

Máster Título Propio

Desarrollo de Aplicaciones Android



Máster Título Propio Desarrollo de Aplicaciones Android

- » Modalidad: No escolarizada (100% en línea)
- » Duración: 12 meses
- » Titulación: TECH Universidad
- » Horario: a tu ritmo
- » Exámenes: online

Acceso web: www.techtitute.com/informatica/master/master-desarrollo-aplicaciones-android

Índice

01

Presentación del programa

pág. 4

02

¿Por qué estudiar en TECH?

pág. 8

03

Plan de estudios

pág. 12

04

Objetivos docentes

pág. 22

05

Salidas profesionales

pág. 28

06

Metodología de estudio

pág. 32

07

Cuadro docente

pág. 42

08

Titulación

pág. 50

01

Presentación del programa

Con más de 3.000 millones de dispositivos Android activos en el mundo, el Desarrollo de *apps* móviles se ha consolidado como uno de los sectores tecnológicos más dinámicos y con mayor proyección laboral. La demanda de especialistas capaces de abordar todo el ciclo de vida de una aplicación —desde la programación hasta su monetización— sigue en aumento. Sin embargo, no abundan los perfiles que dominen los entornos, lenguajes y arquitecturas propias de Android. En este contexto, TECH presenta una titulación innovadora y completamente online en Desarrollo de Aplicaciones Android, que permite al alumno especializarse en tan solo meses con una propuesta académica adaptada al presente y futuro del sector.



“

Domina los lenguajes de programación más relevantes del entorno Android, como Kotlin y Java, aplicándolos en el desarrollo de aplicaciones funcionales, seguras y adaptadas a las necesidades del usuario”

El crecimiento exponencial del ecosistema Android ha transformado radicalmente la manera en que las personas se comunican, consumen información y acceden a servicios digitales. En este escenario, el desarrollo de aplicaciones móviles se ha convertido en una pieza clave para la innovación tecnológica, impulsando sectores como el comercio electrónico, la educación, la salud y el entretenimiento. Dominar las herramientas y lenguajes específicos de este entorno no solo es una ventaja competitiva, sino también una necesidad en un mercado laboral que exige soluciones funcionales, escalables y adaptadas a los nuevos hábitos digitales.

Este programa ofrece una experiencia académica centrada en la adquisición de habilidades técnicas especializadas, que permiten diseñar, desarrollar, probar y optimizar aplicaciones Android de alto nivel. A través de un enfoque práctico y orientado al mercado, se abordan temas como la arquitectura de software, el diseño de interfaces intuitivas, la gestión del ciclo de vida de las *apps* y las estrategias de publicación en tiendas digitales. Quienes dominen estos conocimientos no solo podrán acceder a oportunidades en empresas tecnológicas globales, sino también liderar proyectos propios, emprender y aportar soluciones digitales de alto impacto.

Una de las mayores fortalezas de esta titulación es su modalidad completamente online, que permite acceder a contenidos actualizados, recursos interactivos y clases magistrales sin las limitaciones geográficas o de horarios de los programas tradicionales. Esta flexibilidad resulta especialmente útil para quienes ya se desempeñan en el ámbito tecnológico y desean ampliar sus competencias sin detener su actividad profesional.

Con un enfoque alineado a las exigencias reales de la industria, este programa representa una oportunidad estratégica para integrarse a un mercado en expansión, con una demanda creciente de perfiles especializados en Desarrollo de Aplicaciones Android. Además, los alumnos accederán a 10 *Marteclasses* de gran valor académico llevadas a cabo por un Director Invitado Internacional.

Este **Máster Título Propio en Desarrollo de Aplicaciones Android** contiene el programa universitario más completo y actualizado del mercado. Sus características más destacadas son:

- ♦ El desarrollo de casos prácticos presentados por expertos en Desarrollo de Aplicaciones
- ♦ 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 en Desarrollo de Aplicaciones
- ♦ 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



Un reconocido Director Invitado Internacional ofrecerá 10 exclusivas Masterclasses para profundizar en las últimas tendencias en el Desarrollo de Aplicaciones Android"

“

Gracias al método Relearning de TECH, lograrás afianzar los conceptos claves que te ofrece esta enseñanza universitaria de manera progresiva y natural”

Incluye en su cuadro docente a profesionales pertenecientes al ámbito del Desarrollo de Aplicaciones, 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.

Dispondrás de una sólida comprensión sobre el ciclo de vida de la Aplicación, desde la concepción hasta su publicación y mantenimiento.

Abordarás la realización de pruebas unitarias e instrumentadas, así como control de calidad del código y depuración de errores.


Android Studio
Version 3.3

new Android Studio project
the project

02

¿Por qué estudiar en TECH?

TECH es la mayor Universidad digital del mundo. Con un impresionante catálogo de más de 14.000 programas universitarios, disponibles en 11 idiomas, se posiciona como líder en empleabilidad, con una tasa de inserción laboral del 99%. Además, cuenta con un enorme claustro de más de 6.000 profesores de máximo prestigio internacional.



“

Estudia en la mayor universidad digital del mundo y asegura tu éxito profesional. El futuro empieza en TECH”

La mejor universidad online del mundo según FORBES

La prestigiosa revista Forbes, especializada en negocios y finanzas, ha destacado a TECH como «la mejor universidad online del mundo». Así lo han hecho constar recientemente en un artículo de su edición digital en el que se hacen eco del caso de éxito de esta institución, «gracias a la oferta académica que ofrece, la selección de su personal docente, y un método de aprendizaje innovador orientado a formar a los profesionales del futuro».

Forbes
Mejor universidad
online del mundo

Plan
de estudios
más completo

Los planes de estudio más completos del panorama universitario

TECH ofrece los planes de estudio más completos del panorama universitario, con temarios que abarcan conceptos fundamentales y, al mismo tiempo, los principales avances científicos en sus áreas científicas específicas. Asimismo, estos programas son actualizados continuamente para garantizar al alumnado la vanguardia académica y las competencias profesionales más demandadas. De esta forma, los títulos de la universidad proporcionan a sus egresados una significativa ventaja para impulsar sus carreras hacia el éxito.

El mejor claustro docente top internacional

El claustro docente de TECH está integrado por más de 6.000 profesores de máximo prestigio internacional. Catedráticos, investigadores y altos ejecutivos de multinacionales, entre los cuales se destacan Isaiah Covington, entrenador de rendimiento de los Boston Celtics; Magda Romanska, investigadora principal de MetaLAB de Harvard; Ignacio Wistumba, presidente del departamento de patología molecular traslacional del MD Anderson Cancer Center; o D.W Pine, director creativo de la revista TIME, entre otros.

Profesorado
TOP
Internacional

La metodología
más eficaz

Un método de aprendizaje único

TECH es la primera universidad que emplea el *Relearning* en todas sus titulaciones. Se trata de la mejor metodología de aprendizaje online, acreditada con certificaciones internacionales de calidad docente, dispuestas por agencias educativas de prestigio. Además, este disruptivo modelo académico se complementa con el "Método del Caso", configurando así una estrategia de docencia online única. También en ella se implementan recursos didácticos innovadores entre los que destacan vídeos en detalle, infografías y resúmenes interactivos.

La mayor universidad digital del mundo

TECH es la mayor universidad digital del mundo. Somos la mayor institución educativa, con el mejor y más amplio catálogo educativo digital, cien por cien online y abarcando la gran mayoría de áreas de conocimiento. Ofrecemos el mayor número de titulaciones propias, titulaciones oficiales de posgrado y de grado universitario del mundo. En total, más de 14.000 títulos universitarios, en diez idiomas distintos, que nos convierten en la mayor institución educativa del mundo.

nº1
Mundial
Mayor universidad
online del mundo

La universidad online oficial de la NBA

TECH es la universidad online oficial de la NBA. Gracias a un acuerdo con la mayor liga de baloncesto, ofrece a sus alumnos programas universitarios exclusivos, así como una gran variedad de recursos educativos centrados en el negocio de la liga y otras áreas de la industria del deporte. Cada programa tiene un currículo de diseño único y cuenta con oradores invitados de excepción: profesionales con una distinguida trayectoria deportiva que ofrecerán su experiencia en los temas más relevantes.

