

# Mestrado Próprio

## Desenvolvimento de Aplicações Android



## Mestrado Próprio Desenvolvimento de Aplicações Android

- » Modalidade: **online**
- » Duração: **12 meses**
- » Certificação: **TECH Universidade Tecnológica**
- » Créditos: **60 ECTS**
- » Tempo Dedicado: **16 horas/semana**
- » Horário: **ao seu próprio ritmo**
- » Exames: **online**

Acesso ao site: [www.techtitute.com/pt/informatica/mestrado-proprio/mestrado-proprio-desenvolvimento-aplicacoes-android](http://www.techtitute.com/pt/informatica/mestrado-proprio/mestrado-proprio-desenvolvimento-aplicacoes-android)

# Índice

01

Apresentação

---

*pág. 4*

02

Objetivos

---

*pág. 8*

03

Competências

---

*pág. 14*

04

Direção do curso

---

*pág. 18*

05

Estrutura e conteúdo

---

*pág. 22*

06

Metodologia

---

*pág. 34*

07

Certificação

---

*pág. 42*

# 01

# Apresentação

Atualmente, existem mais de 3 mil milhões de dispositivos Android no mundo e, como resultado, a indústria de desenvolvimento de aplicações móveis tem experimentado um rápido crescimento. Os relatórios do mercado laboral confirmam a crescente procura de perfis de especialistas na conceção de aplicações móveis, valorizando os conhecimentos especializados em todo o ciclo de vida do desenvolvimento, implantação e monetização de aplicações. Não é fácil encontrar profissionais que conheçam a linguagem de programação de Android, arquiteturas e interfaces de utilizador. É por isso que este programa foi concebido para fornecer ao interessado todos os conhecimentos para impulsionar a sua carreira com um perfil especializado. Em modo totalmente online e alcançando a sua certificação em 12 meses.



“

*Especialize-se nos aspetos chave de Customer Experience para desenvolver aplicações Android para que tenham uma maior entrada no mercado”*

Com o programa de Desenvolvimento de Aplicações Android, os estudantes poderão especializar-se numa das tecnologias padrão com a maior projeção profissional destes tempos. Adquirirão os conhecimentos necessários para serem autônomos no desenvolvimento de uma aplicação para dispositivos neste sistema operativo, desde a conceção até à produção. E irão adquirir conhecimentos especializados sobre o ciclo de vida das aplicações Android, a fim de trabalharem de forma mais eficiente.

Para este fim, a TECH tem uma equipa de profissionais altamente qualificados com a experiência necessária para partilhar casos reais que levam a um processo de aprendizagem mais profundo e eficiente. Profissionais em programação, desenvolvimento de aplicações e novas tecnologias que irão acompanhar o profissional no processo de estudo. A partir de uma plataforma segura, com conteúdo interativo, distribuído em vídeos, gráficos, exemplos baseados em problemas, material audiovisual e com uma grande comunidade para discutir as questões levantadas.

Assim, o estudante pode estar relacionado com áreas tais como a informática ou web ou software, desenvolvimento de negócios, Marketing e vendas, consultoria, tecnologia móvel multiplataforma, empresas, automóvel, domótica, Internet das Coisas (IoT), bancos e drones. Poderão aplicar os seus conhecimentos em benefício de qualquer um destes setores.

Além disso, a TECH tem a melhor metodologia de estudo baseada no *Relearning*, e disponível inteiramente online, eliminando a necessidade de frequentar uma aula ou as limitações típicas de um horário pré-determinado. Assim, em 12 meses, os estudantes adquirirão a base necessária para realizar tarefas de consultoria em todos os aspetos relacionados com a tecnologia móvel em Android ou para desenvolver o seu próprio negócio com base no desenvolvimento de aplicações.

Este **Mestrado Próprio em Desenvolvimento de Aplicações Android** conta com o conteúdo educacional mais completo e atualizado do mercado. As suas principais características são:

- ◆ O desenvolvimento de casos práticos apresentados por especialistas em computação paralela e distribuída
- ◆ O conteúdo gráfico, esquemático e eminentemente prático do livro fornece informações práticas sobre as disciplinas que são essenciais para a prática profissional
- ◆ Exercícios práticos onde o processo de autoavaliação pode ser levado a cabo a fim de melhorar a aprendizagem
- ◆ A sua ênfase especial em metodologias inovadoras
- ◆ Lições teóricas, questões ao especialista e trabalhos de reflexão individual
- ◆ A disponibilidade de acesso ao conteúdo a partir de qualquer dispositivo fixo ou portátil com ligação à Internet



*Com o Mestrado Próprio em Desenvolvimento de Aplicações Android pode especializar-se numa das tecnologias padrão com a maior projeção profissional da atualidade”*



*Adquirirá os conhecimentos necessários para ser autónomo no desenvolvimento de uma aplicação para dispositivos Android, desde a conceção até à produção”*

O corpo docente do curso inclui profissionais do setor que trazem a sua experiência profissional para esta capacitação, para além de especialistas reconhecidos de sociedades de referência e universidades de prestígio.

Graças ao seu conteúdo multimédia, desenvolvido com a mais recente tecnologia educacional, o profissional terá acesso a uma aprendizagem situada e contextual, ou seja, um ambiente de simulação que proporcionará um programa imersivo programado para se formar em situações reais.

A conceção deste programa baseia-se na Aprendizagem Baseada nos Problemas, através da qual o instrutor deve tentar resolver as diferentes situações da atividade profissional que surgem ao longo do curso académico. Para tal, contará com a ajuda de um sistema inovador de vídeo interativo desenvolvido por especialistas reconhecidos.

*Aproveite ao máximo o ambiente tecnológico de vanguarda que a TECH lhe oferece. Um programa 100% em online, onde se estabelecerá uma rede com especialistas destacados.*

*Forneça soluções eficientes para os seus empregadores ou para a sua própria empresa. Com um bom domínio dos fundamentos do Desenvolvimento de Aplicações Android.*



Android Studio  
Version 3.3

Android Studio project

# 02 Objetivos

O processo de crescimento provocado pelo desenvolvimento imparável de novas tecnologias e dispositivos é notório. Avanços que requerem a capacitação contínua de profissionais de informática e engenharia. O mercado não fica parado e, para acompanhar as necessidades, é essencial que os profissionais estejam na vanguarda dos conceitos e estilos que estão a ser abordados. Este programa em Desenvolvimento de Aplicações Android ensina todas as ferramentas e fundamentos necessários para crescer nesta carreira e fornecer soluções eficazes no ambiente profissional.





“

*Defina os seus próprios desafios dentro do ecossistema da tecnologia Android. Comece a tomar medidas para uma carreira de sucesso como criador de aplicações”*



## Objetivos gerais

---

- ◆ Determinar os elementos estruturais de um sistema Android
- ◆ Analisar as diferenças entre os diferentes *Frameworks* de Desenvolvimento, pontos fortes e fracos
- ◆ Estabelecer como os estilos e temas são criados, incluindo os de Realidade Virtual, através dos *Layouts*
- ◆ Analisar as diferentes formas de gestão do fluxo do código em Kotlin
- ◆ Desenvolver capacidades avançadas e melhores práticas em Desenvolvimento de Aplicações na linguagem de programação Kotlin
- ◆ Desenvolver uma metodologia para uma gestão ótima dos dados no dispositivo
- ◆ Analisar os casos de utilização de dispositivos Android no mercado
- ◆ Dominar os elementos de um design responsivo e superar os desafios do mesmo
- ◆ Compilar as diferentes fases de um ciclo contínuo de integração centrado no desenvolvimento em Android

“

*Torne-se um especialista no Desenvolvimento de Aplicações Android. Inscreva-se agora neste programa e especialize-se em 12 meses numa modalidade 100% online”*





## Objetivos específicos

---

### Módulo 1. Linguagem de programação Android

- ◆ Examinar o núcleo Linux e a máquina virtual na base Android
- ◆ Analisar as bibliotecas nativas do sistema
- ◆ Estabelecer os pontos fortes do Android sobre outras plataformas
- ◆ Determinar os elementos de uma aplicação Android
- ◆ Apresentar as versões Android e as suas melhorias
- ◆ Avaliar o mercado de aplicações Android
- ◆ Fundamentar a evolução futura do Android

