- 6.5.3. Diseño de vistas por código y SwiftUI

6.6. Montaje de una vista
- 6.6.1. UIViewController y su ciclo de vida
- 6.6.2. Interacción entre diferentes pantallas
- 6.6.3. Tipos de transiciones y modales

6.7. Sensores y localización
- 6.7.1. Acceso a los sensores
- 6.7.2. Acceso a la localización en primer plano
- 6.7.2. Acceso a la localización en segundo plano

6.8. Arquitecturas
- 6.8.1. MVP
- 6.8.2. VIPER
- 6.8.3. Arquitectura de Desarrollo IOS

6.9. Monetización y *Analytics*
- 6.9.1. *Firebase analytics*
- 6.9.2. *Firebase crashlytics*
- 6.9.3. Monetización y anuncios con ADMob de Google

6.10. App store y versionado
- 6.10.1. Configuración de una cuenta en la App Store
- 6.10.2. Versiones de prueba (*test flight*)
- 6.10.3. Lanzamiento a producción

Módulo 7. Despliegues en integración continua para dispositivos móviles

7.1. DevSecOps
- 7.1.1. DevSecOps. Uso
- 7.1.2. Analizadores estáticos
- 7.1.3. Pruebas de seguridad de análisis dinámico

7.2. Monitoreo continuo
- 7.2.1. Monitoreo Continuo
- 7.2.2. Monitoreo Continuo. Análisis y Ventajas
- 7.2.3. Monitoreo Continuo. Plataformas

7.3. Implementación
- 7.3.1. Implementación en máquina local
- 7.3.2. Implementación en máquina compartida
- 7.3.3. Implementación en base servicios en la nube
- 7.3.4. Gestión de la configuración



Módulo 8. *User experience* en dispositivos móviles

8.1. *User experience*
- 8.1.1. *Client experience*
- 8.1.2. *Client experience*. Requisitos
- 8.1.3. Bidireccionalidad con el cliente

8.2. *Client experience*. Objetivos y equipos
- 8.2.1. *Client experience*. Objetivos y equipos
- 8.2.2. Procesos iterativos
- 8.2.3. Información necesaria

8.3. Micro-interacciones
- 8.3.1. Relación *End-to-end*
- 8.3.2. Interacciones
- 8.3.3. Omnicanalidad

8.4. Comportamiento de usuario
- 8.4.1. Diseño de los cimientos
- 8.4.2. Analítica web y de sesión
- 8.4.3. Expertos en Analítica

8.5. Estado del Arte de la Tecnología
- 8.5.1. *Machine Learning*
- 8.5.2. Cadena de bloques
- 8.5.3. Internet de las Cosas

8.6. Componentes técnicos
- 8.6.1. Componentes técnicos
- 8.6.2. Componentes avanzados: dispositivos
- 8.6.3. Componentes avanzados: distintos perfiles

8.7. Usabilidad
- 8.7.1. Heurísticos de Nielsen
- 8.7.2. Pruebas de usuario
- 8.7.3. Usabilidad. Errores

8.8. Técnicas de UX. *User experience*
- 8.8.1. Reglas
- 8.8.2. *Prototyping*
- 8.8.3. Herramientas de *Low-Code*

8.9. Estrategia Visual
- 8.9.1. Diseñador de *User Interface*
- 8.9.2. Trabajo de *User Interface* en la Web
- 8.9.3. Trabajo de *User Interface* en Aplicaciones

8.10. *Developer Frameworks*
- 8.10.1. *Frameworks* de CX
- 8.10.2. *Frameworks* de UX
- 8.10.3. *Frameworks* de UI

Módulo 9. Seguridad en dispositivos móviles

9.1. Arquitectura de seguridad de dispositivos móviles
- 9.1.1. Seguridad física de los dispositivos
- 9.1.2. Seguridad de sistema operativo
- 9.1.3. Seguridad de aplicación
- 9.1.4. Seguridad de datos
- 9.1.5. Seguridad de comunicaciones
- 9.1.6. Seguridad de los dispositivos de empresa

9.2. Securización del Hardware móvil
- 9.2.1. Dispositivos móviles
- 9.2.2. Dispositivos wearables
- 9.2.3. Automóviles
- 9.2.4. Dispositivos IOT
- 9.2.5. Dispositivos TV

9.3. Securización del Sistema Operativo
- 9.3.1. Dispositivos móviles Android
- 9.3.2. Dispositivos móviles Apple IOS
- 9.3.3. Otros dispositivos móviles existentes: Blackberry, etc.

9.3.4. Dispositivos wearables
9.3.5. Sistemas operativos de automóviles
9.3.6. Dispositivos móviles en *Internet of Things* (IoT)
9.3.7. Dispositivos *SmartTV*
9.4. Securización de las aplicaciones móviles
9.4.1. Dispositivos móviles Android
9.4.2. Dispositivos móviles Apple IOS
9.4.3. Otros dispositivos móviles. Blackberry
9.4.4. Dispositivos *Wearables*
9.4.5. Sistemas operativos de automóviles
9.4.6. Dispositivos móviles en *Internet of Things* (IoT)
9.4.7. Dispositivos *SmartTV*
9.5. Securización de los datos en aplicaciones móviles
9.5.1. Dispositivos móviles Android
9.5.2. Dispositivos móviles Apple IOS
9.5.3. Otros dispositivos móviles. Blackberry
9.5.4. Dispositivos *Wearables*
9.5.5. Sistemas operativos de automóviles
9.5.6. Dispositivos móviles en *Internet of Things* (IoT)
9.5.7. Dispositivos *SmartTV*
9.6. Seguridad en los *Market Places* de Móviles
9.6.1. *Google Play* de Google
9.6.2. *Play Store* de Apple
9.6.3. Otros *Market Places*
9.6.4. *Rooting* de dispositivos móviles
9.7. Soluciones de Seguridad Multiplataforma
9.7.1. *Mobile Device Management* (MDM) único
9.7.2. Tipos de soluciones existentes en el mercado
9.7.3. Securización de dispositivos usando un MDM (*Master Data Management*)
9.8. Desarrollo seguro de aplicaciones móviles
9.8.1. Uso de patrones para desarrollo seguro
9.8.2. Gestión de pruebas integradas de seguridad
9.8.3. Despliegue seguro de aplicaciones
9.9. Gestión de permisos en dispositivos móviles
9.9.1. Sistema de permisos
9.9.2. Ejecución de procesos en el núcleo
9.9.3. Hilos de ejecución y eventos
9.10. Recomendaciones de seguridad para dispositivos móviles
9.10.1. Recomendaciones de NSA sobre dispositivos móviles
9.10.2. Recomendaciones de INCIBE sobre dispositivos móviles
9.10.3. ISO 27001:2013 Anexo