Líderes en empleabilidad

TECH ha conseguido convertirse en la universidad líder en empleabilidad. El 99% de sus alumnos obtienen trabajo en el campo académico que ha estudiado, antes de completar un año luego de finalizar cualquiera de los programas de la universidad. Una cifra similar consigue mejorar su carrera profesional de forma inmediata. Todo ello gracias a una metodología de estudio que basa su eficacia en la adquisición de competencias prácticas, totalmente necesarias para el desarrollo profesional.



Google Partner Premier

El gigante tecnológico norteamericano ha otorgado TECH la insignia Google Partner Premier. Este galardón, solo al alcance del 3% de las empresas del mundo, pone en valor la experiencia eficaz, flexible y adaptada que esta universidad proporciona al alumno. El reconocimiento no solo acredita el máximo rigor, rendimiento e inversión en las infraestructuras digitales de TECH, sino que también sitúa a esta universidad como una de las compañías tecnológicas más punteras del mundo.



La universidad mejor valorada por sus alumnos

Los alumnos han posicionado a TECH como la universidad mejor valorada del mundo en los principales portales de opinión, destacando su calificación más alta de 4,9 sobre 5, obtenida a partir de más de 1.000 reseñas. Estos resultados consolidan a TECH como la institución universitaria de referencia a nivel internacional, reflejando la excelencia y el impacto positivo de su modelo educativo.



03

Plan de estudios

El plan de estudios abordará el Desarrollo de Aplicaciones Android, explorando desde los fundamentos del sistema operativo y su arquitectura hasta el uso de *frameworks*. Asimismo, el temario profundizará en el diseño de interfaces, la gestión de datos, la integración con servicios *cloud* y el despliegue continuo en entornos reales. Así, los informáticos adquirirán competencias técnicas y metodológicas para liderar proyectos móviles y crear experiencias digitales optimizadas que respondan a las exigencias del mercado actual.



“

Diseñarás interfaces móviles centradas en la experiencia de usuario, aplicando principios de diseño visual y usabilidad que mejoran la interacción”

Módulo 1. Lenguaje de programación Android

- 1.1. Plataforma Android
 - 1.1.1. Plataforma Android
 - 1.1.2. Sistema operativo Android
 - 1.1.3. *Open handset alliance* en el Desarrollo de Android
- 1.2. Arquitectura Android
 - 1.2.1. Elementos arquitecturales de un sistema Android
 - 1.2.2. Comunicación entre elementos
 - 1.2.3. Extensibilidad de la arquitectura Android
 - 1.2.4. Gestión de recursos máquina: Batería y Memoria
 - 1.2.5. Emuladores Android
- 1.3. Núcleo Linux de Android
 - 1.3.1. Composición del núcleo
 - 1.3.2. Elementos estructurales del núcleo
 - 1.3.3. La máquina virtual Dalvik
 - 1.3.4. La máquina virtual Android RunTime (ART)
- 1.4. Librerías nativas de Android
 - 1.4.1. Librerías nativas de Android
 - 1.4.2. Librerías de compatibilidad (*support library*)
 - 1.4.3. Librerías nativas y extensibilidad
- 1.5. El sistema de ficheros y datos en Android
 - 1.5.1. Estructura de una aplicación Android típica
 - 1.5.2. Sistema de ficheros YAFFS2 y ext4
 - 1.5.3. Uso de SQLite y Room para la gestión de datos
- 1.6. Seguridad en Android
 - 1.6.1. Sistema de permisos
 - 1.6.2. Firmas digitales en los Android *application package* (apk)
 - 1.6.3. Ejecución de procesos en el núcleo
 - 1.6.4. Hilos de ejecución y eventos

- 1.7. Componentes estructurales de una aplicación estándar
 - 1.7.1. Vista (*view*)
 - 1.7.2. Actividad (*activity*)
 - 1.7.3. Fragmento (*fragment*)
 - 1.7.4. Servicio (*service*)
 - 1.7.5. Intención (*intent*)
 - 1.7.6. *Broadcasts receiver* y *content provider*
 - 1.7.7. Gestión de datos y preferencias de usuario
- 1.8. Versiones Android
 - 1.8.1. Versiones Android
 - 1.8.2. Despliegue de versiones Android
 - 1.8.3. Dispersión de distribuciones Android
 - 1.8.4. Android *versus* Apple IOS y otros sistemas para móviles
- 1.9. Android para vehículos
 - 1.9.1. Android y el mundo del automóvil
 - 1.9.2. Elementos estructurales en un sistema Android para automóviles
 - 1.9.3. Comunicación entre dispositivos
- 1.10. Android en domótica, *wearable* y en *internet of things* (IoT)
 - 1.10.1. El mundo conectado
 - 1.10.2. Elementos estructurales en un sistema Android Domótico
 - 1.10.3. Elementos de Android *wearable*
 - 1.10.4. Android en el *Internet of things* (IoT)

Módulo 2. Frameworks en Desarrollo de Aplicaciones Android

- 2.1. Frameworks en Desarrollo de Aplicaciones Android
 - 2.1.1. Framework en Desarrollo de Aplicaciones Android
 - 2.1.2. Frameworks. Tipologías
 - 2.1.3. Elección del *framework* para el proyecto
- 2.2. Implementación de *frameworks* en Android
 - 2.2.1. *Frameworks core* de Android para Java/Kotlin
 - 2.2.2. *Jetpack compose*
 - 2.2.3. Frameworks en otros lenguajes



- 2.3. Sistemas de gestión de librerías en Desarrollo
 - 2.3.1. *Graddle*
 - 2.3.2. Automatización con *Graddle*
 - 2.3.3. Herramienta del Desarrollo Maven
- 2.4. Código limpio
 - 2.4.1. Código ordenado
 - 2.4.2. Preparación de código en Aplicaciones Android
 - 2.4.3. *Bikeshedding* y priorización
- 2.5. Patrones de Desarrollo en Android
 - 2.5.1. Categorías de patrones
 - 2.5.2. Diferencias entre patrones
 - 2.5.3. *Factory, Observer y Singleton*
- 2.6. MVP. Modelo, vista y presentador
 - 2.6.1. MVC. Modelo, vista y controlador
 - 2.6.2. Modelo, vista y presentador
 - 2.6.3. Ejemplo práctico: *Pokemon battle*
- 2.7. MVVM. Modelo, vista y *view model*
 - 2.7.1. MVC vs MVVM
 - 2.7.2. Modelo, vista y *view model*
 - 2.7.3. Ejemplo práctico: *Pokemon battle II*
- 2.8. *Frameworks* y librerías más utilizadas en Android
 - 2.8.1. Librerías de interacción con API
 - 2.8.2. Librerías de conversión de datos
 - 2.8.3. *Firebase y firebase analytics*
- 2.9. *Framework* visual de Android
 - 2.9.1. Ciclo de vida de una aplicación Android
 - 2.9.2. Diseño de vistas en XML
 - 2.9.3. Diseño de elementos y animaciones en XML
- 2.10. *Frameworks* de Android en otros lenguajes
 - 2.10.1. *React native*
 - 2.10.2. *Flutter*
 - 2.10.3. *Ionic*

Módulo 3. Interfaces y layouts en Desarrollo de Aplicaciones Android

- 3.1. Ciclo de vida de un interfaz en Android
 - 3.1.1. Ciclo de vida en Android
 - 3.1.2. Relación proceso-actividades
 - 3.1.3. Persistencia de estado de la aplicación
 - 3.1.4. Arquitectura *clean* aplicada a Android
- 3.2. Vistas en el Desarrollo de Aplicaciones Android (*views*)
 - 3.2.1. Capa de presentación de la arquitectura *clean*
 - 3.2.2. *RecyclerView*
 - 3.2.3. *Adapter view*
- 3.3. Diseños en el Desarrollo de Aplicaciones Android (*layouts*)
 - 3.3.1. *Layouts* en Android
 - 3.3.2. *Constraint layout*
 - 3.3.3. Creación de *layouts* usando *Android Studio Layout Editor*
- 3.4. Animaciones en el Desarrollo de Aplicaciones Android (*animations*)
 - 3.4.1. Iconos e imágenes
 - 3.4.2. Transiciones
 - 3.4.3. Diferencia entre animación de propiedades y animación de vistas
- 3.5. Actividades e intenciones en el Desarrollo de Aplicaciones Android (*activity e intentions*)
 - 3.5.1. Intenciones explícitas e implícitas
 - 3.5.2. Barra de acciones
 - 3.5.3. Comunicación entre actividades
- 3.6. Recursos alternativos y de sistema (*material design, cardboard...*)
 - 3.6.1. *Material design* para Android
 - 3.6.2. Multimedia en Android
 - 3.6.3. Realidad virtual con *Google Cardboard for Android NDK*
- 3.7. Estilos y temas en el Desarrollo de Aplicaciones Android
 - 3.7.1. Estilos en un proyecto Android
 - 3.7.2. Temas para el proyecto Android
 - 3.7.3. Reutilización de estilos y temas