### Módulo 2. *Frameworks* utilizadas em desenvolvimento de aplicações Android

- ◆ Analisar o *Framework* de Android Core
- ◆ Desenvolver outras *Framework* utilizadas para o desenvolvimento de aplicações Android
- ◆ Implementar bibliotecas com *Graddl*
- ◆ Estabelecer os *Frameworks* para ligar a uma API
- ◆ Gerar conhecimentos especializados sobre arquiteturas como MVP
- ◆ Esclarecer os prós e os contras de MVP e MVVM

### Módulo 3. Interfaces e *Layouts* no desenvolvimento de aplicações Android

- ◆ Apresentar o modelo do ciclo de vida das vistas em Android
- ◆ Examinar os atributos mais importantes de um design visual (*Layouts*)
- ◆ Analisar os designs (*Layouts*) disponíveis
- ◆ Gerar um design (*Layouts*) reutilizável
- ◆ Determinar como utilizar recursos alternativos
- ◆ Identificar as diferenças na utilização destes componentes em comparação com outros sistemas de programação
- ◆ Estabelecer o potencial e o uso do ficheiro `AndroidManifest.xml`

### Módulo 4. Linguagem de programação em aplicações Android. Kotlin

- ◆ Desenvolver a linguagem de programação Kotlin
- ◆ Compilar as características e capacidades diferenciais da linguagem Kotlin
- ◆ Exame do modelo básico de execução de um programa em Kotlin
- ◆ Analisar a sintaxe da linguagem e a estrutura de um programa em Kotlin
- ◆ Concretizar o modelo de tipos e variáveis em Kotlin
- ◆ Estabelecer as diferentes formas de gestão de fluxo do código em Kotlin
- ◆ Determinação do modelo de classes, coleções e objetos em Kotlin
- ◆ Gerar conhecimentos especializados sobre o modelo sucessório em Kotlin
- ◆ Concretizar o modelo de gestão de exceções e tipos nulos em Kotlin

### Módulo 5. Linguagem de programação em aplicações Android. Kotlin avançado genericidade, programação funcional e paralelismo

- ◆ Examinar o modelo de genericidade covariante e contravariante em Kotlin
- ◆ Analisar o modelo de programação funcional com Lambdas de Kotlin
- ◆ Concretizar as funções de ordem superior de Kotlin
- ◆ Desenvolver as extensões e os objetos de companhia em Kotlin
- ◆ Examinar a utilização do padrão *Delegation* em Kotlin
- ◆ Compilar as anotações e a reflexão em Kotlin
- ◆ Aprofundar o modelo de *Testing* em Kotlin
- ◆ Estabelecer os diferentes modelos de programação assíncrona disponíveis em Kotlin
- ◆ Determinar o modelo de corrotina de Kotlin
- ◆ Compilar as diferentes bibliotecas e ferramentas utilitárias do ecossistema Kotlin

### Módulo 6. Gestão de dados em dispositivos Android

- ◆ Analisar as diferentes técnicas de gestão de dados em Android
- ◆ Propor métodos para uma utilização ótima dos dados no dispositivo
- ◆ Identificar as ferramentas necessárias para a otimização dos dados
- ◆ Examinar as características do JSON e XML para a gestão de dados em Android
- ◆ Avaliar questões gerais de sistemas distribuídos aplicáveis ao mundo das aplicações de dispositivos móveis
- ◆ Determinar a utilização da biblioteca Room como abstração para a utilização do SQLite no Android e as suas vantagens e desvantagens
- ◆ Definir as permissões de segurança necessárias para a gestão de dados em qualquer uma das técnicas utilizadas em Android

### Módulo 7. Ferramentas de dispositivos Android

- ◆ Compilar as ferramentas mais avançadas na gestão diária
- ◆ Avaliar ferramentas de controlo de dispositivos Android
- ◆ Demonstrar as vantagens de utilizar Android em Drones
- ◆ Concretizar as funcionalidades de ferramentas CRM
- ◆ Demonstrar os benefícios dos dispositivos Android nas plataformas bancárias
- ◆ Analisar as possibilidades da plataforma IoT e Android
- ◆ Examinar ferramentas de eficiência de processo em Android

### Módulo 8. Design *Responsive* em Android

- ◆ Analisar os principais elementos de um design
- ◆ Definir uma metodologia de design de elementos visuais e uma metodologia de design de ecrãs
- ◆ Solucionar os vários problemas encontrados no desenvolvimento multidispositivo
- ◆ Solucionar os vários problemas encontrados no desenvolvimento multidispositivo
- ◆ Examinar *Frameworks* alternativos ao desenvolvimento nativo responsive
- ◆ Gerar uma metodologia própria para desenvolver aplicações utilizando as melhores práticas para a visualização multidispositivo desde o início do projeto

### Módulo 9. Marketing em aplicações Android

- ◆ Analisar as novas metodologias centradas no utilizador
- ◆ Determinar como a Inteligência Artificial levou o CX ao nível seguinte
- ◆ Estabelecer a importância da acessibilidade e da mobilidade
- ◆ Desenvolver as técnicas mais avançadas de análise sessão e de comportamento
- ◆ Estabelecer objetivos de micropersonalização no decurso do *Journey* de utilizador
- ◆ Compilar as novas metodologias para um ambiente em mudança e animado
- ◆ Propor técnicas para a criação de protótipos

### Módulo 10. Ciclo de vida em aplicações Android. Cloud, Playstore e versioning

- ◆ Concretizar os benefícios de adotar um modelo automatizado de implementação de versões
- ◆ Estabelecer as diferenças entre integração contínua, entrega e a implementação contínuas
- ◆ Definir as principais características de DevOps
- ◆ Avaliar algumas das ferramentas-chave para implementar pipelines de CI/CD
- ◆ Esclarecer os fatores essenciais para o desenvolvimento de aplicações preparadas para apoiar os processos CI/CD
- ◆ Examinar as Tecnologias de Contentores como pilar fundamental do CI/CD
- ◆ Identificar práticas, casos de utilização, tecnologias e ferramentas do ecossistema de CI/CD

# 03

## Competências

Não é fácil encontrar perfis que conheçam a linguagem de programação Android, arquiteturas e interfaces de utilizador, dependentes do tipo de dispositivo em que a aplicação é executada. Com o Mestrado Próprio em Desenvolvimento de Aplicações Android, os estudantes poderão especializar-se numa das tecnologias padrão com a maior projeção profissional destes tempos. Adquirirá os conhecimentos necessários para ser autónomo no desenvolvimento de uma aplicação para dispositivos Android, desde a conceção até à produção. Adquirirão conhecimentos especializados sobre o ciclo de vida das aplicações Android, a fim de trabalharem mais eficientemente como criadores de aplicações para dispositivos móveis.



“

*Com este Mestrado Próprio irá impulsionar a sua carreira nos negócios ou por conta própria. Desenvolvendo aplicações que tenham sucesso no mercado”*



## Competências gerais

- ◆ Desenvolver aplicações móveis para dispositivos com sistema operativo Android
- ◆ Dominar os elementos estruturais de um sistema Android
- ◆ Analisar os diversos *Frameworks* para ser utilizados por múltiplas arquiteturas
- ◆ Definir as propriedades de uma interface do ponto de vista da usabilidade
- ◆ Desenvolver conhecimentos especializados sobre a linguagem de programação Kotlin e o contexto da sua aparição
- ◆ Gerir o *Content Provider* para o intercâmbio de dados e a sua segurança
- ◆ Abordar as ferramentas empresariais em Android: a sua eficiência, utilidade
- ◆ Analisar os elementos de um design responsive
- ◆ Explorar as novas metodologias centradas no utilizador
- ◆ Ajustar o sistema de Pipelines automatizados ao ecossistema Android

“

*Compilar todas estas competências no seu perfil profissional irá acrescentar valor às suas propostas e torná-lo mais atrativo para o mercado de trabalho”*







## Competências específicas

---

- ◆ Utilizar recursos alternativos analisando os designs (*Layouts*) disponíveis
- ◆ Desenvolver as diferentes formas de gestão de fluxo do código em Kotlin e as suas capacidades
- ◆ Melhorar a produtividade das aplicações através das capacidades diferenciais da linguagem Kotlin
- ◆ Desenvolver as extensões e os objetos de acompanhantes em Kotlin
- ◆ Utilizar as bases de dados ou serviços de rede para criar ficheiros em diferentes formatos
- ◆ Gerir a biblioteca Room como abstração para utilização de SQLite em Android
- ◆ Avaliar as ferramentas de controlo de dispositivos Android, analisando as possibilidades da plataforma IoT e Android
- ◆ Adotar um design responsive para proporcionar experiências de utilização mais satisfatórias
- ◆ Combinar técnicas de prototipagem e *Wireframing* com as novas tecnologias disruptivas tais como a Inteligência Artificial e a Internet das Coisas

# 04

## Direção do curso

Este programa de Desenvolvimento de Aplicações Android é liderado por professores com vastos conhecimentos e experiência em novas tecnologias, arquitetura de soluções e infraestrutura digital, especialistas em programação Android e criadores de aplicações. Os seus extensos CVs oferecem uma garantia da qualidade do conteúdo selecionado para este programa. Apostando na otimização do processo de aprendizagem dos profissionais que procuram neste espaço a contribuição que necessitam para o seu sucesso profissional.