Módulo 10. Lenguaje de programación Android

10.1. Plataforma Android
10.1.1. Plataforma Android
10.1.2. Sistema Operativo Android
10.1.3. *Open Handset Alliance* en el desarrollo de Android
10.2. Arquitectura Android
10.2.1. Elementos arquitecturales de un Sistema Android
10.2.2. Comunicación entre elementos
10.2.3. Extensibilidad de la arquitectura Android
10.2.4. Gestión de recursos máquina: Batería y Memoria
10.2.5. Emuladores Android
10.3. Núcleo Linux de Android
10.3.1. Composición del núcleo
10.3.2. Elementos estructurales del núcleo
10.3.3. La Máquina Virtual Dalvik
10.3.4. La Máquina Virtual Android RunTime (ART)

10.4. Librerías Nativas de Android
- 10.4.1. Librerías Nativas de Android
- 10.4.2. Librerías de compatibilidad (*Support Library*)
- 10.4.3. Librerías nativas y extensibilidad

10.5. El Sistema de Ficheros y Datos en Android
- 10.5.1. Estructura de una aplicación Android típica
- 10.5.2. Sistema de ficheros YAFFS2 y ext4
- 10.5.3. Uso de SQLite y Room para la Gestión de Datos

10.6. Seguridad en Android
- 10.6.1. Sistema de Permisos
- 10.6.2. Firmas digitales en los *Android Application Package* (apk)
- 10.6.3. Ejecución de procesos en el núcleo
- 10.6.4. Hilos de ejecución y Eventos

10.7. Componentes estructurales de una aplicación estándar
- 10.7.1. Vista (*View*)
- 10.7.2. Actividad (*Activity*)
- 10.7.3. Fragmento (*Fragment*)
- 10.7.4. Servicio (*Service*)
- 10.7.5. Intención *(Intent*)
- 10.7.6. *Broadcasts Receiver* y *Content Provider*
- 10.7.7. Gestión de datos y preferencias de usuario

10.8. Versiones Android
- 10.8.1. Versiones Android
- 10.8.2. Despliegue de versiones Android
- 10.8.3. Dispersión de distribuciones Android
- 10.8.4. Android versus Apple IOS y otros sistemas para móviles

10.9. Android para vehículos
- 10.9.1. Android y el mundo del automóvil
- 10.9.2. Elementos estructurales en un sistema Android para automóviles
- 10.9.3. Comunicación entre dispositivos

10.10. Android en Domótica, *Wearable* y en *Internet of Things* (IoT)
- 10.10.1. El mundo conectado
- 10.10.2. Elementos estructurales en un sistema Android Domótico
- 10.10.3. Elementos de Android *Wearable*
- 10.10.4. Android en el *Internet of Things* (IoT)

Módulo 11. *Frameworks* en desarrollo de aplicaciones Android

11.1. *Frameworks* en Desarrollo de Aplicaciones Android
- 11.1.1. *Framework* en Desarrollo de Aplicaciones Android
- 11.1.2. *Frameworks*. Tipologías
- 11.1.3. Elección del *Framework* para el proyecto

11.2. Implementación de *Frameworks* en Android
- 11.2.1. Frameworks *Core* de Android para Java/Kotlin
- 11.2.2. *Jetpack compose*
- 11.2.3. *Frameworks* en otros Lenguajes

11.3. Sistemas de Gestión de Librerías en Desarrollo
- 11.3.1. *Gradle*
- 11.3.2. Automatización con *Gradle*
- 11.3.3. Herramienta del Desarrollo Maven

11.4. Código limpio
- 11.4.1. Código ordenado
- 11.4.2. Preparación de Código en Aplicaciones Android
- 11.4.3. *Bikeshedding* y priorización

11.5. Patrones de Desarrollo en Android
- 11.5.1. Categorías de patrones
- 11.5.2. Diferencias entre patrones
- 11.5.3. *Factory, Observer* y *Singleton*

11.6. MVP. Modelo, vista y presentador
- 11.6.1. MVC. Modelo, vista y controlador
- 11.6.2. Modelo, vista y presentador
- 11.6.3. Ejemplo práctico: Pokémon Battle

11.7. MVVM. Modelo, vista y *View Model*
- 11.7.1. MVC vs MVVM
- 11.7.2. Modelo, vista y *View Model*
- 11.7.3. Ejemplo Práctico: Pokemon Battle II

11.8. *Frameworks* y librerías más utilizadas en Android
- 11.8.1. Librerías de interacción con API
- 11.8.2. Librerías de conversión de datos
- 11.8.3. *Firebase* y *Firebase Analytics*

11.9. *Framework* visual de Android
- 11.9.1. Ciclo de vida de una aplicación Android
- 11.9.2. Diseño de vistas en XML
- 11.9.3. Diseño de elementos y animaciones en XML

11.10. *Frameworks* de Android en otros lenguajes
- 11.10.1. *React Native*
- 11.10.2. *Flutter*
- 11.10.3. *Ionic*

Módulo 12. Interfaces y *layouts* en desarrollo de aplicaciones Android

12.1. Ciclo de Vida de un Interfaz en Android
- 12.1.1. Ciclo de vida en Android
- 12.1.2. Relación proceso-actividades
- 12.1.3. Persistencia de estado de la aplicación
- 12.1.4. Arquitectura *Clean* aplicada a Android

12.2. Vistas en el Desarrollo de Aplicaciones Android (*Views*)
- 12.2.1. Capa de Presentación de la Arquitectura *Clean*
- 12.2.2. *Recicler View*
- 12.2.3. *Adapter View*

12.3. Diseños en el Desarrollo de Aplicaciones Android (*Layouts*)
- 12.3.1. *Layouts* en Android
- 12.3.2. *Constraint Layout*
- 12.3.3. Creación de *Layouts* usando *Android Studio Layout Editor*