- 3.8. Gráficos, pantalla táctil y sensores
 - 3.8.1. Trabajo con gráficos avanzados
 - 3.8.2. Gestión de dispositivos con pantalla táctil y con teclado
 - 3.8.3. Uso de sensores del dispositivo con Android
- 3.9. Diseños para realidad aumentada
 - 3.9.1. Interfaces complejas usando la cámara
 - 3.9.2. Sensores de posición y GPS en la realidad aumentada
 - 3.9.3. Presentación en pantallas no estándar
 - 3.9.4. Errores y problemas comunes
- 3.10. Configuración avanzada de interfaz con *AndroidManifest.xml*
 - 3.10.1. La potencia del fichero de manifiesto de Android
 - 3.10.2. Diseño programático *versus* diseño declarativo
 - 3.10.3. Componentes clave del archivo

Módulo 4. Lenguaje de programación en Aplicaciones Android. Kotlin

- 4.1. Lenguaje de programación Kotlin
 - 4.1.1. Lenguaje de programación Kotlin
 - 4.1.2. Lenguaje de programación Kotlin. Características
 - 4.1.3. Funcionamiento de un programa en Kotlin
- 4.2. Programar en Kotlin
 - 4.2.1. Estructura de un programa en Kotlin
 - 4.2.2. Palabras reservadas y sintaxis
 - 4.2.3. Escribir por consola y leer *inputs* del usuario – hola mundo
- 4.3. Tipos y variables en Kotlin
 - 4.3.1. Tipos y variables en Kotlin
 - 4.3.2. Declaración de variables: *Var Vs Val*
 - 4.3.3. Operadores
 - 4.3.4. Conversión de tipos
 - 4.3.5. *Arrays*
- 4.4. Control de flujo en Kotlin
 - 4.4.1. Control de flujo
 - 4.4.2. Expresiones condicionales
 - 4.4.3. Bucles

- 4.5. Funciones en Kotlin
 - 4.5.1. Funciones en Kotlin
 - 4.5.2. Estructura de una función
 - 4.5.3. Funciones de alcance (*scope functions*)
- 4.6. Clases y objetos en Kotlin
 - 4.6.1. Clases y objetos en Kotlin
 - 4.6.2. Clases
 - 4.6.3. Objetos
 - 4.6.4. Constructores e Inicialización de propiedades
 - 4.6.5. Clases anidadas y clases internas
 - 4.6.6. Clases de datos
- 4.7. Herencia en Kotlin
 - 4.7.1. Herencia
 - 4.7.2. Superclases y subclases
 - 4.7.3. Sobreescritura de propiedades y funciones
 - 4.7.4. Herencia vs otros tipos de relación entre clases
 - 4.7.5. Clases selladas
 - 4.7.6. Enumerados
- 4.8. Clases abstractas e Interfaces en Kotlin
 - 4.8.1. Clases abstractas e Interfaces
 - 4.8.2. Clases abstractas
 - 4.8.3. Interfaces
 - 4.8.4. Validación y conversión de tipos – operadores *is*, *when*, *as*
- 4.9. Colecciones en Kotlin
 - 4.9.1. Colecciones en Kotlin
 - 4.9.2. Lista
 - 4.9.3. Conjunto
 - 4.9.4. Mapa
- 4.10. Gestión de excepciones y valores nulos en Kotlin
 - 4.10.1. Gestión de excepciones y valores nulos
 - 4.10.2. Valor nulo, tipos *nullable* y *non-nullable*
 - 4.10.3. Excepciones

Módulo 5. Lenguaje de programación en Aplicaciones Android. Kotlin avanzado

- 5.1. Genericidad en Kotlin
 - 5.1.1. Genericidad en Kotlin
 - 5.1.2. Genericidad en colecciones, funciones, clases e interfaces
 - 5.1.3. Covarianza y contravarianza: *out* o *In*
- 5.2. Funciones lambda en Kotlin
 - 5.2.1. Funciones lambdas
 - 5.2.2. Estructura de una función lambda
 - 5.2.3. Uso de funciones lambda
- 5.3. Funciones de orden superior en Kotlin
 - 5.3.1. Funciones de orden superior
 - 5.3.2. Funciones de orden superior estándares de Kotlin
 - 5.3.3. Enlazando llamadas a funciones
- 5.4. Extensiones en Kotlin
 - 5.4.1. Extensiones en Kotlin
 - 5.4.2. Funciones de extensión
 - 5.4.3. Propiedades de extensión
 - 5.4.4. Objetos acompañantes
- 5.5. Patrón *delegation* en Kotlin
 - 5.5.1. Patrón *delegation*
 - 5.5.2. *Delegation* en Kotlin
 - 5.5.3. Propiedades delegadas
- 5.6. Anotaciones y reflexión en Kotlin
 - 5.6.1. Anotaciones y reflexión
 - 5.6.2. Anotaciones en Kotlin
 - 5.6.3. Reflexión en Kotlin
- 5.7. *Testing* en Kotlin
 - 5.7.1. *Testing* en Kotlin
 - 5.7.2. *Frameworks* y librerías de *testing* en Kotlin
 - 5.7.3. Kotest

- 5.8. Programación asíncrona en Kotlin
 - 5.8.1. Programación asíncrona
 - 5.8.2. Técnicas de programación asíncrona en Kotlin
 - 5.8.3. Comparativa en técnicas de programación
- 5.9. Corrutinas en Kotlin
 - 5.9.1. Corrutinas
 - 5.9.2. Canales
 - 5.9.3. Contexto y *dispatchers*
 - 5.9.4. Estado compartido y concurrencia
 - 5.9.5. Gestión de excepciones en corrutinas
- 5.10. Ecosistema Kotlin
 - 5.10.1. Ecosistema Kotlin
 - 5.10.2. Librerías para Kotlin
 - 5.10.3. Herramientas para Kotlin

Módulo 6. Gestión de datos en dispositivos Android

- 6.1. Gestión de datos. Tipología
 - 6.1.1. La Gestión de datos en Dispositivos móviles
 - 6.1.2. Alternativas para la gestión de datos en dispositivos Android
 - 6.1.3. Generación de datos para trabajo por inteligencia artificial y analítica de uso
 - 6.1.4. Herramientas de medición de rendimiento para una óptima gestión de datos
- 6.2. Gestión de preferencias de usuario
 - 6.2.1. Tipos de datos involucrados en los archivos de preferencias
 - 6.2.2. Gestión de las preferencias de usuario
 - 6.2.3. Exportación de las preferencias. Gestión de permisos
- 6.3. Sistema de almacenamiento de ficheros
 - 6.3.1. Clasificación de sistemas de ficheros en dispositivos móviles
 - 6.3.2. Sistema interno de ficheros
 - 6.3.3. Sistema externo de ficheros
- 6.4. Archivos JSON como almacenamiento en Android
 - 6.4.1. Información no estructurada en ficheros JSON
 - 6.4.2. Librerías para gestión de datos con JSON
 - 6.4.3. Uso de JSON en Android. Recomendaciones y optimizaciones