“

*Profissionais especialistas no desenvolvimento de aplicações e na área tecnológica acompanhá-lo-ão ao longo de todo o processo de estudo, fornecendo-lhe o melhor conteúdo”*

## Direção



### Sr. Martín Olalla Bonal

- ◆ Atual *Blockchain Technical Specialist* em IBM SPGI
- ◆ Técnico em eletrónica Digital
- ◆ Arquiteto *Blockchain*
- ◆ Arquiteto de Infraestrutura na Banca
- ◆ Formação *Hyperledger Fabric* para empresas
- ◆ Formação *Blockchain* orientado para negócio em empresas
- ◆ Gestão de projetos e implementação de soluções
- ◆ Mais de 25 anos de experiência no setor IT

## Professores

### Sr. Arturo Guerrero Díaz-Pintado

- ◆ Consultor de serviços profissionais que trabalha com organizações de vanguarda na Europa, Médio Oriente e América Latina desde a IBM
- ◆ Destaca colaborações em reconhecidas universidades e centros de ensino superior em temas relacionados com a tecnologia, tais como *Inteligência Artificial, Internet of Things, Cloud, Customer Experience e Digital Transformation*
- ◆ Engenheiro Técnico de Pré-venda através de *Watson Customer Engagement* portfolio (soluções de Marketing e Customer Experience) em Espanha, Portugal, Grécia e Israel na IBM
- ◆ Engenheiro de redes de I+D na Telefónica
- ◆ Licenciado em Engenharia Superior de Telecomunicações pela Universidade de Alcalá e pela *Danish Technical University*

### Sr. Pablo Villot Guisán

- ◆ Arquiteto Cloud, de Soluções Exponenciais e Subject Mater Expert *Blockchain* em KPMG
- ◆ Arquiteto Cloud, de Soluções Exponenciais e Subject Mater Expert *Blockchain* e de Integração em Everis
- ◆ Desenvolvedor e Responsável Técnico de aplicações web e de escritório para a área de Logística Comercial da Inditex, Connectis
- ◆ Licenciado em Engenharia Informática na Universidade da Corunha
- ◆ Certificado Microsoft na MSCA: *Cloud Platform*

**Sr. Pablo Noguera Rodríguez**

- ◆ Desenvolvedor de Apps Nativas (iOS & Android) - Starman Aviation (Aviaze App)
- ◆ Programador de Apps Nativas (iOS) - Stef (Mtrack App)
- ◆ Programador de Apps Nativas (iOS & Android) - Bitnovo (Bitnovo App)
- ◆ Programador Especialista em Java: JSE, JEE e Android - Ilabora Formación
- ◆ Programação de Aplicaciones en Android - EOI – Madrid

**Sr. Javier Pérez Rico**

- ◆ Líder técnico atual em Android na Nologis
- ◆ Líder técnico em Android na Seekie
- ◆ Programador Android na Gowex-Ideup
- ◆ Programador Junior Android na Tecnocom
- ◆ Orador no II Simpósio iTest, E@tic2011
- ◆ Engenheiro em Técnicas Informática de sistemas pela Universidade Complutense de Madrid
- ◆ Mestrado em Investigação pela Universidade Complutense de Madrid

**Sr. Carlos Jiménez Pérez**

- ◆ Desenvolvedor sénior Android no OnTheSpot - Telefónica Tech
- ◆ Engenheira em Automação e Eletrónica
- ◆ Mestrado em Engenharia de Sistemas Eletrónicos e Aplicações
- ◆ Professora Assistente na Universidade Carlos III de Madrid

**Sr. Juan Luis Gozalo Fernández**

- ◆ Web 3.0 Blockchain Product Manager
- ◆ Professor Associado em DevOps e em Blockchain em UNIR
- ◆ Exdiretor Blockchain DevOps em Alastria
- ◆ Diretor Desenvolvimento Aplicação Móvel Tinkerlink em Cronos Telecom
- ◆ Diretor Informático em Banco Santander
- ◆ Diretor Tecnologia Gestão de Serviço IT em Barclays Bank Espanha

**Sr. Alejandro Antonio Marcano Van Grieken**

- ◆ Gestor de Produto - Vikua, Remote (Jira, SCRUM, Figma, Slack, Notion)
- ◆ Programador Backend – InnovativeGX
- ◆ Licenciado em Engenharia de Sistemas, Universidade Metropolitana Caracas, Venezuela
- ◆ Mestrado em Cibersegurança, Universidade de León Online

**Sr. Emilio Arevalillo González**

- ◆ DBA Oracle BBVA
- ◆ *Assistant Project Manager Archibus Solution Center Spain*
- ◆ Backend developer na Telefónica I+D
- ◆ Licenciado em Engenharia Informática pela Universidade Politécnica de Madrid
- ◆ Mestrado em Software e Sistemas pela Universidade Politécnica de Madrid

**Sr. Héctor Arranz**

- ◆ Gestor de projetos software na Ezenit
- ◆ Licenciatura em Engenharia de Software pela Universidade Complutense de Madrid
- ◆ Mestrado MBA Power Leaders por The Power MBA Experiência Docente
- ◆ Formador de competências digitais na Three Life
- ◆ Formador de competências digitais na Fundação Esplai
- ◆ Professor adjunto para a licenciatura em desenvolvimento de aplicações multiplataforma no MEDAC
- ◆ Apoio aos trabalhos de empreendimento na Universidade Complutense de Madrid
- ◆ Formador de competências digitais na Three Life Fundação Esplai
- ◆ Consultor de empreendimento na Cink Venturing
- ◆ Professor adjunto da licenciatura em desenvolvimento de aplicações multiplataforma no MEDAC

# 05

## Estrutura e conteúdo

Este programa foi concebido por profissionais no ativo que compreendem a realidade atual e as tendências futuras no desenvolvimento de aplicações em sistemas operativos Android. São 10 módulos que ensinam na perfeição tudo o que precisa de saber para se tornar um criador de aplicações. A partir do módulo 1, a técnica e o conhecimento profundo do assunto evidencia-se e pode ser dominado até à perfeição em 12 meses. Obtendo uma qualificação 100% online graças à metodologia do *Relearning* que permitem uma rápida aprendizagem e memorização de conceitos e práticas.