12.4. Animaciones en el Desarrollo de Aplicaciones Android (*Animations*)
- 12.4.1. Iconos e imágenes
- 12.4.2. Transiciones
- 12.4.3. Diferencia entre Animación de Propiedades y Animación de Vistas

12.5. Actividades e Intenciones en el Desarrollo de Aplicaciones Android (*Activity* e *Intentions*)
- 12.5.1. Intenciones explícitas e implícitas
- 12.5.2. Barra de acciones
- 12.5.3. Comunicación entre actividades

12.6. Recursos alternativos y de Sistema (*Material Design, Cardboard,* etc.)
- 12.6.1. Material *Design* para Android
- 12.6.2. Multimedia en Android
- 12.6.3. Realidad Virtual con *Google Cardboard for Android* NDK

12.7. Estilos y Temas en el Desarrollo de Aplicaciones Android
- 12.7.1. Estilos en un proyecto Android
- 12.7.2. Temas para el proyecto Android
- 12.7.3. Reutilización de Estilos y Temas

12.8. Gráficos, Pantalla Táctil y Sensores
- 12.8.1. Trabajo con Gráficos avanzados
- 12.8.2. Gestión de dispositivos con Pantalla Táctil y con Teclado
- 12.8.3. Uso de Sensores del Dispositivo con Android

12.9. Diseños para Realidad Aumentada
- 12.9.1. Interfaces complejas usando la Cámara
- 12.9.2. Sensores de posición y GPS en la Realidad Aumentada
- 12.9.3. Presentación en pantallas no estándar
- 12.9.4. Errores y problemas comunes

12.10. Configuración avanzada de interfaz con *AndroidManifest.xml*
- 12.10.1. La potencia del fichero de manifiesto de Android
- 12.10.2. Diseño programático versus diseño declarativo
- 12.10.3. Componentes clave del archivo



Módulo 13. Lenguaje de Programación en Aplicaciones Android. Kotlin

13.1. Lenguaje de Programación Kotlin
- 13.1.1. Lenguaje de Programación Kotlin
- 13.1.2. Lenguaje de Programación Kotlin. Características
- 13.1.3. Funcionamiento de un programa en Kotlin

13.2. Programar en Kotlin
- 13.2.1. Estructura de un programa en Kotlin
- 13.2.2. Palabras reservadas y sintaxis
- 13.2.3. Escribir por consola y leer *inputs* del usuario-Hola Mundo

13.3. Tipos y Variables en Kotlin
- 13.3.1. Tipos y Variables en Kotlin
- 13.3.2. Declaración de Variables: Var vs. Val
- 13.3.3. Operadores
- 13.3.4. Conversión de tipos
- 13.3.5. *Arrays*

13.4. Control de Flujo en Kotlin
- 13.4.1. Control de Flujo
- 13.4.2. Expresiones condicionales
- 13.4.3. Bucles

13.5. Funciones en Kotlin
- 13.5.1. Funciones en Kotlin
- 13.5.2. Estructura de una función
- 13.5.3. Funciones de alcance (*Scope functions*)

13.6. Clases y Objetos en Kotlin
- 13.6.1. Clases y Objetos en Kotlin
- 13.6.2. Clases
- 13.6.3. Objetos
- 13.6.4. Constructores e inicialización de propiedades
- 13.6.5. Clases anidadas y clases internas
- 13.6.6. Clases de datos

13.7. Herencia en Kotlin
- 13.7.1. Herencia
- 13.7.2. Superclases y subclases
- 13.7.3. Sobreescritura de propiedades y funciones
- 13.7.4. Herencia vs. otros tipos de relación entre clases
- 13.7.5. Clases selladas
- 13.7.6. Enumerados

13.8. Clases abstractas e Interfaces en Kotlin
- 13.8.1. Clases abstractas e Interfaces
- 13.8.2. Clases abstractas
- 13.8.3. Interfaces
- 13.8.4. Validación y conversión de Tipos-Operadores *is, when, as*

13.9. Colecciones en Kotlin
- 13.9.1. Colecciones en Kotlin
- 13.9.2. Lista
- 13.9.3. Conjunto
- 13.9.4. Mapa

13.10. Gestión de excepciones y valores nulos en Kotlin
- 13.10.1. Gestión de Excepciones y Valores Nulos
- 13.10.2. Valor nulo, tipos *nullable* y *non-nullable*
- 13.10.3. Excepciones

Módulo 14. Lenguaje de programación en aplicaciones Android. Kotlin avanzado. Genericidad, programación funcional y paralelismo

14.1. Genericidad en Kotlin
- 14.1.1. Genericidad en Kotlin
- 14.1.2. Genericidad en colecciones, funciones, clases e interfaces
- 14.1.3. Covarianza y Contravarianza: *Out* o *In*

14.2. Funciones Lambda en Kotlin
- 14.2.1. Funciones Lambdas
- 14.2.2. Estructura de una función lambda
- 14.2.3. Uso de funciones lambda

14.3. Funciones de orden superior en Kotlin
- 14.3.1. Funciones de orden superior
- 14.3.2. Funciones de orden superior estándares de Kotlin
- 14.3.3. Enlazando llamadas a funciones

14.4. Extensiones en Kotlin
- 14.4.1. Extensiones en Kotlin
- 14.4.2. Funciones de extensión
- 14.4.3. Propiedades de extensión
- 14.4.4. Objetos acompañantes

14.5. Patrón *Delegation* en Kotlin
- 14.5.1. Patrón *Delegation*
- 14.5.2. *Delegation* en Kotlin
- 14.5.3. Propiedades delegadas

14.6. Anotaciones y Reflexión en Kotlin
- 14.6.1. Anotaciones y Reflexión
- 14.6.2. Anotaciones en Kotlin
- 14.6.3. Reflexión en Kotlin

14.7. *Testing* en Kotlin
- 14.7.1. *Testing* en Kotlin
- 14.7.2. *Frameworks* y Librerías de *Testing* en Kotlin
- 14.7.3. Kotest

14.8. Programación asíncrona en Kotlin
- 14.8.1. Programación asíncrona
- 14.8.2. Técnicas de programación asíncrona en Kotlin
- 14.8.3. Comparativa en técnicas de Programación

14.9. Corrutinas en Kotlin
- 14.9.1. Corrutinas
- 14.9.2. Canales
- 14.9.3. Contexto y *Dispatchers*
- 14.9.4. Estado compartido y concurrencia
- 14.9.5. Gestión de excepciones en corrutinas