- 6.5. Archivos XML como almacenamiento en Android
 - 6.5.1. Formato XML en Android
 - 6.5.2. XML a través de librerías SAX
 - 6.5.3. XML a través de librerías DOM
- 6.6. Bases de datos SQLite
 - 6.6.1. Base de datos relacional para la gestión de datos
 - 6.6.2. Uso de la base de datos
 - 6.6.3. Métodos SQLite para la gestión de datos
- 6.7. Uso avanzado de bases de datos SQLite
 - 6.7.1. Recuperación de fallos usando transacciones en SQLite
 - 6.7.2. Uso de caché para acelerar el acceso a datos
 - 6.7.3. Base de datos en móviles
- 6.8. Librería *Room*
 - 6.8.1. Arquitectura de la librería *Room*
 - 6.8.2. Librería *Room*. Funcionalidad
 - 6.8.3. Librería *Room*. Ventajas y desventajas
- 6.9. *Content provider* para compartir información
 - 6.9.1. *Content provider* para compartir información
 - 6.9.2. *Content provider* en Android. Técnico de uso
 - 6.9.3. Seguridad en *content provider*
- 6.10. Recogida de datos en *cloud* de internet
 - 6.10.1. Android y sistemas de almacenamiento en la nube (*cloud*)
 - 6.10.2. Servicios SOAP y REST para Android
 - 6.10.3. Problemática de los sistemas distribuidos
 - 6.10.4. Internet como copia de seguridad de los datos de la aplicación

Módulo 7. Herramientas de dispositivos Android

- 7.1. Gestión: Herramientas "TO DO"
 - 7.1.1. Herramientas de mercado
 - 7.1.2. Herramientas de mercado. Comparativa de funcionalidades
 - 7.1.3. Herramientas de gestión. Diferencias

- 7.2. MDM: Gestión empresarial de dispositivos
 - 7.2.1. Control de los dispositivos de empresa
 - 7.2.2. Análisis de las principales herramientas de mercado
 - 7.2.3. Elección de la herramienta
- 7.3. CRM: Herramientas de mercado
 - 7.3.1. Análisis de las herramientas de mercado con aplicación en Android
 - 7.3.2. Herramientas de mercado. Eficiencia
 - 7.3.3. Herramientas de mercado. Usos
- 7.4. Drones con Android
 - 7.4.1. Apps de dispositivos Android para control de dron
 - 7.4.2. Controles autónomos
 - 7.4.3. Usos de dron en Android
- 7.5. Android, valor añadido en plataformas bancarias
 - 7.5.1. Android en plataformas bancarias
 - 7.5.2. Riesgos y fraudes de ciberdelincuentes
 - 7.5.3. Usos de dispositivos móviles
- 7.6. *Brokering* en dispositivos móviles
 - 7.6.1. Herramientas de mercado y su uso
 - 7.6.2. Comparativa de las herramientas
 - 7.6.3. Elección de herramienta para cada uso
- 7.7. Herramientas de entretenimiento y formación
 - 7.7.1. Usos
 - 7.7.2. Herramientas del mercado
 - 7.7.3. Comparativas de funcionalidades entre Herramientas de Desarrollo Android
- 7.8. IoT Android
 - 7.8.1. *Framework* y plataformas de mercado
 - 7.8.2. Riesgos y consideraciones de IoT en Android
 - 7.8.3. Usos de IoT en Android
- 7.9. Eficiencia de procesos
 - 7.9.1. Análisis de herramientas de mercado para creación de Apps
 - 7.9.2. Comparativa de herramientas de creación de Aplicaciones para Android
 - 7.9.3. *Use case*

- 7.10. Aplicaciones más descargadas en la actualidad
 - 7.10.1. Herramientas más descargadas en la actualidad
 - 7.10.2. Agrupación por familias
 - 7.10.3. Usos principales, secundarios y comparativas con IOS

Módulo 8. Diseño *responsive* en Android

- 8.1. *Responsive design*
 - 8.1.1. Diseño *responsive*
 - 8.1.2. Usabilidad, accesibilidad y UX
 - 8.1.3. Diseño *responsive*. Ventajas y desventajas
- 8.2. *Mobile vs tablet vs web vs smartwatches*
 - 8.2.1. Diferentes formatos, diferentes tamaños, diferentes necesidades
 - 8.2.2. Problemas en el diseño
 - 8.2.3. Adaptativo vs responsivo
- 8.3. Guías de estilos
 - 8.3.1. Guías de estilos. Utilidad
 - 8.3.2. *Material design*
 - 8.3.3. Guía de estilos propia
- 8.4. *Layouting flexible*
 - 8.4.1. *Layouting flexible*
 - 8.4.2. *Layouting* básico
 - 8.4.3. *Layouting* en *Grid*
 - 8.4.4. *Layouting* con *RelativeLayout*
 - 8.4.5. *Layouting* con *ConstraintLayout*
- 8.5. Recursos flexibles
 - 8.5.1. Recursos flexibles
 - 8.5.2. Imágenes
 - 8.5.3. 9patch
 - 8.5.4. Recursos globales
- 8.6. Navegación flexible
 - 8.6.1. Navegación flexible
 - 8.6.2. Navegación con *activities*
 - 8.6.3. Navegación con *fragments*

- 8.7. Herramientas externas
 - 8.7.1. Generadores automáticos
 - 8.7.2. Herramientas de prototipado
 - 8.7.3. Herramientas de diseño
- 8.8. *Debug y tests*
 - 8.8.1. *Debug layouts*
 - 8.8.2. *Tests automáticos*
 - 8.8.3. Desarrollo basado en componentes
 - 8.8.4. Buenas prácticas de *testing* y pruebas
- 8.9. Alternativas a Android nativo I. *Web pages*
 - 8.9.1. Diseño en un *webView*
 - 8.9.2. *ChromeCustomTabs*
 - 8.9.3. *Debug y tests* en *web pages*
- 8.10. Alternativas a Android nativo II. Aplicaciones híbridas
 - 8.10.1. *React/react native*
 - 8.10.2. *Flutter*
 - 8.10.3. *Ionic*
 - 8.10.4. *Apache Cordova*

Módulo 9. Marketing en Aplicaciones Android

- 9.1. De *customer service* a *customer experience*
 - 9.1.1. *Customer service*. Desarrollo del cliente actual
 - 9.1.2. Usuario con acceso a la información. Exigencias y necesidades
 - 9.1.3. La retroalimentación como fuente de conocimiento
- 9.2. *Customer journey*
 - 9.2.1. Camino del usuario a la conversión
 - 9.2.2. Microsegmentación
 - 9.2.3. Experiencia a través de los canales
- 9.3. Medición de la experiencia de usuario
 - 9.3.1. Arquitectura web y móvil
 - 9.3.2. Analítica de sesión como nuevo estándar
 - 9.3.3. Estado del arte de la experiencia de usuario

- 9.4. Marketing de Aplicaciones Android
 - 9.4.1. CX+IA
 - 9.4.2. CX+*Blockchain*
 - 9.4.3. CX+IoT
- 9.5. Productos de CX (experiencia de cliente)
 - 9.5.1. Estándares de la industria
 - 9.5.2. Telepresencia
 - 9.5.3. Experiencia de cliente para todos los agentes del desarrollo
- 9.6. Trabajo centrado en el usuario
 - 9.6.1. Equipos
 - 9.6.2. Pensamiento de diseñador
 - 9.6.3. Trabajo de campo
- 9.7. La ciencia sobre el usuario
 - 9.7.1. La ciencia sobre el usuario. Reglas de oro
 - 9.7.2. Iteración
 - 9.7.3. Errores comunes
- 9.8. Prototipado y *wireframing*
 - 9.8.1. Prototipado y *wireframing*
 - 9.8.2. *Hands-on*
 - 9.8.3. Nivel avanzado
- 9.9. Interfaces móviles
 - 9.9.1. Diseño visual. Reglas
 - 9.9.2. Interfaz de apps. Claves
 - 9.9.3. Buenas prácticas en el desarrollo de interfaces móviles
- 9.10. Buenas prácticas en la experiencia de usuario. Consejos para desarrolladores
 - 9.10.1. Nivel uno. Buenas prácticas en CX
 - 9.10.2. Nivel dos. Buenas prácticas en UX
 - 9.10.3. Nivel tres. Buenas prácticas en UI

Módulo 10. Ciclo de vida en Aplicaciones Android. *Cloud, play store* y versionado