“

*Conteúdo dinâmico com exercícios práticos, vídeos e formatos interativos determinará uma capacitação ágil e digerível”*

## Módulo 1. Linguagem de programação Android

- 1.1. Plataforma Android
  - 1.1.1. Plataforma Android
  - 1.1.2. Sistema Operativo Android
  - 1.1.3. *Open Handset Alliance* no desenvolvimento de Android
- 1.2. Arquitetura Android
  - 1.2.1. Elementos arquitetónicos de um sistema Android
  - 1.2.2. Comunicação entre elementos
  - 1.2.3. Extensibilidade da arquitetura Android
  - 1.2.4. Gestão de recursos máquina: bateria e memória
  - 1.2.5. Emuladores Android
- 1.3. Núcleo Linux de Android
  - 1.3.1. Composição do núcleo
  - 1.3.2. Elementos estruturais do núcleo
  - 1.3.3. A máquina Virtual Dalvik
  - 1.3.4. A máquina Virtual Android Run Time (ART)
- 1.4. Bibliotecas nativas de Android
  - 1.4.1. Bibliotecas nativas de Android
  - 1.4.2. Bibliotecas de compatibilidade (*Support Library*)
  - 1.4.3. Bibliotecas nativas e extensibilidade
- 1.5. O sistema de ficheiros e dados em Android
  - 1.5.1. Estrutura de uma aplicação Android típica
  - 1.5.2. Sistema de ficheiros YAFFS2 e ext4
  - 1.5.3. Utilização de SQLite e Room para a gestão de dados
- 1.6. Segurança em Android
  - 1.6.1. Sistemas de permissões
  - 1.6.2. Assinaturas digitais em Android *Application Package* (apk)
  - 1.6.3. Execução de processos no núcleo
  - 1.6.4. Linhas de execução e eventos
- 1.7. Componentes estruturais de uma aplicação padrão
  - 1.7.1. Vista (*View*)
  - 1.7.2. Atividade (*Activity*)
  - 1.7.3. Fragmento (*Fragment*)

- 1.7.4. Serviço (*Service*)
  - 1.7.5. Intenção (*Intent*)
  - 1.7.6. *Broadcasts Receiver* e *Content Provider*
  - 1.7.7. Gestão de dados e preferências de utilizador
- 1.8. Versões Android
  - 1.8.1. Versões Android
  - 1.8.2. Implementação de versões Android
  - 1.8.3. Dispersão de distribuições Android
  - 1.8.4. Android vs. Apple IOS e outros sistemas móveis
- 1.9. Android para veículos
  - 1.9.1. Android e o mundo do automóvel
  - 1.9.2. Elementos estruturais num sistema Android para automóveis
  - 1.9.3. Comunicação entre dispositivos
- 1.10. Android em Domótica, *Wearable* e na *Internet of Things* (IoT)
  - 1.10.1. O mundo conetado
  - 1.10.2. Elementos estruturais num sistema Android Domótico
  - 1.10.3. Elementos de Android *Wearable*
  - 1.10.4. Android em Internet of Things (IoT)

## Módulo 2. *Frameworks* utilizadas no desenvolvimento de aplicações Android

- 2.1. *Frameworks* no Desenvolvimento de Aplicações Android
  - 2.1.1. *Framework* em Desenvolvimento de Aplicações Android
  - 2.1.2. *Frameworks*. Tipologias
  - 2.1.3. Escolha do *Frameworks* para o projeto
- 2.2. Implementação de *Frameworks* em Android
  - 2.2.1. *Frameworks* Core de Android para Java/Kotlin
  - 2.2.2. *Jetpack Compose*
  - 2.2.3. *Frameworks* noutras linguagens
- 2.3. Sistemas de gestão de bibliotecas em desenvolvimento
  - 2.3.1. *Graddle*
  - 2.3.2. Automatização com *Graddle*
  - 2.3.3. Ferramenta do desenvolvimento Maven



- 2.4. Código limpo
  - 2.4.1. Código ordenado
  - 2.4.2. Preparação de códigos em Aplicações Android
  - 2.4.3. *Bikeshedding* e priorização
- 2.5. Padrões de desenvolvimento em Android
  - 2.5.1. Categorias de padrões
  - 2.5.2. Diferenças entre padrões
  - 2.5.3. *Factory*, *Observer* e *Singleton*
- 2.6. MVP Modelo, Vista e Apresentador
  - 2.6.1. MVC. Modelo, Vista e Controlador
  - 2.6.2. Modelo, Vista e Apresentador
  - 2.6.3. Exemplo prático: *Pokémon Battle*
- 2.7. MVVM. Modelo, Vista e View Model
  - 2.7.1. MVC vs. MVVM
  - 2.7.2. Modelo, Vista e *View Model*
  - 2.7.3. Exemplo Prático: *Pokémon Battle II*
- 2.8. *Frameworks* e bibliotecas mais utilizadas em Android
  - 2.8.1. Bibliotecas de interação com API
  - 2.8.2. Bibliotecas de conversão de dados
  - 2.8.3. *Firebase* e *Firebase Analytics*
- 2.9. *Framework* visual de Android
  - 2.9.1. Ciclo de vida de uma aplicação Android
  - 2.9.2. Design de vistas em XML
  - 2.9.3. Design de elementos e animações em XML
- 2.10. *Frameworks* de Android noutras linguagens
  - 2.10.1. *React Native*
  - 2.10.2. Flutter
  - 2.10.3. Ionic

### Módulo 3. Interfaces e *Layouts* no desenvolvimento de aplicações Android

- 3.1. Ciclo de vida de interface em Android
  - 3.1.1. Ciclo de vida em Android
  - 3.1.2. Relação processo-atividades
  - 3.1.3. Persistência do estado da aplicação
  - 3.1.4. Arquitetura *Clean* aplicada a Android
- 3.2. Vistas no Desenvolvimento de Aplicações Android (*Views*)
  - 3.2.1. Camada de apresentação da arquitetura *Clean*
  - 3.2.2. *Recycler View*
  - 3.2.3. *Adapter View*
- 3.3. Design no Desenvolvimento de Aplicações Android (*Layouts*)
  - 3.3.1. *Layouts* em Android
  - 3.3.2. *Constraint Layout*
  - 3.3.3. Criação de *Layouts* usando *Android Studio Layout Editor*
- 3.4. Animações no Desenvolvimento de Aplicações Android (*Animations*)
  - 3.4.1. Ícones e imagens
  - 3.4.2. Transições
  - 3.4.3. Diferença entre animação de propriedades e animação de vistas
- 3.5. Atividades e Intenções no Desenvolvimento de Aplicações Android (*Activity* e *Intentions*)
  - 3.5.1. Intenções explícitas e implícitas
  - 3.5.2. Barra de ações
  - 3.5.3. Comunicação entre atividades
- 3.6. Recursos alternativos e de sistema (*Material Design*, *Cardboard*, etc.)
  - 3.6.1. *Material Design* para Android
  - 3.6.2. Multimédia em Android
  - 3.6.3. Realidade virtual com *Google Cardboard for Android NDK*
- 3.7. Estilos e temas no Desenvolvimento de Aplicações Android
  - 3.7.1. Estilos num projeto Android
  - 3.7.2. Temas para o projeto Android
  - 3.7.3. Reutilização de estilos e temas
- 3.8. Gráficos, ecrã tátil e sensores
  - 3.8.1. Trabalho com gráficos avançados
  - 3.8.2. Gestão de dispositivos de ecrã tátil e com teclado
  - 3.8.3. Utilização de sensores de dispositivos Android

- 3.9. Design para a realidade aumentada
  - 3.9.1. Interfaces complexas usando a câmara
  - 3.9.2. Sensores de posição e GPS em realidade aumentada
  - 3.9.3. Apresentação em ecrãs não standard
  - 3.9.4. Erros e problemas comuns
- 3.10. Configuração avançada de interface com AndroidManifest.xml
  - 3.10.1. O poder do ficheiro de manifesto Android
  - 3.10.2. Design programático versus design declarativo
  - 3.10.3. Componentes chave do ficheiro

## Módulo 4. Linguagem de programação em aplicações Android. Kotlin

- 4.1. Linguagem de programação Kotlin
  - 4.1.1. Linguagem de programação Kotlin
  - 4.1.2. Linguagem de programação Kotlin. Caraterísticas
  - 4.1.3. Funcionamento de um programa em Kotlin
- 4.2. Programar em Kotlin
  - 4.2.1. Estrutura de um programa em Kotlin
  - 4.2.2. Palavras reservadas e sintaxe
  - 4.2.3. Escrever por consola y ler Inputs do utilizador – Hola Mundo
- 4.3. Tipos e variáveis em Kotlin
  - 4.3.1. Tipos e variáveis em Kotlin
  - 4.3.2. Declaração de variáveis Var vs. Val
  - 4.3.3. Operadores
  - 4.3.4. Conversão de tipos
  - 4.3.5. Arrays
- 4.4. Controlo de fluxo em Kotlin
  - 4.4.1. Controlo de fluxo
  - 4.4.2. Expressões condicionais
  - 4.4.3. Laços
- 4.5. Funções em Kotlin
  - 4.5.1. Funções em Kotlin
  - 4.5.2. Estrutura de uma função
  - 4.5.3. Funções de alcance (*Scope Functions*)