14.10. Ecosistema Kotlin
- 14.10.1. Ecosistema Kotlin
- 14.10.2. Librerías para Kotlin
- 14.10.3. Herramientas para Kotlin

Módulo 15. Gestión de datos en dispositivos Android

15.1. Gestión de Datos. Tipología
- 15.1.1. La Gestión de Datos en dispositivos móviles
- 15.1.2. Alternativas para la Gestión de Datos en dispositivos Android
- 15.1.3. Generación de datos para trabajo por Inteligencia Artificial y Analítica de Uso
- 15.1.4. Herramientas de medición de rendimiento para una óptima gestión de datos

15.2. Gestión de preferencias de usuario
- 15.2.1. Tipos de datos involucrados en los archivos de preferencias
- 15.2.2. Gestión de las preferencias de usuario
- 15.2.3. Exportación de las preferencias. Gestión de permisos

15.3. Sistema de almacenamiento de ficheros
- 15.3.1. Clasificación de sistemas de ficheros en dispositivos móviles
- 15.3.2. Sistema interno de ficheros
- 15.3.3. Sistema externo de ficheros

15.4. Archivos JSON como almacenamiento en Android
- 15.4.1. Información no estructurada en ficheros JSON
- 15.4.2. Librerías para Gestión de Datos con JSON
- 15.4.3. Uso de JSON en Android. Recomendaciones y optimizaciones

15.5. Archivos XML como Almacenamiento en Android
- 15.5.1. Formato XML en Android
- 15.5.2. XML a través de librerías SAX
- 15.5.3. XML a través de librerías DOM

15.6. Bases de Datos SQLite
- 15.6.1. Base de datos relacional para la Gestión de Datos
- 15.6.2. Uso de la base de datos
- 15.6.3. Métodos SQLite para la gestión de datos

15.7. Uso avanzado de Bases de Datos SQLite
- 15.7.1. Recuperación de fallos usando transacciones en SQLite
- 15.7.2. Uso de caché para acelerar el acceso a datos
- 15.7.3. Base de datos en móviles

15.8. Librería Room
- 15.8.1. Arquitectura de la librería Room
- 15.8.2. Librería Room. Funcionalidad
- 15.8.3. Librería Room. Ventajas y desventajas

15.9. *Content Provider* para compartir información
- 15.9.1. *Content Provider* para compartir información
- 15.9.2. *Content Provider* en Android. Técnico de Uso
- 15.9.3. Seguridad en *Content Provider*

15.10. Recogida de Datos en Cloud de Internet
- 15.10.1. Android y Sistemas de almacenamiento en la Nube (*Cloud*)
- 15.10.2. Servicios SOAP y REST para Android
- 15.10.3. Problemática de los Sistemas distribuidos
- 15.10.4. Internet como copia de seguridad de los datos de la aplicación

Módulo 16. Herramientas de dispositivos Android

16.1. Gestión: herramientas "TO DO"
- 16.1.1. Herramientas de mercado
- 16.1.2. Herramientas de mercado. Comparativa de funcionalidades
- 16.1.3. Herramientas de gestión. Diferencias

16.2. MDM: gestión empresarial de dispositivos
- 16.2.1. Control de los dispositivos de empresa
- 16.2.2. Análisis de las principales herramientas de mercado
- 16.2.3. Elección de la herramienta

16.3. CRM: herramientas de mercado
- 16.3.1. Análisis de las herramientas de mercado con aplicación en Android
- 16.3.2. Herramientas de Mercado. Eficiencia
- 16.3.3. Herramientas de Mercado. Usos

16.4. Drones con Android
- 16.4.1. Apps de dispositivos Android para control de dron
- 16.4.2. Controles autónomos
- 16.4.3. Usos de dron en Android

16.5. Android, valor añadido en plataformas Bancarias
- 16.5.1. Android en plataformas Bancarias
- 16.5.2. Riesgos y fraudes de ciberdelincuentes
- 16.5.3. Usos de dispositivos móviles

16.6. *Brokering* en dispositivos móviles
- 16.6.1. Herramientas de mercado y su uso
- 16.6.2. Comparativa de las herramientas
- 16.6.3. Elección de herramienta para cada uso

16.7. Herramientas de entretenimiento y formación
- 16.7.1. Usos
- 16.7.2. Herramientas del mercado
- 16.7.3. Comparativas de funcionalidades entre Herramientas de Desarrollo Android

16.8. IoT Android
- 16.8.1. *Framework* y plataformas de mercado
- 16.8.2. Riesgos y consideraciones de IoT en Android
- 16.8.3. Usos de IoT en Android

16.9. Eficiencia de procesos
- 16.9.1. Análisis de herramientas de mercado para creación de Apps.
- 16.9.2. Comparativa de herramientas de creación de aplicaciones para Android
- 16.9.3. *Use Case*

16.10. Aplicaciones más descargadas en la actualidad
- 16.10.1. Herramientas más descargadas en la actualidad
- 16.10.2. Agrupación por familias
- 16.10.3. Usos principales, secundarios y comparativas con IOS

Módulo 17. Diseño *responsive* en Android

17.1. *Responsive design*
- 17.1.1. Diseño *responsive*
- 17.1.2. Usabilidad, accesibilidad y UX
- 17.1.3. Diseño *responsive*. Ventajas y desventajas

17.2. *Mobile vs Tablet vs Web vs Smartwatches*
- 17.2.1. Diferentes formatos, diferentes tamaños, diferentes necesidades
- 17.2.2. Problemas en el diseño
- 17.2.3. Adaptativo vs responsivo

17.3. Guías de estilos
- 17.3.1. Guías de Estilos. Utilidad
- 17.3.2. Material *design*
- 17.3.3. Guía de estilos propia

17.4. *Layouting* flexible
- 17.4.1. *Layouting* Flexible
- 17.4.2. *Layouting* Básico
- 17.4.3. *Layouting* en *Grid*
- 17.4.4. *Layouting* con *Relative Layout*
- 17.4.5. *Layouting* con *Constraint Layout*