- 10.1. Ciclo de vida del *software*
 - 10.1.1. Ciclo de vida del *software*
 - 10.1.2. Metodologías ágiles
 - 10.1.3. El ciclo continuo ágil del *software*
- 10.2. Desarrollo del producto manual
 - 10.2.1. Integración manual
 - 10.2.2. Entrega manual
 - 10.2.3. Despliegue manual
- 10.3. Integración supervisada
 - 10.3.1. Integración continua
 - 10.3.2. Revisión manual
 - 10.3.3. Revisiones automáticas estáticas
- 10.4. Pruebas lógicas
 - 10.4.1. Pruebas unitarias
 - 10.4.2. Pruebas de integración
 - 10.4.3. Pruebas de comportamiento
- 10.5. Integración continua
 - 10.5.1. Ciclo de integración continua
 - 10.5.2. Dependencias entre integraciones
 - 10.5.3. Integración continua como metodología de gestión del repositorio
- 10.6. Entrega continua
 - 10.6.1. Entrega continua. Tipología de problemas a resolver
 - 10.6.2. Entrega continua. Resolución de problemas
 - 10.6.3. Ventajas de la entrega continua
- 10.7. Despliegue continuo
 - 10.7.1. Despliegue continuo. Tipología de problemas a resolver
 - 10.7.2. Despliegue continuo. Resolución de problemas
- 10.8. *Firebase test lab*
 - 10.8.1. Configuración desde GCloud
 - 10.8.2. Configuración de Jenkins
 - 10.8.3. Uso de Jenkins. Ventajas

- 10.9. Configuración de *gradle*
 - 10.9.1. Sistema de automatización *gradle*
 - 10.9.2. Componente de *gradle build flavors*
 - 10.9.3. Componente de *gradle linteo*
- 10.10. Ciclo de vida en Aplicaciones Android. Ejemplo
 - 10.10.1. Configuración de *semaphoreCI* y GitHub
 - 10.10.2. Configuración de los bloques de trabajo
 - 10.10.3. Promociones y *deployment*



Entenderás el funcionamiento de arquitecturas modernas como MVVM y su importancia para crear aplicaciones sostenibles”

04

Objetivos docentes

Los objetivos docentes de este programa han sido diseñados para garantizar una especialización integral en el Desarrollo de Aplicaciones Android. Por un lado, se busca dominar los lenguajes, entornos y herramientas clave del ecosistema Android; por otro, se promueve la capacidad de diseñar interfaces intuitivas, aplicar buenas prácticas de arquitectura y gestionar el ciclo completo de una *app*. Además, se impulsa una visión analítica orientada a la optimización del rendimiento y la experiencia de usuario. Así, el itinerario académico permite adquirir competencias alineadas con las demandas reales del mercado y con los avances constantes del sector tecnológico.



“

Integra servicios web, APIs y bases de datos en la nube, garantizando la conectividad de las aplicaciones y ampliando sus funcionalidades a través de recursos externos”



Objetivos generales

- ♦ Determinar los elementos estructurales de un Sistema Android
- ♦ Analizar las diferencias entre los distintos *Frameworks* de Desarrollo, puntos fuertes y debilidades
- ♦ Establecer cómo se crean estilos y temas incluyendo los de Realidad Virtual a través de los *Layouts*
- ♦ Analizar las distintas formas de gestión de flujo del código en Kotlin
- ♦ Desarrollar capacidades avanzadas y mejores prácticas en el Desarrollo de Aplicaciones en el Lenguaje de Programación Kotlin
- ♦ 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. 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

Módulo 2. *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 librerías 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 3. 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

Módulo 4. 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

Módulo 5. 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

Módulo 6. 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

Módulo 7. 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

Módulo 8. 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 9. 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

Módulo 10. Ciclo de vida en Aplicaciones Android. *Cloud, play store* 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



Optimizarás el rendimiento de tus desarrollos mediante técnicas de depuración, control de memoria y gestión eficiente de recursos, asegurando la fluidez de las aplicaciones”

05 Salidas profesionales

El Desarrollo de Aplicaciones Android se ha convertido en una de las competencias más demandadas en la industria tecnológica global. Gracias a la expansión del *mobile first* y la necesidad de soluciones digitales personalizadas, las salidas profesionales en este ámbito son amplias y versátiles. Desde puestos en grandes corporaciones tecnológicas hasta roles clave en *startups*, agencias digitales o proyectos freelance, las oportunidades son crecientes. Además, el dominio de este entorno facilita el acceso a mercados internacionales y permite emprender con autonomía. Por ello, especializarse en esta área representa una decisión estratégica con proyección laboral sólida y sostenida.



```
/*  
 * Parses Point from its String representation.  
 * @param locationString - String that represents location, as 2 double values split with comma. Always space after comma.  
 * @return org.springframework.data.geo.Point instance  
 */  
public static Point parseLocation(String locationString) {  
    Preconditions.checkNotNull(locationString);  
    Preconditions.checkArgument(locationString.contains(","), "Location should contain a comma");  
    locationString = locationString.trim();  
  
    if (locationString.contains(" "))  
        locationString = locationString.replace(" ", "");  
  
    if (locationString.contains(",,") || locationString.contains(",."))  
        locationString = locationString.replaceAll("[,\\.]", "");  
  
    String[] location = locationString.split(",");  
    Preconditions.checkArgument(location.length == 2, "Location should consist of two values");  
    double lat = Double.parseDouble(location[0]);  
    double lon = Double.parseDouble(location[1]);  
  
    return new Point(lat, lon);  
}
```

“

Aplicarás metodologías ágiles como Scrum o Kanban para planificar, ejecutar y mejorar proyectos de desarrollo”

Perfil del egresado

El egresado de este programa destaca por su capacidad para desarrollar soluciones móviles innovadoras dentro del ecosistema Android, combinando conocimientos técnicos avanzados con una comprensión profunda de la experiencia de usuario y la lógica de negocio digital. Gracias a una educación orientada a la práctica, adquiere competencias para liderar proyectos de desarrollo, optimizar procesos y adaptarse con agilidad a los constantes cambios del sector. Además, su perfil se alinea con las exigencias del mercado global, permitiéndole integrarse en equipos multidisciplinares, asumir roles estratégicos y aportar valor en entornos altamente competitivos e impulsados por la tecnología.

¿Buscas desempeñarte como Arquitecto de Aplicaciones Móviles? Lógralo por medio de este programa universitario en solamente meses.

- ♦ **Capacidad de resolución de problemas:** Desarrollo de un pensamiento analítico y estructurado que permite identificar, abordar y resolver desafíos técnicos en el ciclo completo de una aplicación móvil, desde la planificación hasta el mantenimiento post-lanzamiento
- ♦ **Adaptación al cambio tecnológico:** Habilidad para incorporar de manera ágil nuevas herramientas, frameworks y tendencias del ecosistema Android, respondiendo eficazmente a la evolución constante del entorno digital
- ♦ **Trabajo en equipo multidisciplinar:** Integración efectiva en equipos compuestos por diseñadores, programadores, analistas y gestores de producto, favoreciendo la colaboración y la comunicación fluida en entornos ágiles
- ♦ **Gestión del tiempo y autonomía:** Fortalecimiento de la capacidad para organizar tareas, priorizar actividades y cumplir objetivos en proyectos de desarrollo, especialmente en contextos de modalidad online o bajo metodologías *remote-first*





Después de realizar el programa universitario, podrás desempeñar tus conocimientos y habilidades en los siguientes cargos:

- 1. Desarrollador Android:** Crea, prueba y mantiene aplicaciones móviles nativas para dispositivos Android utilizando lenguajes como Kotlin o Java.
- 2. Ingeniero de Software Móvil:** Diseña soluciones técnicas avanzadas dentro del entorno móvil, integrando funcionalidades complejas y asegurando el rendimiento de las aplicaciones.
- 3. Diseñador de Experiencia de Usuario Móvil:** Se enfoca en la creación de interfaces intuitivas y funcionales, mejorando la interacción entre el usuario y la aplicación.
- 4. Especialista en Control de Calidad Móvil:** Realiza pruebas automatizadas y manuales para garantizar la estabilidad, seguridad y usabilidad de las aplicaciones.
- 5. Desarrollador Full Stack:** Combina conocimientos de *backend* y *frontend* para desarrollar aplicaciones completas con integración de servicios en la nube.
- 6. Experto en Proyectos de Aplicaciones Móviles:** Coordina equipos técnicos y creativos, gestionando tiempos, recursos y objetivos en el desarrollo de productos móviles.
- 7. Consultor en Soluciones Móviles:** Asesora a empresas en la implementación de estrategias tecnológicas basadas en aplicaciones Android, adaptadas a sus necesidades de negocio.
- 8. Arquitecto de Software Móvil:** Define la estructura técnica y lógica de las aplicaciones, seleccionando tecnologías y estándares que aseguren escalabilidad y eficiencia.