#### 4.6. Classes e objetos em Kotlin

- 4.6.1. Classes e objetos em Kotlin
- 4.6.2. Classes
- 4.6.3. Objetos
- 4.6.4. Construtores e Inicialização de propriedades
- 4.6.5. Classes aninhadas e classes internas
- 4.6.6. Classes de dados

#### 4.7. Herência em Kotlin

- 4.7.1. Herança
- 4.7.2. Superclasses e subclasses
- 4.7.3. Sobre-escrita de propriedades e funções
- 4.7.4. Herança vs. Outros tipos de relações entre classes
- 4.7.5. Classes seladas
- 4.7.6. Enumerados

#### 4.8. Classes abstratas e interfaces em Kotlin

- 4.8.1. Classes abstratas e interfaces
- 4.8.2. Classes abstratas
- 4.8.3. Interfaces
- 4.8.4. Validação e conversão de tipos - operadores Is, When, As

#### 4.9. Coleções em Kotlin

- 4.9.1. Coleções em Kotlin
- 4.9.2. Lista
- 4.9.3. Conjunto
- 4.9.4. Mapa

#### 4.10. Gestão de exceções e valores nulos em Kotlin

- 4.10.1. Gestão de exceções e valores nulos
- 4.10.2. Valor nulo, tipos *Nullable* y *Non-Nullable*
- 4.10.3. Exceções

## Módulo 5. Linguagem de programação em aplicações Android. Kotlin avançado genericidade, programação funcional e paralelismo

- 5.1. Genericidade em Kotlin
  - 5.1.1. Genericidade em Kotlin
  - 5.1.2. Genericidade em coleções, funções, classes e interfaces
  - 5.1.3. Covariância e contravariância: *Out ou In*
- 5.2. Funções *Lambda* em Kotlin
  - 5.2.1. Funções *Lambdas*
  - 5.2.2. Estrutura de uma função *Lambda*
  - 5.2.3. Uso de funções *Lambda*
- 5.3. Funções de ordem superior em Kotlin
  - 5.3.1. Funções de ordem superior
  - 5.3.2. Funções de ordem superior padrões de Kotlin
  - 5.3.3. Ligando chamadas a funções
- 5.4. Extensões em Kotlin
  - 5.4.1. Extensões em Kotlin
  - 5.4.2. Funções de extensão
  - 5.4.3. Propriedades de extensão
  - 5.4.4. Objetos acompanhantes
- 5.5. Padrão *Delegation* em Kotlin
  - 5.5.1. Padrão *Delegation*
  - 5.5.2. *Delegation* em Kotlin
  - 5.5.3. Propriedades delegadas
- 5.6. Anotações e reflexão em Kotlin
  - 5.6.1. Anotações e reflexão
  - 5.6.2. Anotações em Kotlin
  - 5.6.3. Reflexão em Kotlin
- 5.7. *Testing* em Kotlin
  - 5.7.1. *Testing* em Kotlin
  - 5.7.2. *Frameworks* e bibliotecas de *Testing* em Kotlin
  - 5.7.3. Kotest

- 5.8. Programação assíncrona em Kotlin
  - 5.8.1. Programação assíncrona
  - 5.8.2. Técnicas de Programação assíncrona em Kotlin
  - 5.8.3. Comparativas em técnicas de programação
- 5.9. Corrotinas em Kotlin
  - 5.9.1. Corrotinas
  - 5.9.2. Canais
  - 5.9.3. Contexto e *Dispatchers*
  - 5.9.4. Estado partilhado e concorrência
  - 5.9.5. Gestão de exceções em corrotinas
- 5.10. Ecossistema Kotlin
  - 5.10.1. Ecossistema Kotlin
  - 5.10.2. Bibliotecas para Kotlin
  - 5.10.3. Ferramentas para Kotlin

## Módulo 6. Gestão de dados em dispositivos Android

- 6.1. Gestão de dados. Tipologia
  - 6.1.1. A gestão de dados em Dispositivos móveis
  - 6.1.2. Alternativas para a gestão de dados em dispositivos Android
  - 6.1.3. Geração de dados para trabalho por inteligência artificial e análise de uso
  - 6.1.4. Ferramentas de medição do desempenho para uma ótima gestão de dados
- 6.2. Gestão de preferência de utilizador
  - 6.2.1. Tipos de dados envolvidos nos ficheiros de preferências
  - 6.2.2. Gestão das preferências de utilizador
  - 6.2.3. Exportação das preferências Gestão das licenças
- 6.3. Sistema de armazenamento de ficheiros
  - 6.3.1. Classificação de sistemas de ficheiros em dispositivos móveis
  - 6.3.2. Sistema interno de ficheiros
  - 6.3.3. Sistema externo de ficheiros
- 6.4. Ficheiros JSON como armazenamento em Android
  - 6.4.1. Informação não estruturada em ficheiros JSON
  - 6.4.2. Bibliotecas de gestão de dados com JSON
  - 6.4.3. Utilização de JSON em Android. Recomendações e otimizações

- 6.5. Ficheiros XML como armazenamento em Android
  - 6.5.1. Formato XML em Android
  - 6.5.2. XML através de bibliotecas SAX
  - 6.5.3. XML através de bibliotecas DOM
- 6.6. Bases de dados SQLite
  - 6.6.1. Base de dados relacional para gestão de dados
  - 6.6.2. Utilização da base de dados
  - 6.6.3. Métodos SQLite para a gestão de dados
- 6.7. Utilização avançada de bases de dados SQLite
  - 6.7.1. Recuperação de falhas utilizando transações em SQLite
  - 6.7.2. Utilização de cache para acelerar o acesso aos dados
  - 6.7.3. Bases de dados móvel
- 6.8. Biblioteca Room
  - 6.8.1. Arquitetura da biblioteca Room
  - 6.8.2. Biblioteca Room. Funcionalidade
  - 6.8.3. Biblioteca Room: vantagens e desvantagens
- 6.9. *Content Provider* para partilhar informação
  - 6.9.1. *Content Provider* para partilhar informação
  - 6.9.2. *Content Provider* em Android. Técnicas de utilização
  - 6.9.3. Segurança em *Content Provider*
- 6.10. Recolha de dados em Cloud da Internet
  - 6.10.1. Android e sistemas de armazenamento na Cloud
  - 6.10.2. Serviços SOAP e REST para Android
  - 6.10.3. Problemática dos sistemas distribuídos
  - 6.10.4. Internet como cópia de segurança dos dados da aplicação

## Módulo 7. Ferramentas de dispositivos Android

- 7.1. Gestão: Ferramentas “TO DO”
  - 7.1.1. Ferramentas de mercado
  - 7.1.2. Ferramentas de mercado. Comparação de funcionalidades
  - 7.1.3. Ferramentas de gestão. Diferenças
- 7.2. MDM: gestão empresarial de dispositivos
  - 7.2.1. Controlo dos dispositivos de empresa
  - 7.2.2. Análise das principais ferramentas de mercado
  - 7.2.3. Escolha da ferramenta
- 7.3. CRM: ferramentas de mercado
  - 7.3.1. Análise de ferramentas de mercado com aplicação Android
  - 7.3.2. Ferramentas de mercado. Eficiência
  - 7.3.3. Ferramentas de mercado. Usos
- 7.4. Drones com Android
  - 7.4.1. Apps de dispositivos Android para controlo de Drone
  - 7.4.2. Controlos autónomos
  - 7.4.3. Utilização de Drone em Android
- 7.5. Android, valor acrescentado em plataformas bancárias
  - 7.5.1. Android em plataformas bancárias
  - 7.5.2. Riscos e fraudes de ciberdelinquentes
  - 7.5.3. Utilizações de dispositivos móveis
- 7.6. *Segurança* em dispositivos móveis
  - 7.6.1. Ferramentas de mercado e a sua utilização
  - 7.6.2. Comparação das ferramentas
  - 7.6.3. Escolha de ferramenta para cada utilização
- 7.7. Ferramentas de entretenimento e formação
  - 7.7.1. Usos
  - 7.7.2. Ferramentas de mercado
  - 7.7.3. Comparações de funcionalidades entre ferramentas de desenvolvimento Android