17.5. Recursos flexibles
- 17.5.1. Recursos flexibles
- 17.5.2. Imágenes
- 17.5.3. 9 patch
- 17.5.4. Recursos globales

17.6. Navegación flexible
- 17.6.1. Navegación flexible
- 17.6.2. Navegación con *Activities*
- 17.6.3. Navegación con *Fragments*

17.7. Herramientas externas
- 17.7.1. Generadores automáticos
- 17.7.2. Herramientas de prototipado
- 17.7.3. Herramientas de diseño

17.8. *Debug* y *Tests*
- 17.8.1. *Debug layouts*
- 17.8.2. *Tests* automáticos
- 17.8.3. Desarrollo basado en componentes
- 17.8.4. Buenas prácticas de *Testing* y Pruebas

17.9. Alternativas a Android nativo I. *Web Pages*
- 17.9.1. Diseño en un *WebView*
- 17.9.2. *ChromeCustomTabs*
- 17.9.3. *Debug* y *Tests* en *web pages*

17.10. Alternativas a Android nativo II. Aplicaciones híbridas
- 17.10.1. *React/React native*
- 17.10.2. *Flutter*
- 17.10.3. *Ionic*
- 17.10.4. Apache Cordova

Módulo 18. Marketing en aplicaciones Android

18.1. De *Customer Service* a *Customer Experience*
- 18.1.1. *Customer Service*. Desarrollo del cliente actual
- 18.1.2. Usuario con acceso a la información. Exigencias y necesidades
- 18.1.3. La retroalimentación como fuente de conocimiento

18.2. *Customer Journey*
- 18.2.1. Camino del usuario a la conversión
- 18.2.2. Micro-segmentación
- 18.2.3. Experiencia a través de los canales

18.3. Medición de la experiencia de Usuario
- 18.3.1. Arquitectura web y móvil
- 18.3.2. Analítica de sesión como nuevo estándar
- 18.3.3. Estado del Arte de la experiencia de usuario

18.4. Marketing de Aplicaciones Android
- 18.4.1. CX+IA
- 18.4.2. CX+Blockchain
- 18.4.3. CX+IoT

18.5. Productos de CX (experiencia de cliente)
- 18.5.1. Estándares de la industria
- 18.5.2. Telepresencia
- 18.5.3. Experiencia de cliente para todos los Agentes del Desarrollo

18.6. Trabajo centrado en el usuario
- 18.6.1. Equipos
- 18.6.2. Pensamiento de diseñador
- 18.6.3. Trabajo de campo

18.7. La Ciencia sobre el usuario
- 18.7.1. La Ciencia sobre el Usuario. Reglas de oro
- 18.7.2. Iteración
- 18.7.3. Errores comunes

18.8. Prototipado y *Wireframing*
- 18.8.1. Prototipado y *Wireframing*
- 18.8.2. *Hands-on*
- 18.8.3. Nivel avanzado

18.9. Interfaces móviles
- 18.9.1. Diseño visual. Reglas
- 18.9.2. Interfaz de apps. Claves
- 18.9.3. Buenas prácticas en el desarrollo de Interfaces Móviles

18.10. Buenas prácticas en la Experiencia de Usuario. Consejos para Desarrolladores
- 18.10.1. Nivel Uno. Buenas prácticas en CX
- 18.10.2. Nivel Dos. Buenas prácticas en UX
- 18.10.3. Nivel Tres. Buenas prácticas en UI

Módulo 19. Ciclo de vida en aplicaciones Android. Cloud, Playstore y versionado

19.1. Ciclo de vida del *software*
- 19.1.1. Ciclo de vida del *software*
- 19.1.2. Metodologías Ágiles
- 19.1.3. El ciclo continuo ágil del *software*

19.2. Desarrollo del producto manual
- 19.2.1. Integración manual
- 19.2.2. Entrega manual
- 19.2.3. Despliegue manual

19.3. Integración supervisada
- 19.3.1. Integración continua
- 19.3.2. Revisión manual
- 19.3.3. Revisiones automáticas estáticas

19.4. Pruebas lógicas
- 19.4.1. Pruebas unitarias
- 19.4.2. Pruebas de integración
- 19.4.3. Pruebas de comportamiento

19.5. Integración continua
- 19.5.1. Ciclo de integración continua
- 19.5.2. Dependencias entre integraciones
- 19.5.3. Integración continua como Metodología de Gestión del Repositorio





19.6. Entrega continua
- 19.6.1. Entrega continua. Tipología de problemas a resolver
- 19.6.2. Entrega continua. Resolución de problemas
- 19.6.3. Ventajas de la Entrega Continua

19.7. Despliegue continuo
- 19.7.1. Despliegue Continuo. Tipología de Problemas a resolver
- 19.7.2. Despliegue Continuo. Resolución de Problemas

19.8. *Firebase Test Lab*
- 19.8.1. Configuración desde *GCloud*
- 19.8.2. Configuración de Jenkins
- 19.8.3. Uso de Jenkins. Ventajas

19.9. Configuración de *Gradle*
- 19.9.1. Sistema de Automatización *Gradle*
- 19.9.2. Componente de *Gradle Build Flavors*
- 19.9.3. Componente de *Gradle Linteo*

19.10. Ciclo de Vida en Aplicaciones Android. Ejemplo
- 19.10.1. Configuración de *SemaphoreCI* y *GitHub*
- 19.10.2. Configuración de los bloques de trabajo
- 19.10.3. Promociones y *Deployment*

Ahora empieza el camino hacia un mejor futuro y tú formas parte. Empieza a diseñar las mejores aplicaciones para móviles con este Grand Master"

06

Metodología

Este programa de capacitación ofrece una forma diferente de aprender. Nuestra metodología se desarrolla a través de un modo de aprendizaje de forma cíclica: ***el Relearning***.

Este sistema de enseñanza es utilizado, por ejemplo, en las facultades de medicina más prestigiosas del mundo y se ha considerado uno de los más eficaces por publicaciones de gran relevancia como el ***New England Journal of Medicine.***





Descubre el Relearning, un sistema que abandona el aprendizaje lineal convencional para llevarte a través de sistemas cíclicos de enseñanza: una forma de aprender que ha demostrado su enorme eficacia, especialmente en las materias que requieren memorización"

Estudio de Caso para contextualizar todo el contenido

Nuestro programa ofrece un método revolucionario de desarrollo de habilidades y conocimientos. Nuestro objetivo es afianzar competencias en un contexto cambiante, competitivo y de alta exigencia.