06

Metodología de estudio

TECH es la primera universidad en el mundo que combina la metodología de los **case studies** con el **Relearning**, un sistema de aprendizaje 100% online basado en la reiteración dirigida.

Esta disruptiva estrategia pedagógica ha sido concebida para ofrecer a los profesionales la oportunidad de actualizar conocimientos y desarrollar competencias de un modo intenso y riguroso. Un modelo de aprendizaje que coloca al estudiante en el centro del proceso académico y le otorga todo el protagonismo, adaptándose a sus necesidades y dejando de lado las metodologías más convencionales.



“

TECH te prepara para afrontar nuevos retos en entornos inciertos y lograr el éxito en tu carrera”

El alumno: la prioridad de todos los programas de TECH

En la metodología de estudios de TECH el alumno es el protagonista absoluto. Las herramientas pedagógicas de cada programa han sido seleccionadas teniendo en cuenta las demandas de tiempo, disponibilidad y rigor académico que, a día de hoy, no solo exigen los estudiantes sino los puestos más competitivos del mercado.

Con el modelo educativo asincrónico de TECH, es el alumno quien elige el tiempo que destina al estudio, cómo decide establecer sus rutinas y todo ello desde la comodidad del dispositivo electrónico de su preferencia. El alumno no tendrá que asistir a clases en vivo, a las que muchas veces no podrá acudir. Las actividades de aprendizaje las realizará cuando le venga bien. Siempre podrá decidir cuándo y desde dónde estudiar.

“

*En TECH NO tendrás clases en directo
(a las que luego nunca puedes asistir)”*



Los planes de estudios más exhaustivos a nivel internacional

TECH se caracteriza por ofrecer los itinerarios académicos más completos del entorno universitario. Esta exhaustividad se logra a través de la creación de temarios que no solo abarcan los conocimientos esenciales, sino también las innovaciones más recientes en cada área.

Al estar en constante actualización, estos programas permiten que los estudiantes se mantengan al día con los cambios del mercado y adquieran las habilidades más valoradas por los empleadores. De esta manera, quienes finalizan sus estudios en TECH reciben una preparación integral que les proporciona una ventaja competitiva notable para avanzar en sus carreras.

Y además, podrán hacerlo desde cualquier dispositivo, pc, tableta o smartphone.

“

El modelo de TECH es asincrónico, de modo que te permite estudiar con tu pc, tableta o tu smartphone donde quieras, cuando quieras y durante el tiempo que quieras”

Case studies o Método del caso

El método del caso ha sido el sistema de aprendizaje más utilizado por las mejores escuelas de negocios del mundo. Desarrollado en 1912 para que los estudiantes de Derecho no solo aprendiesen las leyes a base de contenidos teóricos, su función era también presentarles situaciones complejas reales. Así, podían tomar decisiones y emitir juicios de valor fundamentados sobre cómo resolverlas. En 1924 se estableció como método estándar de enseñanza en Harvard.

Con este modelo de enseñanza es el propio alumno quien va construyendo su competencia profesional a través de estrategias como el *Learning by doing* o el *Design Thinking*, utilizadas por otras instituciones de renombre como Yale o Stanford.

Este método, orientado a la acción, será aplicado a lo largo de todo el itinerario académico que el alumno emprenda junto a TECH. De ese modo se enfrentará a múltiples situaciones reales y deberá integrar conocimientos, investigar, argumentar y defender sus ideas y decisiones. Todo ello con la premisa de responder al cuestionamiento de cómo actuaría al posicionarse frente a eventos específicos de complejidad en su labor cotidiana.



Método Relearning

En TECH los *case studies* son potenciados con el mejor método de enseñanza 100% online: el *Relearning*.

Este método rompe con las técnicas tradicionales de enseñanza para poner al alumno en el centro de la ecuación, proveyéndole del mejor contenido en diferentes formatos. De esta forma, consigue repasar y reiterar los conceptos clave de cada materia y aprender a aplicarlos en un entorno real.

En esta misma línea, y de acuerdo a múltiples investigaciones científicas, la reiteración es la mejor manera de aprender. Por eso, TECH ofrece entre 8 y 16 repeticiones de cada concepto clave dentro de una misma lección, presentada de una manera diferente, con el objetivo de asegurar que el conocimiento sea completamente afianzado durante el proceso de estudio.

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



Un Campus Virtual 100% online con los mejores recursos didácticos

Para aplicar su metodología de forma eficaz, TECH se centra en proveer a los egresados de materiales didácticos en diferentes formatos: textos, vídeos interactivos, ilustraciones y mapas de conocimiento, entre otros. Todos ellos, diseñados por profesores cualificados que centran el trabajo en combinar casos reales con la resolución de situaciones complejas mediante simulación, el estudio de contextos aplicados a cada carrera profesional y el aprendizaje basado en la reiteración, a través de audios, presentaciones, animaciones, imágenes, etc.

Y es que las últimas evidencias científicas en el ámbito de las Neurociencias apuntan a la importancia de tener en cuenta el lugar y el contexto donde se accede a los contenidos antes de iniciar un nuevo aprendizaje. Poder ajustar esas variables de una manera personalizada favorece que las personas puedan recordar y almacenar en el hipocampo los conocimientos para retenerlos a largo plazo. Se trata de un modelo denominado *Neurocognitive context-dependent e-learning* que es aplicado de manera consciente en esta titulación universitaria.

Por otro lado, también en aras de favorecer al máximo el contacto mentor-alumno, se proporciona un amplio abanico de posibilidades de comunicación, tanto en tiempo real como en diferido (mensajería interna, foros de discusión, servicio de atención telefónica, email de contacto con secretaría técnica, chat y videoconferencia).

Asimismo, este completísimo Campus Virtual permitirá que el alumnado de TECH organice sus horarios de estudio de acuerdo con su disponibilidad personal o sus obligaciones laborales. De esa manera tendrá un control global de los contenidos académicos y sus herramientas didácticas, puestas en función de su acelerada actualización profesional.



La modalidad de estudios online de este programa te permitirá organizar tu tiempo y tu ritmo de aprendizaje, adaptándolo a tus horarios”

La eficacia del método se justifica con cuatro logros fundamentales:

1. Los alumnos que siguen este método no solo consiguen la asimilación de conceptos, sino un desarrollo de su capacidad mental, mediante ejercicios de evaluación de situaciones reales y aplicación de conocimientos.
2. El aprendizaje se concreta de una manera sólida en capacidades prácticas que permiten al alumno una mejor integración en el mundo real.
3. Se consigue una asimilación más sencilla y eficiente de las ideas y conceptos, gracias al planteamiento de situaciones que han surgido de la realidad.
4. La sensación de eficiencia del esfuerzo invertido se convierte en un estímulo muy importante para el alumnado, que se traduce en un interés mayor en los aprendizajes y un incremento del tiempo dedicado a trabajar en el curso.

La metodología universitaria mejor valorada por sus alumnos

Los resultados de este innovador modelo académico son constatables en los niveles de satisfacción global de los egresados de TECH.

La valoración de los estudiantes sobre la calidad docente, calidad de los materiales, estructura del curso y sus objetivos es excelente. No en valde, la institución se convirtió en la universidad mejor valorada por sus alumnos según el índice global score, obteniendo un 4,9 de 5.

Accede a los contenidos de estudio desde cualquier dispositivo con conexión a Internet (ordenador, tablet, smartphone) gracias a que TECH está al día de la vanguardia tecnológica y pedagógica.

Podrás aprender con las ventajas del acceso a entornos simulados de aprendizaje y el planteamiento de aprendizaje por observación, esto es, Learning from an expert.



Así, en este programa estarán disponibles los mejores materiales educativos, preparados a conciencia:



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 que creará nuestra manera de trabajo online, con las técnicas más novedosas que nos permiten ofrecerte una gran calidad, en cada una de las piezas que pondremos a tu servicio.