## 7.8. IoT Android

7.8.1. *Framework* e plataformas de mercado

7.8.2. Riscos e considerações de IoT em Android

7.8.3. Utilização de IoT em Android

## 7.9. Eficiência de processos

7.9.1. Análise de ferramentas de mercado para a criação de Apps

7.9.2. Comparação de ferramentas de criação de aplicações para Android

7.9.3. *Use Case*

## 7.10. Aplicações mais descarregadas hoje em dia

7.10.1. Ferramentas mais descarregadas hoje em dia

7.10.2. Agrupação por famílias

7.10.3. Utilizações principais, secundárias e comparativas com IOS

## Módulo 8. Design *Responsive* em Android

### 8.1. *Responsive design*

8.1.1. Design *Responsive*

8.1.2. Usabilidade, acesibilidade e UX

8.1.3. Design *Responsive* Vantagens e desvantagens

### 8.2. *Mobile vs. Tablet vs. Web vs. Smartwatches*

8.2.1. Diferentes formatos, diferentes tamanhos, diferentes necessidades

8.2.2. Problemas no design

8.2.3. Adaptivo vs. Responsivo

### 8.3. Guias de estilo

8.3.1. Guias de estilo. Utilidade

8.3.2. *Material Design*

8.3.3. Guia de estilos própria

### 8.4. *Layouting* flexível

8.4.1. *Layouting* flexível

8.4.2. *Layouting* básico

8.4.3. *Layouting* em *Grid*

8.4.4. *Layouting* com *Relative Layout*

8.4.5. *Layouting* com *Constraint Layout*

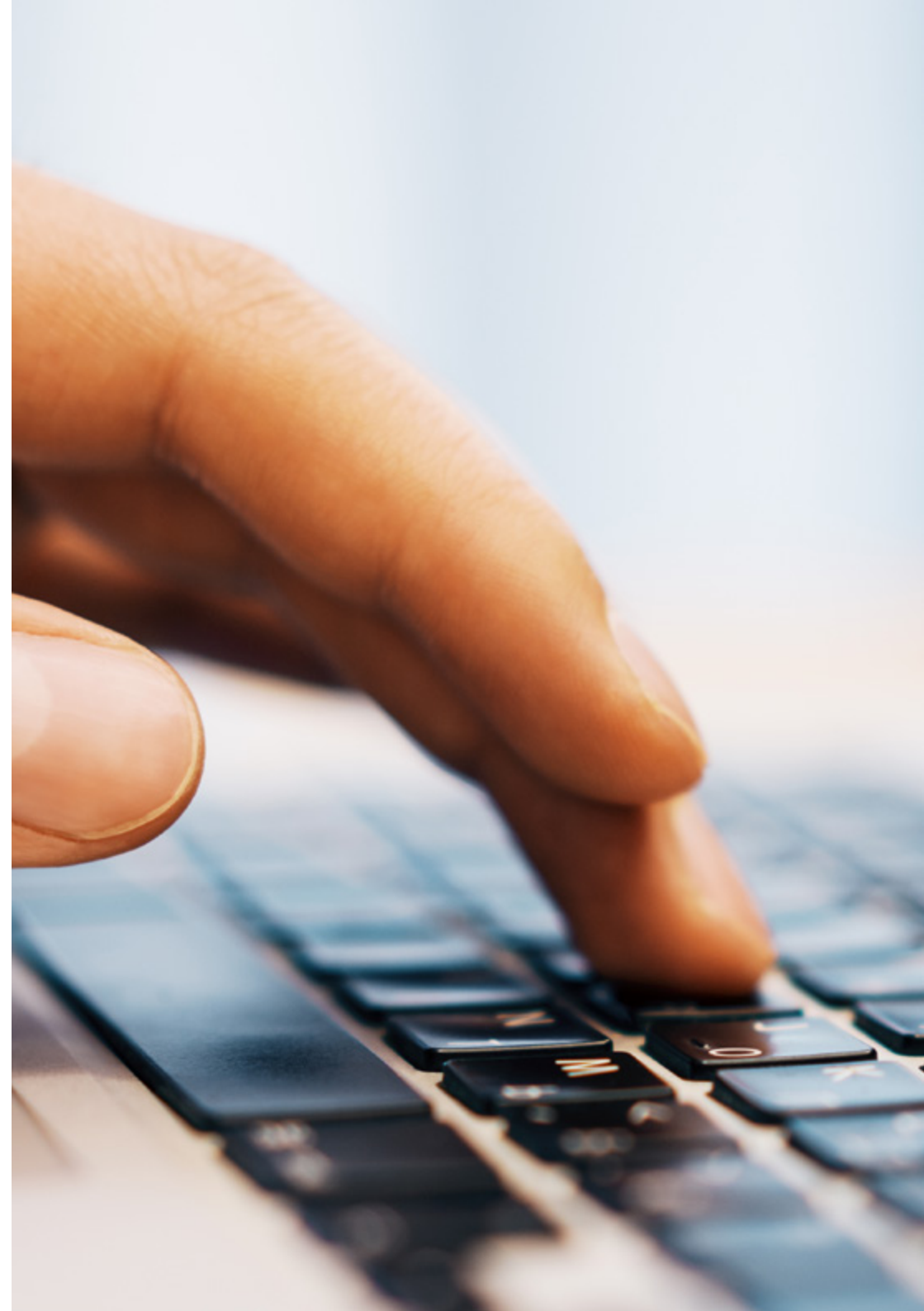




- 8.5. Recursos flexíveis
  - 8.5.1. Recursos flexíveis
  - 8.5.2. Imagens
  - 8.5.3. 9-Patch
  - 8.5.4. Recursos globais
- 8.6. Navegação flexível
  - 8.6.1. Navegação flexível
  - 8.6.2. Navegação com *Activities*
  - 8.6.3. Navegação com *Fragments*
- 8.7. Ferramentas externas
  - 8.7.1. Geradores automáticos
  - 8.7.2. Ferramentas de prototipagem
  - 8.7.3. Ferramentas de design
- 8.8. Debug e Tests
  - 8.8.1. *Debug Layouts*
  - 8.8.2. *Tests* automáticos
  - 8.8.3. Desenvolvimento baseado em componentes
  - 8.8.4. Boas práticas de *Testing e provas*
- 8.9. Alternativas a Android nativo I. *Web Pages*
  - 8.9.1. Design num *WebView*
  - 8.9.2. *Chrome Custom Tabs*
  - 8.9.3. *Debug e Tests* em *web Pages*
- 8.10. Alternativas a Android nativo II. Aplicações híbridas
  - 8.10.1. *React/React native*
  - 8.10.2. *Flutter*
  - 8.10.3. *Ionic*
  - 8.10.4. Apache Cordova

## Módulo 9. Marketing em aplicações Android

- 9.1. De *Customer Service* a *Customer Experience*
  - 9.1.1. *Customer Service*. Desenvolvimento do cliente atual
  - 9.1.2. Utilizador com acesso à informação. Exigências e necessidades
  - 9.1.3. A retroalimentação como fonte de conhecimento
- 9.2. *Customer Journey*
  - 9.2.1. Caminho do utilizador à conversão
  - 9.2.2. Micro-segmentação
  - 9.2.3. Experiência através dos canais
- 9.3. Medição da experiência do utilizador
  - 9.3.1. Arquitetura web e móvel
  - 9.3.2. Análise de sessão como novo padrão
  - 9.3.3. Estado da arte da experiência de utilizador
- 9.4. Marketing de Aplicações Android
  - 9.4.1. CX+IA
  - 9.4.2. CX+Blockchain
  - 9.4.3. CX+IoT
- 9.5. Produtos de CX (Experiência do Cliente)
  - 9.5.1. Padrões da indústria
  - 9.5.2. Telepresença
  - 9.5.3. Experiência do cliente para todos os agentes do desenvolvimento
- 9.6. Trabalho centrado no utilizador
  - 9.6.1. Equipamentos
  - 9.6.2. Pensamento de designer
  - 9.6.3. Trabalho de Campo
- 9.7. A ciência sobre o utilizador
  - 9.7.1. A ciência sobre o utilizador. Regras de ouro
  - 9.7.2. Iteração
  - 9.7.3. Erros comuns
- 9.8. Prototipagem e Wireframing
  - 9.8.1. Prototipagem e *Wireframing*
  - 9.8.2. *Hands-On*
  - 9.8.3. Nível avançado