Con TECH podrás experimentar una forma de aprender que está moviendo los cimientos de las universidades tradicionales de todo el mundo"



Accederás a un sistema de aprendizaje basado en la reiteración, con una enseñanza natural y progresiva a lo largo de todo el temario.



El alumno aprenderá, mediante actividades colaborativas y casos reales, la resolución de situaciones complejas en entornos empresariales reales.

Un método de aprendizaje innovador y diferente

El presente programa de TECH es una enseñanza intensiva, creada desde 0, que propone los retos y decisiones más exigentes en este campo, ya sea en el ámbito nacional o internacional. Gracias a esta metodología se impulsa el crecimiento personal y profesional, dando un paso decisivo para conseguir el éxito. El método del caso, técnica que sienta las bases de este contenido, garantiza que se sigue la realidad económica, social y profesional más vigente.

> *Nuestro programa te prepara para afrontar nuevos retos en entornos inciertos y lograr el éxito en tu carrera"*

El método del caso ha sido el sistema de aprendizaje más utilizado por las mejores escuelas de Informática del mundo desde que éstas existen. Desarrollado en 1912 para que los estudiantes de Derecho no solo aprendiesen las leyes a base de contenidos teóricos, el método del caso consistió en presentarles situaciones complejas reales para que tomasen decisiones y emitiesen juicios de valor fundamentados sobre cómo resolverlas. En 1924 se estableció como método estándar de enseñanza en Harvard.

Ante una determinada situación, ¿qué debería hacer un profesional? Esta es la pregunta a la que te enfrentamos en el método del caso, un método de aprendizaje orientado a la acción. A lo largo del curso, los estudiantes se enfrentarán a múltiples casos reales. Deberán integrar todos sus conocimientos, investigar, argumentar y defender sus ideas y decisiones.

Relearning Methodology

TECH aúna de forma eficaz la metodología del Estudio de Caso con un sistema de aprendizaje 100% online basado en la reiteración, que combina elementos didácticos diferentes en cada lección.

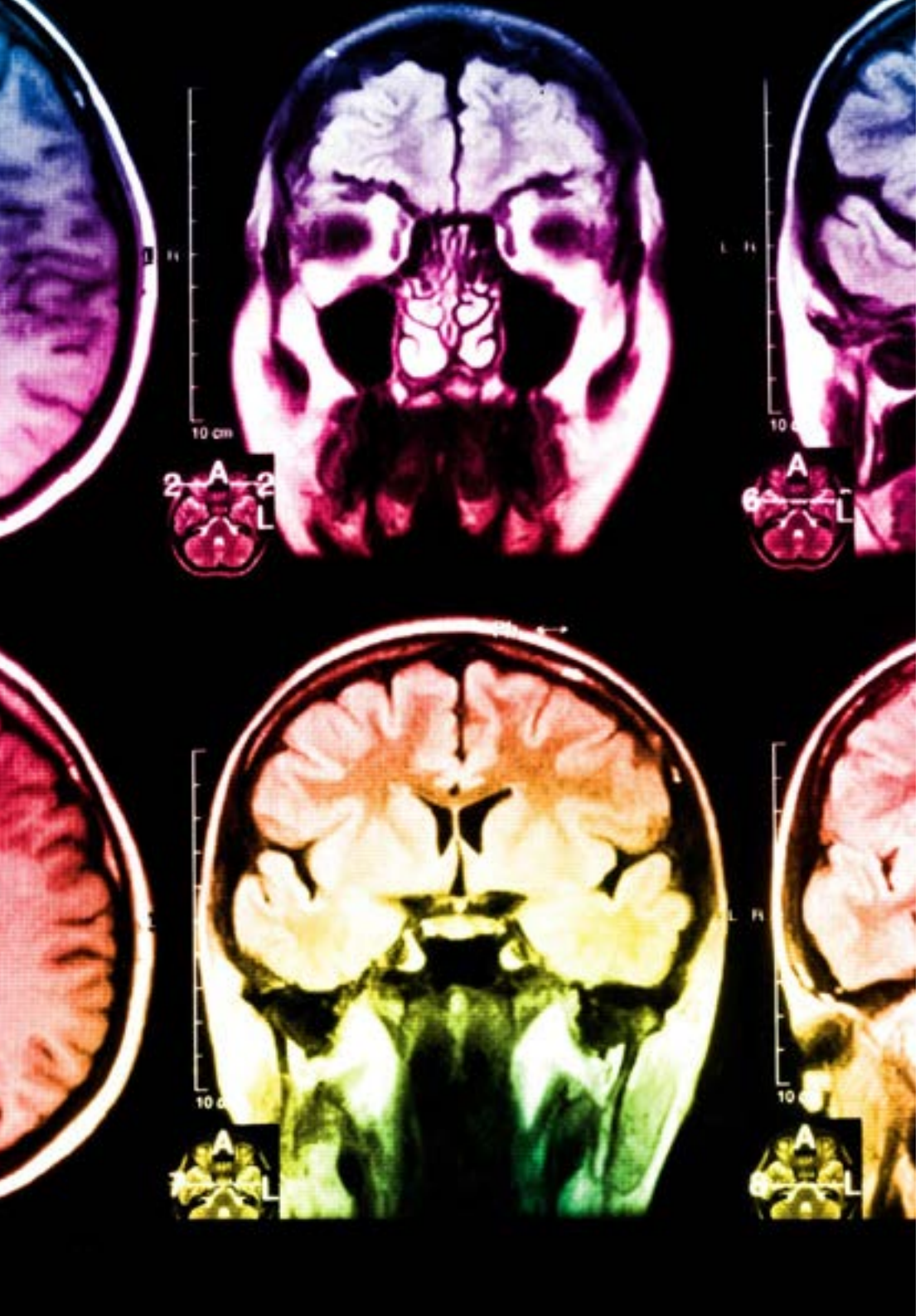
Potenciamos el Estudio de Caso con el mejor método de enseñanza 100% online: el Relearning.

En 2019 obtuvimos los mejores resultados de aprendizaje de todas las universidades online en español en el mundo.

En TECH aprenderás con una metodología vanguardista concebida para capacitar a los directivos del futuro. Este método, a la vanguardia pedagógica mundial, se denomina Relearning.