Prácticas de habilidades y competencias

Realizarás 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.



Resúmenes interactivos

Presentamos los contenidos de manera atractiva y dinámica en píldoras multimedia que incluyen audio, vídeos, imágenes, esquemas y mapas conceptuales con el fin de afianzar el conocimiento.

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



Lecturas complementarias

Artículos recientes, documentos de consenso, guías internacionales... En nuestra biblioteca virtual tendrás acceso a todo lo que necesitas para completar tu capacitación.





Case Studies

Completarás una selección de los mejores *case studies* de la materia. Casos presentados, analizados y tutorizados por los mejores especialistas del panorama internacional.



Testing & Retesting

Evaluamos y reevaluamos periódicamente tu conocimiento a lo largo del programa. Lo hacemos sobre 3 de los 4 niveles de la Pirámide de Miller.



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 nuestras futuras decisiones difíciles.



Guías rápidas de actuación

TECH ofrece los contenidos más relevantes del curso en forma de fichas o guías rápidas de actuación. Una manera sintética, práctica y eficaz de ayudar al estudiante a progresar en su aprendizaje.



07

Cuadro docente

El cuadro docente de este programa está conformado por profesionales en activo con amplia experiencia en el Desarrollo de Aplicaciones Android y en la industria tecnológica global. Gracias a su trayectoria en empresas líderes y proyectos de innovación digital, aportan una visión práctica, actualizada y orientada al mercado real. Además, combinan el dominio técnico con una sólida capacidad para transmitir conocimientos complejos de manera clara y aplicable. Esta integración entre experiencia profesional y excelencia académica garantiza un acompañamiento riguroso, que responde a las exigencias del sector y a los estándares de calidad más altos en el ámbito del desarrollo móvil.



stylów Wydajność

description" style="clear: both;"

ft: 5px;"></div>

label>

ng-t

“

Aprende de un equipo docente compuesto por expertos en Desarrollo de Aplicaciones Android, que aporta una visión orientada a los desafíos reales del mercado digital”

Director Invitado Internacional

Colin Lee es un exitoso **desarrollador de aplicaciones móviles** especializado en **código nativo de Android**, cuya influencia se extiende a nivel internacional. El experto es una autoridad en el **área de Twin Cities** y en el **manejo de Kotlin**. Uno de sus aportes más recientes fue demostrar, en código en vivo, cómo **construir rápidamente un navegador** utilizando el mencionado lenguaje de programación y los componentes de navegador de código abierto de Mozilla para Android.

Además, sus aplicaciones han estado ligadas a empresas de gran relevancia mundial, por ejemplo, fue encargado de crear **soluciones digitales** para **Pearson**, una de las editoriales más grandes del panorama internacional. Asimismo, desarrolló un **grabador de vídeo** de Android de bajo nivel para la *startup* Flipgrid, luego adquirido por Microsoft.

También construyó una exitosa VPN de Android para un gran **cliente del mundo de la consultoría**. A su vez, es el creador de una herramienta para la gestión de cargas que implementa la trasnacional **Amazon** para facilitar el trabajo de sus camioneros contratados. Por otro lado, ha ayudado a construir las **versiones móviles del navegador Firefox** para Mozilla.

Hoy en día, realiza trabajos como contratista, incluyendo **revisiones de código** y **comprobaciones de seguridad**. Su impacto en el desarrollo de aplicaciones móviles y su experiencia a lo largo de los años lo convierten en una figura destacada en el ámbito de la tecnología global.



D. Lee, Colin

- Ingeniero Senior de Android para Meetup, Mineápolis, Estados Unidos
- Director en ColinTheShots LLC
- Ingeniero de Software Android para Specto Inc.
- Ingeniero Sénior de Android para Mozilla
- Ingeniero Desarrollador de Software para Amazon
- Ingeniero de Aplicaciones Móviles para Flipgrid
- Especialista de Configuración de Software para Pearson VUE
- Licenciado por la Universidad de Florida

“

Gracias a TECH podrás aprender con los mejores profesionales del mundo”

Dirección



D. Olalla Bonal, Martín

- Gerente Senior de Práctica de *Blockchain* en EY
- Especialista Técnico Cliente *Blockchain* para IBM
- Director de Arquitectura para Blocknitive
- Coordinador de Equipo en Bases de Datos Distribuidas no Relacionales para WedoIT, Subsidiaria de IBM
- Arquitecto de Infraestructuras en Bankia
- Responsable del Departamento de Maquetación en T-Systems
- Coordinador de Departamento para Bing Data España SL



Profesores

D. Gozalo Fernández, Juan Luis

- ◆ Gerente de Productos basados en Blockchain para Open Canarias
- ◆ Director 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. Grillo Hernández, José Enrique

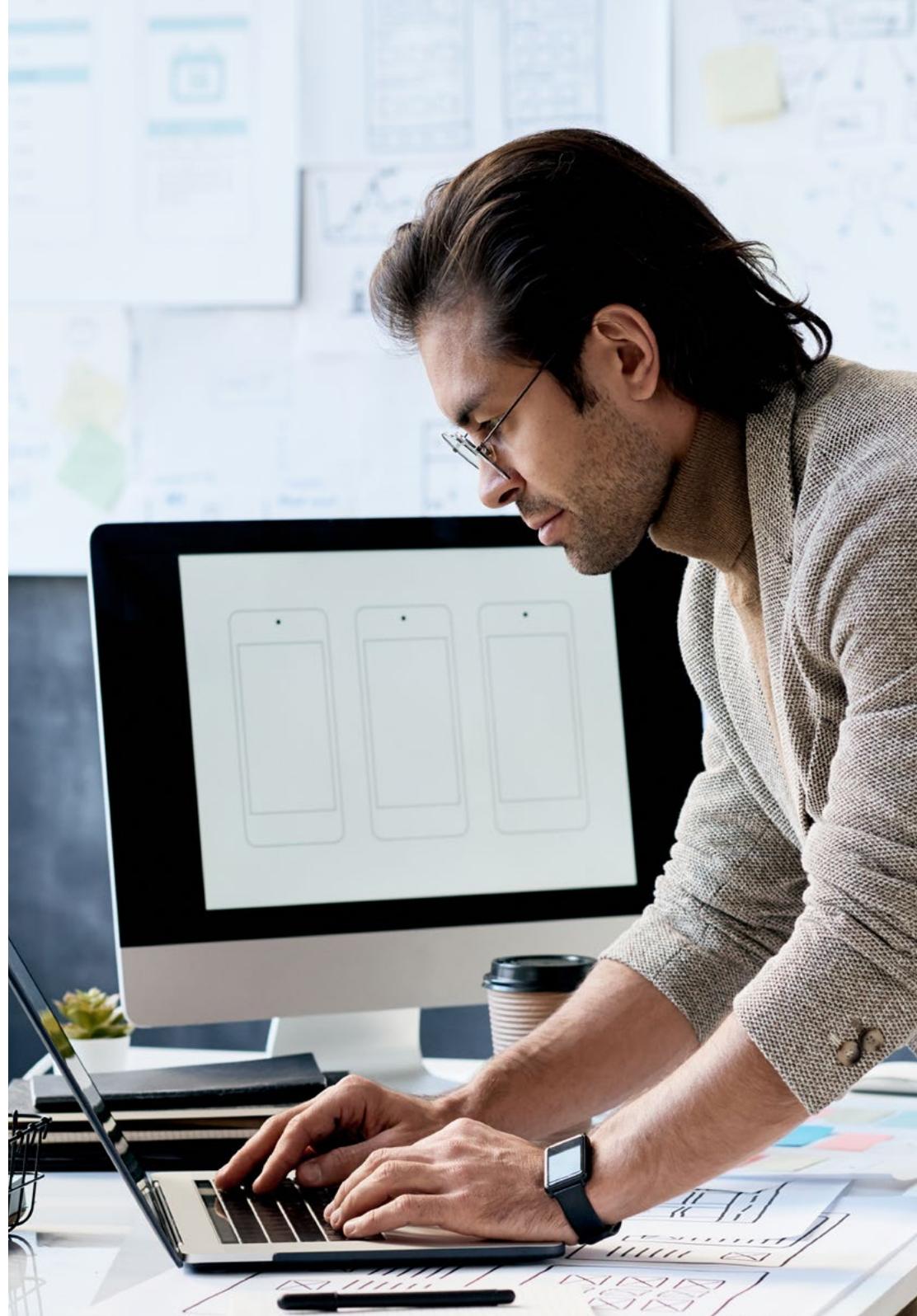
- ◆ Desarrollador de aplicaciones y analista tecnológico
- ◆ Desarrollador Senior de Aplicaciones Móviles en Globant
- ◆ Desarrollador Android en Plexus Tech
- ◆ Desarrollador Senior Android en RoadStr
- ◆ Desarrollador Senior Mobile en Avantgarde IT-Information Technology Services
- ◆ Líder de Proyecto en Smartdoss
- ◆ Desarrollado en Educatablet
- ◆ Analista de Tecnología en Corporate Mobile Solutions
- ◆ Máster en Ingeniería de Sistema por la Universidad Simón Bolívar