- 9.9. Interfaces móveis
  - 9.9.1. Design visual. Regras
  - 9.9.2. Interface de apps. Chaves
  - 9.9.3. Boas práticas no Desenvolvimento de Interfaces Móveis
- 9.10. Boas práticas na Experiência de Utilizador. Conselhos para Programadores
  - 9.10.1. Nível Um. boas práticas em CX
  - 9.10.2. Nível Dois. boas práticas em UX
  - 9.10.3. Nível Três. boas práticas em UI

## Módulo 10 Ciclo de vida em aplicações Android. Cloud, Playstore e versioning

- 10.1. Ciclo de vida do software
  - 10.1.1. Ciclo de vida do software
  - 10.1.2. Metodologias ágeis
  - 10.1.3. O ciclo continuo ágil do software
- 10.2. Desenvolvimento do produto manual
  - 10.2.1. Integração manual
  - 10.2.2. Entrega manual
  - 10.2.3. Implementação manual
- 10.3. Integração supervisionada
  - 10.3.1. Integração contínua
  - 10.3.2. Revisão manual
  - 10.3.3. Revisões automáticas estáticas
- 10.4. Testes lógicos
  - 10.4.1. Provas unitárias
  - 10.4.2. Testes de integração
  - 10.4.3. Testes de comportamento
- 10.5. Integração contínua
  - 10.5.1. Ciclo de integração contínua
  - 10.5.2. Dependências entre integrações
  - 10.5.3. Integração contínua como metodologia de gestão do repositório
- 10.6. Entrega contínua
  - 10.6.1. Entrega contínua. Tipologia de problemas a resolver
  - 10.6.2. Entrega contínua. Resolução de problemas
  - 10.6.3. Vantagens da entrega contínua
- 10.7. Implantação contínua
  - 10.7.1. Implantação contínua. Tipologia de problemas a resolver
  - 10.7.2. Implantação contínua. Resolução de Problemas
- 10.8. *Firebase Test Lab*
  - 10.8.1. Configuração a partir de *GCloud*
  - 10.8.2. Configuração de Jenkins
  - 10.8.3. Utilização de Jenkins. Vantagens
- 10.9. Configuração de *Gradle*
  - 10.9.1. Sistema de automatização *Gradle*
  - 10.9.2. Componente de *Gradle Build Flavors*
  - 10.9.3. Componente de *Gradle Linteo*
- 10.10. Ciclo de Vida em Aplicações Android. Exemplos
  - 10.10.1. Configuração de *SemaphoreCI* e GitHub
  - 10.10.2. Configuração de blocos de trabalho
  - 10.10.3. Promoções e *Deployment*



*Não espere mais para dar o passo para a profissionalização, você faz parte do futuro. Comece a conceber as melhores aplicações móveis com este Mestrado Próprio”*

06

# Metodologia

Este programa de capacitação oferece uma forma diferente de aprendizagem.

A nossa metodologia é desenvolvida através de um modo de aprendizagem

cíclico: **o Relearning.**

Este sistema de ensino é utilizado, por exemplo, nas escolas médicas mais prestigiadas

do mundo e tem sido considerado um dos mais eficazes pelas principais publicações,

tais como a ***New England Journal of Medicine.***



“

*Descubra o Relearning, um sistema que abandona a aprendizagem linear convencional para o levar através de sistemas de ensino cíclicos: uma forma de aprendizagem que provou ser extremamente eficaz, especialmente em disciplinas que requerem memorização"*

## Estudo de Caso para contextualizar todo o conteúdo

O nosso programa oferece um método revolucionário de desenvolvimento de competências e conhecimentos. O nosso objetivo é reforçar as competências num contexto de mudança, competitivo e altamente exigente.

“

*Com a TECH pode experimentar uma forma de aprendizagem que abala as fundações das universidades tradicionais de todo o mundo”*



*Terá acesso a um sistema de aprendizagem baseado na repetição, com ensino natural e progressivo ao longo de todo o programa de estudos.*



## Um método de aprendizagem inovador e diferente

Este programa da TECH é um programa de ensino intensivo, criado de raiz, que propõe os desafios e decisões mais exigentes neste campo, tanto a nível nacional como internacional. Graças a esta metodologia, o crescimento pessoal e profissional é impulsionado, dando um passo decisivo para o sucesso. O método do caso, a técnica que constitui a base deste conteúdo, assegura que a realidade económica, social e profissional mais atual é seguida.

“

*O nosso programa prepara-o para enfrentar novos desafios em ambientes incertos e alcançar o sucesso na sua carreira”*

*O estudante aprenderá, através de atividades de colaboração e casos reais, a resolução de situações complexas em ambientes empresariais reais.*

O método do caso tem sido o sistema de aprendizagem mais amplamente utilizado nas principais escolas de informática do mundo desde que existem. Desenvolvido em 1912 para que os estudantes de direito não só aprendessem o direito com base no conteúdo teórico, o método do caso consistia em apresentar-lhes situações verdadeiramente complexas, a fim de tomarem decisões informadas e valorizarem juízos sobre a forma de as resolver. Em 1924 foi estabelecido como um método de ensino padrão em Harvard.

Numa dada situação, o que deve fazer um profissional? Esta é a questão que enfrentamos no método do caso, um método de aprendizagem orientado para a ação. Ao longo do programa, os estudantes serão confrontados com múltiplos casos da vida real. Terão de integrar todo o seu conhecimento, investigar, argumentar e defender as suas ideias e decisões.

## Relearning Methodology

A TECH combina eficazmente a metodologia do Estudo de Caso com um sistema de aprendizagem 100% online baseado na repetição, que combina elementos didáticos diferentes em cada lição.

Melhoramos o Estudo de Caso com o melhor método de ensino 100% online: o Relearning.

*Em 2019 obtivemos os melhores resultados de aprendizagem de todas as universidades online do mundo.*

Na TECH aprende- com uma metodologia de vanguarda concebida para formar os gestores do futuro. Este método, na vanguarda da pedagogia mundial, chama-se Relearning.

A nossa universidade é a única universidade de língua espanhola licenciada para utilizar este método de sucesso. Em 2019, conseguimos melhorar os níveis globais de satisfação dos nossos estudantes (qualidade de ensino, qualidade dos materiais, estrutura dos cursos, objetivos...) no que diz respeito aos indicadores da melhor universidade online do mundo.



No nosso programa, a aprendizagem não é um processo linear, mas acontece numa espiral (aprender, desaprender, esquecer e reaprender). Portanto, cada um destes elementos é combinado de forma concêntrica. Esta metodologia formou mais de 650.000 licenciados com sucesso sem precedentes em áreas tão diversas como a bioquímica, genética, cirurgia, direito internacional, capacidades de gestão, ciência do desporto, filosofia, direito, engenharia, jornalismo, história, mercados e instrumentos financeiros. Tudo isto num ambiente altamente exigente, com um corpo estudantil universitário com um elevado perfil socioeconómico e uma idade média de 43,5 anos.

*O Relearning permitir-lhe-á aprender com menos esforço e mais desempenho, envolvendo-o mais na sua capacitação, desenvolvendo um espírito crítico, defendendo argumentos e opiniões contrastantes: uma equação direta ao sucesso.*

A partir das últimas provas científicas no campo da neurociência, não só sabemos como organizar informação, ideias, imagens e memórias, mas sabemos que o lugar e o contexto em que aprendemos algo é fundamental para a nossa capacidade de o recordar e armazenar no hipocampo, para o reter na nossa memória a longo prazo.

Desta forma, e no que se chama Neurocognitive context-dependent e-learning, os diferentes elementos do nosso programa estão ligados ao contexto em que o participante desenvolve a sua prática profissional.



Este programa oferece o melhor material educativo, cuidadosamente preparado para profissionais:



#### Material de estudo

Todos os conteúdos didáticos são criados pelos especialistas que irão ensinar o curso, especificamente para o curso, para que o desenvolvimento didático seja realmente específico e concreto.

Estes conteúdos são depois aplicados ao formato audiovisual, para criar o método de trabalho online da TECH. Tudo isto, com as mais recentes técnicas que oferecem peças de alta-qualidade em cada um dos materiais que são colocados à disposição do aluno.