Nuestra universidad es la única en habla hispana licenciada para emplear este exitoso método. En 2019, conseguimos mejorar los niveles de satisfacción global de nuestros alumnos (calidad docente, calidad de los materiales, estructura del curso, objetivos…) con respecto a los indicadores de la mejor universidad online en español.





En nuestro programa, el aprendizaje no es un proceso lineal, sino que sucede en espiral (aprender, desaprender, olvidar y reaprender). Por eso, se combinan cada uno de estos elementos de forma concéntrica. Con esta metodología se han capacitado más de 650.000 graduados universitarios con un éxito sin precedentes en ámbitos tan distintos como la bioquímica, la genética, la cirugía, el derecho internacional, las habilidades directivas, las ciencias del deporte, la filosofía, el derecho, la ingeniería, el periodismo, la historia o los mercados e instrumentos financieros. Todo ello en un entorno de alta exigencia, con un alumnado universitario de un perfil socioeconómico alto y una media de edad de 43,5 años.

El Relearning te permitirá aprender con menos esfuerzo y más rendimiento, implicándote más en tu capacitación, desarrollando el espíritu crítico, la defensa de argumentos y el contraste de opiniones: una ecuación directa al éxito.

A partir de la última evidencia científica en el ámbito de la neurociencia, no solo sabemos organizar la información, las ideas, las imágenes y los recuerdos, sino que sabemos que el lugar y el contexto donde hemos aprendido algo es fundamental para que seamos capaces de recordarlo y almacenarlo en el hipocampo, para retenerlo en nuestra memoria a largo plazo.

De esta manera, y en lo que se denomina Neurocognitive context-dependent e-learning, los diferentes elementos de nuestro programa están conectados con el contexto donde el participante desarrolla su práctica profesional.

Este programa ofrece los mejores materiales educativos, preparados a conciencia para los profesionales:

Material de estudio

Todos los contenidos didácticos son creados por los especialistas que van a impartir el curso, específicamente para él, de manera que el desarrollo didáctico sea realmente específico y concreto.

Estos contenidos son aplicados después al formato audiovisual, para crear el método de trabajo online de TECH. Todo ello, con las técnicas más novedosas que ofrecen piezas de gran calidad en todos y cada uno los materiales que se ponen a disposición del alumno.

Clases magistrales

Existe evidencia científica sobre la utilidad de la observación de terceros expertos.

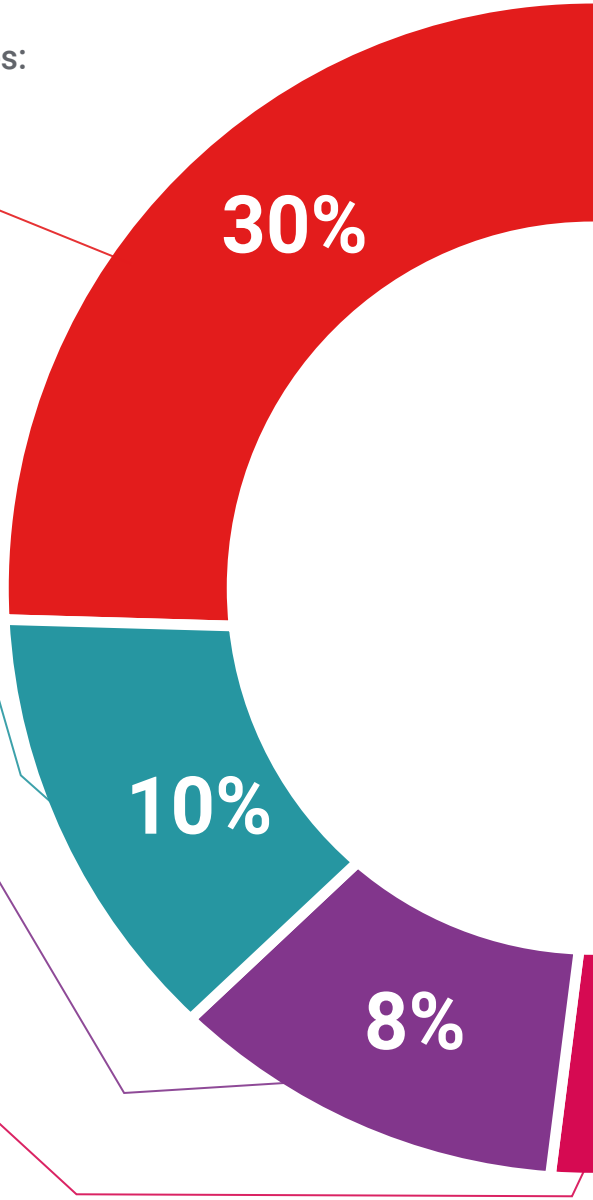
El denominado Learning from an Expert afianza el conocimiento y el recuerdo, y genera seguridad en las futuras decisiones difíciles.