Dr. Peralta Martín-Palomino, Arturo

- ♦ CEO y CTO en Prometeus Global Solutions
- ♦ CTO en Korporate Technologies
- ♦ CTO en AI Shepherds GmbH
- ♦ Consultor y Asesor Estratégico Empresarial en Alliance Medical
- ♦ Director de Diseño y Desarrollo en DocPath
- ♦ Doctor en Ingeniería Informática por la Universidad de Castilla-La Mancha
- ♦ Doctor en Economía, Empresas y Finanzas por la Universidad Camilo José Cela
- ♦ Doctor en Psicología por la Universidad de Castilla-La Mancha
- ♦ Máster en Executive MBA por la Universidad Isabel I
- ♦ Máster en Dirección Comercial y Marketing por la Universidad Isabel I
- ♦ Máster Experto en Big Data por Formación Hadoop
- ♦ Máster en Tecnologías Informáticas Avanzadas por la Universidad de Castilla-La Mancha
- ♦ Miembro: Grupo de Investigación SMILE

Dña. Martínez Cerrato, Yésica

- ♦ Responsable de Capacitaciones Técnicas en Securitas Seguridad España
- ♦ Especialista en Educación, Negocios y Marketing
- ♦ *Product Manager* en Seguridad Electrónica en Securitas Seguridad España
- ♦ Analista de Inteligencia Empresarial en Ricopia Technologies
- ♦ Técnico Informático y Responsable de Aulas informáticas OTEC en la Universidad de Alcalá de Henares
- ♦ Colaboradora en la Asociación ASALUMA
- ♦ Grado en Ingeniería Electrónica de Comunicaciones en la Escuela Politécnica Superior, Universidad de Alcalá de Henares



D. Natal Rodríguez, Daniel

- ♦ Ingeniero Informático especializado en Appian
- ♦ Desarrollador Appian en Vass, consultora especializada en soluciones IT
- ♦ Desarrollador de Plataformas en Telefónica
- ♦ Programador Java en Babel
- ♦ Grado en Ingeniería Informática por la Escuela Técnica Superior de Ingenieros Informáticos en la Universidad Politécnica de Madrid

Dña. Portalatín Romero, Isabel

- ♦ Ingeniera Informática
- ♦ Responsable de ofertas en el área de Informática a diferentes Organismos Públicos y Privados
- ♦ Docente online en disímiles programas de Formación Profesional
- ♦ Ingeniería Técnica en Informática De Gestión por la Escuela Universitaria Politécnica de Informática de la Universidad de Extremadura

D. Arranz Torres, Héctor

- ♦ Director de Proyectos *Software* en Ezenit
- ♦ Miembro del Equipo de Operaciones de *Cink Venturing*
- ♦ Docente en cursos relacionados con el desarrollo de proyectos digitales
- ♦ Graduado en Ingeniería Informática por la Universidad Complutense de Madrid
- ♦ MBA en Dirección y Gestión de Empresas por The Power MBA

D. Guerrero Díaz-Pintado, Arturo

- ♦ Director de Experiencia de Clientes para IBM
- ♦ Ingeniero Técnico de Preventa a través de *Watson Customer Engagement* portfolio
- ♦ Ingeniero de redes de I+D en Telefónica
- ♦ Licenciado en Ingeniería Superior de Telecomunicación por la Universidad de Alcalá y la *Danish Technical University*

D. Centeno Martín-Romero, Alfonso

- ♦ Consultor de Ciberseguridad en Deloitte
- ♦ Becario en el departamento de Planificación Comercial & Business Intelligence en Telefónica Tech
- ♦ Becario en B2B Market Intelligence
- ♦ Becario en el departamento de Derivados Financieros y Análisis de Renta Variable en Ahorro Corporación Financiera
- ♦ Doble Grado en Administración y Dirección de Empresas & Relaciones Internacionales en la Universidad Pontificia de Comillas
- ♦ Bootcamp en Ciberseguridad en el centro de estudios tecnológicos Ironhack

D. Noguera Rodríguez, Pablo

- ♦ Ingeniero de software aplicado a soluciones *Blockchain* en EY
- ♦ Desarrollador de aplicaciones para móviles en Bitnovo
- ♦ Desarrollador de aplicaciones nativas para iOS en Umani y Stef
- ♦ Programador autónomo y Creador de Aviaze App, en colaboración con Starman Aviation

08

Titulación

El Máster Título Propio en Desarrollo de Aplicaciones Android garantiza, además de la capacitación más rigurosa y actualizada, el acceso a un título de Máster Propio expedido por TECH Universidad.



“

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

Este **Máster Título Propio en Desarrollo de Aplicaciones Android** contiene el programa universitario más completo y actualizado del mercado.

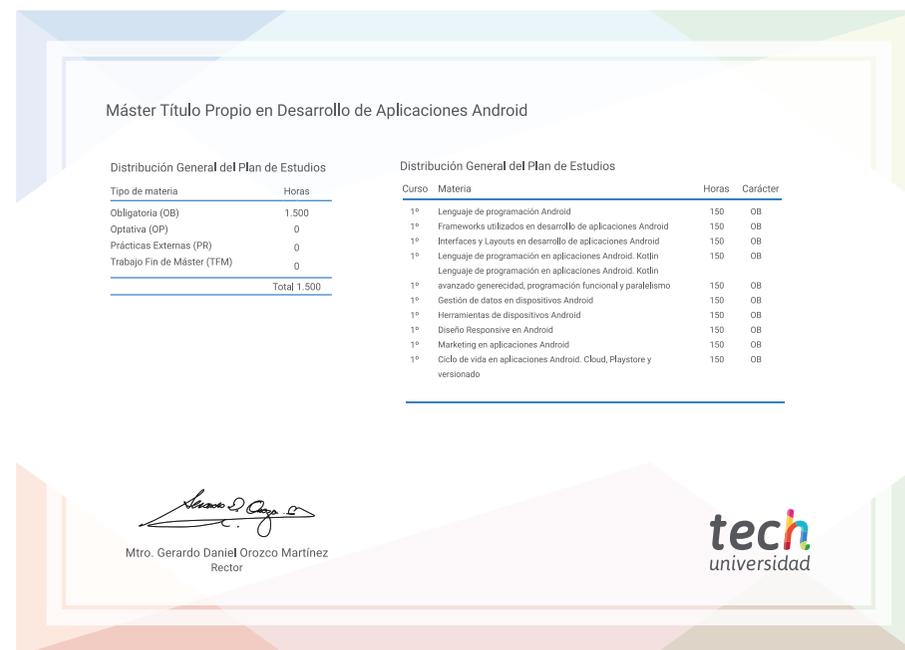
Tras la superación de la evaluación, el alumno recibirá por correo postal* con acuse de recibo su correspondiente título de **Máster Propio** emitido por **TECH Universidad**.

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

Título: **Máster Título Propio en Desarrollo de Aplicaciones Android**

Modalidad: **No escolarizada (100% en línea)**

Duración: **12 meses**



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



Máster Título Propio Desarrollo de Aplicaciones Android

- » Modalidad: No escolarizada (100% en línea)
- » Duración: 12 meses
- » Titulación: TECH Universidad
- » Horario: a tu ritmo
- » Exámenes: online