#### Masterclasses

Existem provas científicas sobre a utilidade da observação por terceiros especializada.

O denominado Learning from an Expert constrói conhecimento e memória, e gera confiança em futuras decisões difíceis.



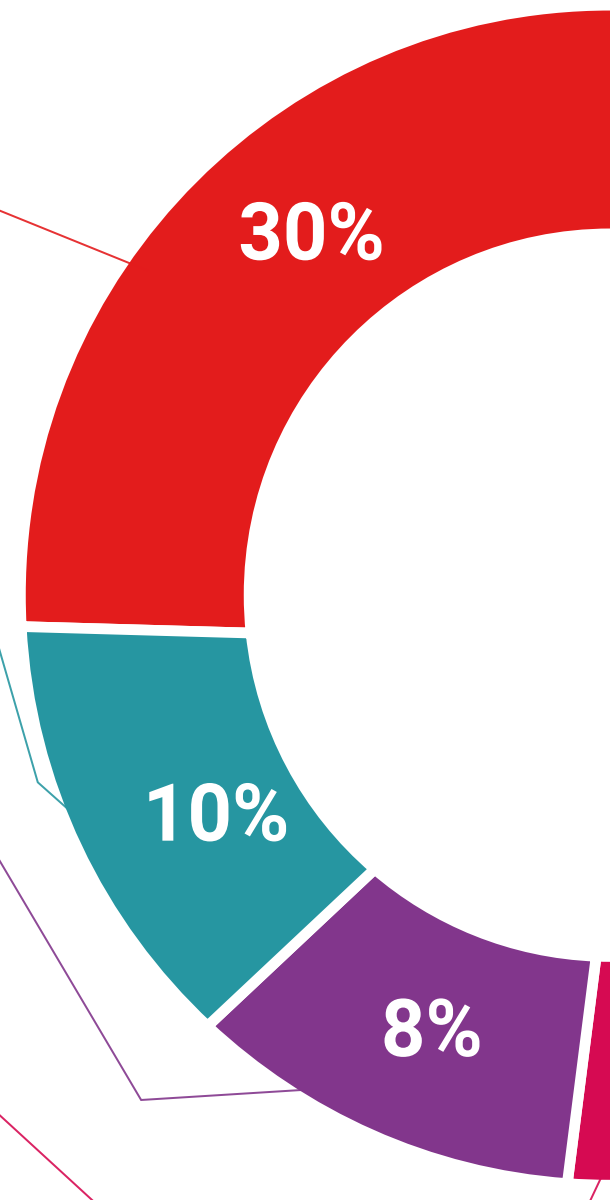
#### Práticas de aptidões e competências

Realizarão atividades para desenvolver competências e aptidões específicas em cada área temática. Práticas e dinâmicas para adquirir e desenvolver as competências e capacidades que um especialista necessita de desenvolver no quadro da globalização em que vivemos.



#### Leituras complementares

Artigos recentes, documentos de consenso e diretrizes internacionais, entre outros. Na biblioteca virtual da TECH o aluno terá acesso a tudo o que necessita para completar a sua capacitação







#### Case studies

Completarão uma seleção dos melhores estudos de casos escolhidos especificamente para esta situação. Casos apresentados, analisados e instruídos pelos melhores especialistas na cena internacional.



#### Resumos interativos

A equipa da TECH apresenta os conteúdos de uma forma atrativa e dinâmica em comprimidos multimédia que incluem áudios, vídeos, imagens, diagramas e mapas conceituais a fim de reforçar o conhecimento.

Este sistema educativo único para a apresentação de conteúdos multimédia foi premiado pela Microsoft como uma "História de Sucesso Europeu"



#### Testing & Retesting

Os conhecimentos do aluno são periodicamente avaliados e reavaliados ao longo de todo o programa, através de atividades e exercícios de avaliação e auto-avaliação, para que o aluno possa verificar como está a atingir os seus objetivos.



07

# Certificação

O Mestrado Próprio em Desenvolvimento de Aplicações Android garante, para além de um conteúdo mais rigoroso e atualizado, o acesso a um grau de Mestre emitido pela TECH Universidade Tecnológica.



“

*Conclua este plano de estudos com sucesso e receba o seu certificado sem sair de casa e sem burocracias”*

Este **Mestrado Próprio em Desenvolvimento de Aplicações Android** conta com o conteúdo educacional mais completo e atualizado do mercado.

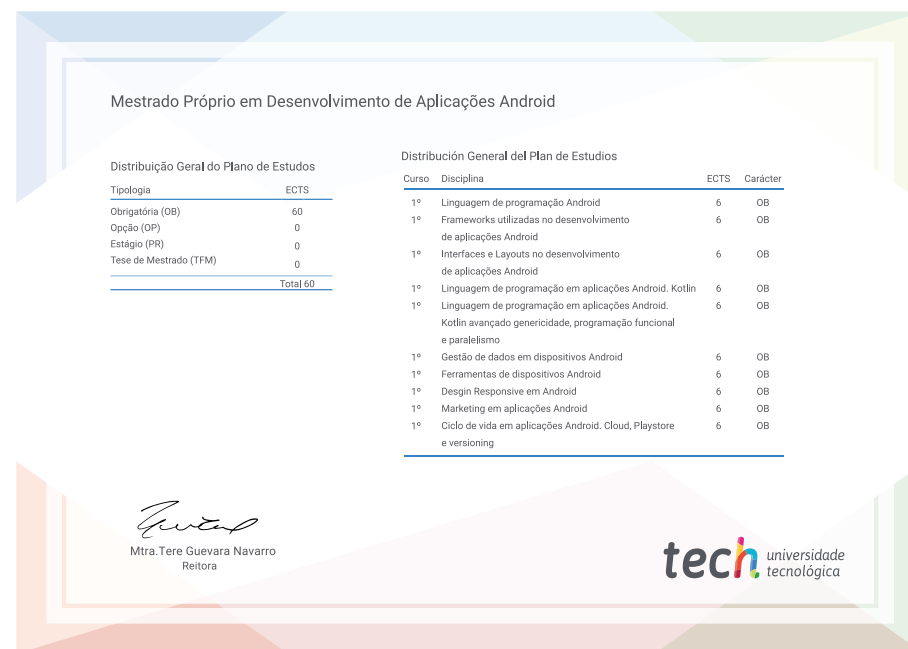
Uma vez aprovadas as avaliações, o aluno receberá por correio, com aviso de receção, o certificado\* correspondente ao título de **Mestrado Próprio** emitido pela **TECH Universidade Tecnológica**.

O certificado emitido pela **TECH Universidade Tecnológica** expressará a qualificação obtida no Mestrado Próprio, atendendo aos requisitos normalmente exigidos pelas bolsas de emprego, concursos públicos e avaliação de carreiras profissionais.

Título: **Mestrado Próprio em Desenvolvimento de Aplicações Android**

ECTS: **60**

Carga horária: **1.500 horas**



\*Apostila de Haia Caso o aluno solicite que o seu certificado seja apostilado, a TECH EDUCATION providenciará a obtenção do mesmo com um custo adicional.

futuro  
saúde confiança pessoas  
informação orientadores  
educação certificação ensino  
garantia aprendizagem  
instituições tecnologia  
comunidade compromisso  
atenção personalizada  
conhecimento inovação  
presente qualificação  
desenvolvimento si

**tech** universidade  
tecnológica

## Mestrado Próprio Desenvolvimento de Aplicações Android

- » Modalidade: online
- » Duração: 12 meses
- » Certificação: TECH Universidade Tecnológica
- » Créditos: 60 ECTS
- » Tempo Dedicado: 16 horas/semana
- » Horário: ao seu próprio ritmo
- » Exames: online

# Mestrado Próprio

## Desenvolvimento de Aplicações Android

```
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
//up/down  
if (e.keyCode == 38 || e.keyCode == 40) {  
    e.preventDefault();  
    e.stopImmediatePropagation();  
    if (e.keyCode == 38) { // up  
        // show previous search query  
        if (hist.currentIndex == hist.history.length) {  
            hist.currentQuery = input.value;  
            // skip previous search if we're already at the top  
            if (hist.currentQuery == hist.history[0].currentQuery) {  
                hist.currentIndex--;  
            }  
        }  
    }  
}
```