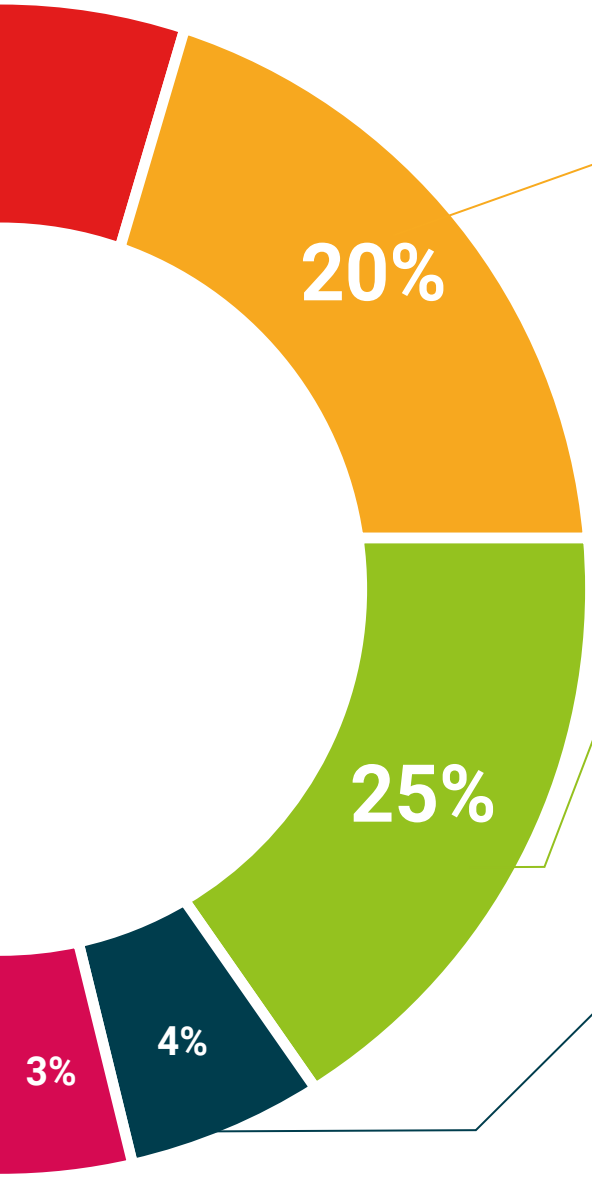
Prácticas de habilidades y competencias

Realizarán actividades de desarrollo de competencias y habilidades específicas en cada área temática. Prácticas y dinámicas para adquirir y desarrollar las destrezas y habilidades que un especialista precisa desarrollar en el marco de la globalización que vivimos.

Lecturas complementarias

Artículos recientes, documentos de consenso y guías internacionales, entre otros. En la biblioteca virtual de TECH el estudiante tendrá acceso a todo lo que necesita para completar su capacitación.





Case studies

Completarán una selección de los mejores casos de estudio elegidos expresamente para esta titulación. Casos presentados, analizados y tutorizados por los mejores especialistas del panorama internacional.

Resúmenes interactivos

El equipo de TECH presenta los contenidos de manera atractiva y dinámica en píldoras multimedia que incluyen audios, vídeos, imágenes, esquemas y mapas conceptuales con el fin de afianzar el conocimiento.

Este exclusivo sistema educativo para la presentación de contenidos multimedia fue premiado por Microsoft como "Caso de éxito en Europa".

Testing & Retesting

Se evalúan y reevalúan periódicamente los conocimientos del alumno a lo largo del programa, mediante actividades y ejercicios evaluativos y autoevaluativos para que, de esta manera, el estudiante compruebe cómo va consiguiendo sus metas.

07

Titulación

El Grand Master en Desarrollo de Aplicaciones Móviles, Experto en Android garantiza, además de la capacitación más rigurosa y actualizada, el acceso a un título de Grand Master expedido por TECH Universidad Tecnológica.





Supera con éxito este programa y recibe tu titulación universitaria sin desplazamientos ni farragosos trámites”

Este **Grand Master en Desarrollo de Aplicaciones Móviles, Experto en Android** contiene el programa más completo y actualizado del mercado.

Tras la superación de la evaluación, el alumno recibirá por correo postal* con acuse de recibo su correspondiente título de **Grand Master** emitido por **TECH Universidad Tecnológica.**

El título expedido por **TECH Universidad Tecnológica** expresará la calificación que haya obtenido en el Grand Master, y reunirá los requisitos comúnmente exigidos por las bolsas de trabajo, oposiciones y comités evaluadores de carreras profesionales.

Título: **Grand Master en Desarrollo de Aplicaciones Móviles, Experto en Android**
N.º Horas Oficiales: **3.000 h.**



tech universidad tecnológica

Otorga la presente

CONSTANCIA

a

C. _____, con documento de identificación nº_____

Por haber superado con éxito y acreditado el programa de

GRAND MASTER

en

Desarrollo de Aplicaciones Móviles, Experto en Android

Se trata de un título propio de esta Universidad con una duración de 3.000 horas, con fecha de inicio dd/mm/aaaa y fecha de finalización dd/mm/aaaa.

TECH es una Institución Particular de Educación Superior reconocida por la Secretaría de Educación Pública a partir del 28 de junio de 2018.

A 17 de junio de 2020

Mtra.Tere Guevara Navarro
Rectora

Este título propio se deberá acompañar siempre del título universitario habilitante expedido por la autoridad competente para ejercer profesionalmente en cada país. código único TECH: AFWOR23S techtitute.com/titulos



Grand Master en Desarrollo de Aplicaciones Móviles, Experto en Android

Distribución General del Plan de Estudios

Curso	Materia	Horas	Carácter
1º	Metodologías de programación en Desarrollo de Aplicaciones para dispositivos móviles	150	OB
1º	Tecnologías en Desarrollo de Aplicaciones para dispositivos móviles	150	OB
1º	Herramientas de trabajo en Desarrollo de Aplicaciones para dispositivos móviles	150	OB
1º	Desarrollo web multiplataforma orientado a dispositivos móviles	150	OB
1º	Base de datos para el Desarrollo de Aplicaciones para dispositivos móviles	150	OB
1º	Desarrollo de Aplicaciones para sistemas iOS	150	OB
1º	Despliegues en integración continua para dispositivos móviles	150	OB
1º	User experience en dispositivos móviles	150	OB
2º	Seguridad en dispositivos móviles	150	OB
2º	Lenguaje de programación Android	150	OB
2º	Frameworks en desarrollo de aplicaciones Android	150	OB
2º	Interfaces y layouts en desarrollo de aplicaciones Android	150	OB
2º	Lenguaje de Programación en Aplicaciones Android. Kotlin	150	OB
2º	Lenguaje de programación en aplicaciones Android. Kotlin avanzado. Genericidad, programación funcional y paralelismo	175	OB
2º	Gestión de datos en dispositivos Android	175	OB
2º	Herramientas de dispositivos Android	175	OB
2º	Diseño responsive en Android	175	OB
2º	Marketing en aplicaciones Android	175	OB
2º	Ciclo de vida en aplicaciones Android. Cloud, Playstore y versionado	175	OB

Mtra.Tere Guevara Navarro
Rectora

tech universidad tecnológica

*Apostilla de La Haya. En caso de que el alumno solicite que su título en papel recabe la Apostilla de La Haya, TECH EDUCATION realizará las gestiones oportunas para su obtención, con un coste adicional

tech universidad tecnológica

Grand Master

Desarrollo de Aplicaciones Móviles, Experto en Android

Modalidad: **Online**

Duración: **2 años**

Titulación: **TECH Universidad Tecnológica**

Horas lectivas: **3.000 h.**

Grand Master

Desarrollo de Aplicaciones Móviles, Experto en Android



tech universidad tecnológica