

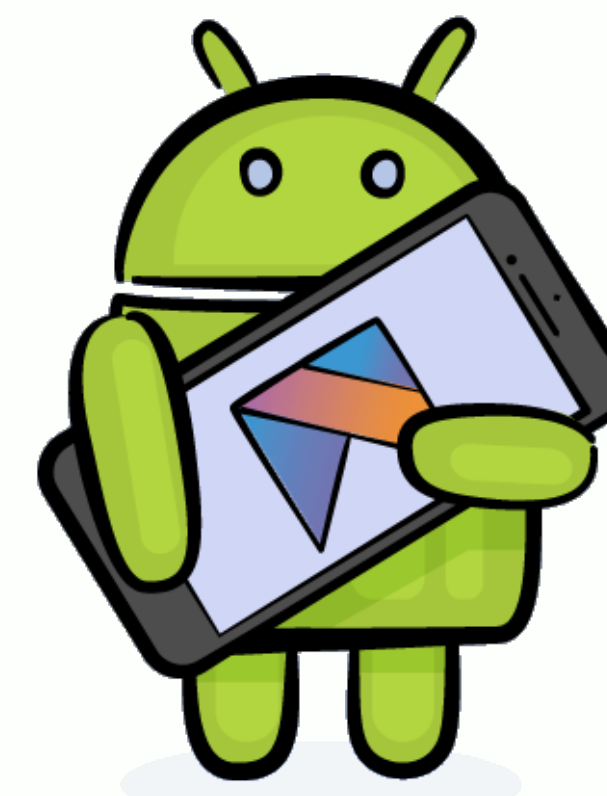


UNIVERSIDADE
FEDERAL DO CEARÁ
CAMPUS QUIXADÁ

Introdução ao desenvolvimento Android

QXD0276 - Desenvolvimento de Software para Dispositivos Móveis

Prof. Bruno Góis Mateus (brunomateus@ufc.br)



Agenda

- Introdução
- A evolução dos celulares
- O mercado de aplicativos
- Plataformas de desenvolvimento
- A evolução do Android
- Hello World

Introdução

Introdução

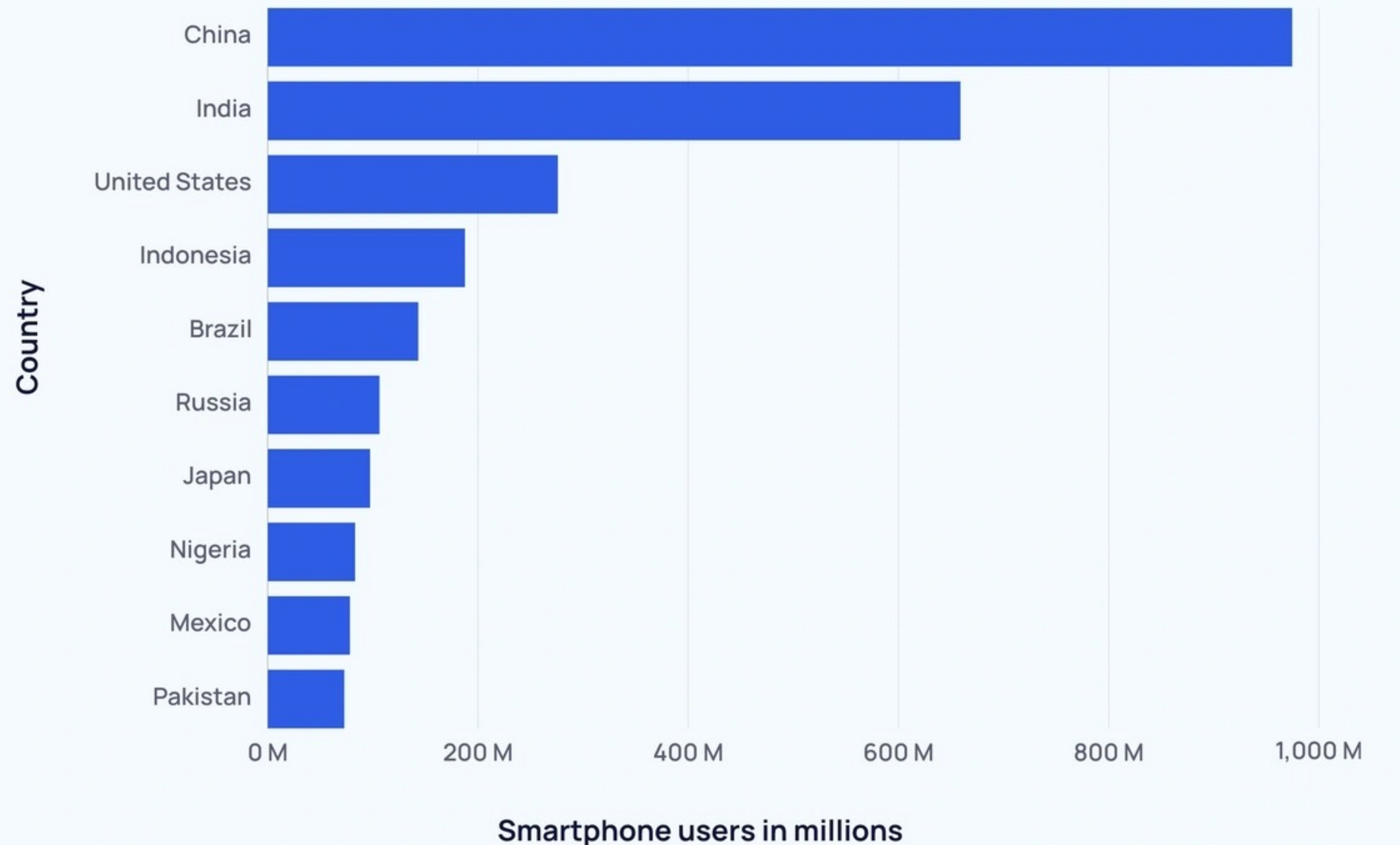
O uso de smartphones em números

- 90% do celulares ao redor do mundo são smartphones
- São mais de 7.2 bilhões de aparelhos
 - Esse número continua crescendo
 - Somos 8 bilhões de pessoas no mundo
- 94.2% dos dispositivos que acessam a Internet são smartphones



Fonte: <https://www.nextpit.com/best-products/best-camera-phones>

Active smartphone users by country



Introdução

O uso de smartphones em números

Rank	País	N de usuários	População total	Penetração
1	China	974.69 milhão	1.43 bilhão	68.4%
2	India	659 milhão	1.42 bilhão	46.5%
3	EUA	276.14 milhão	338.29 milhão	81.6%
4	Indonésia	187.7 milhão	275.5 milhão	68.1%
5	Brasil	143.43 milhão	215.31 milhão	66.6%
6	Rússia	106.44 milhão	144.71 milhão	73.6%
7	Japão	97.44 milhão	123.95 milhão	78.6%
8	Nigéria	83.34 milhão	218.54 milhão	38.1%
9	México	78.37 milhão	127.5 milhão	61.5%
10	Paquistão	72.99 milhão	235.82 milhão	31%

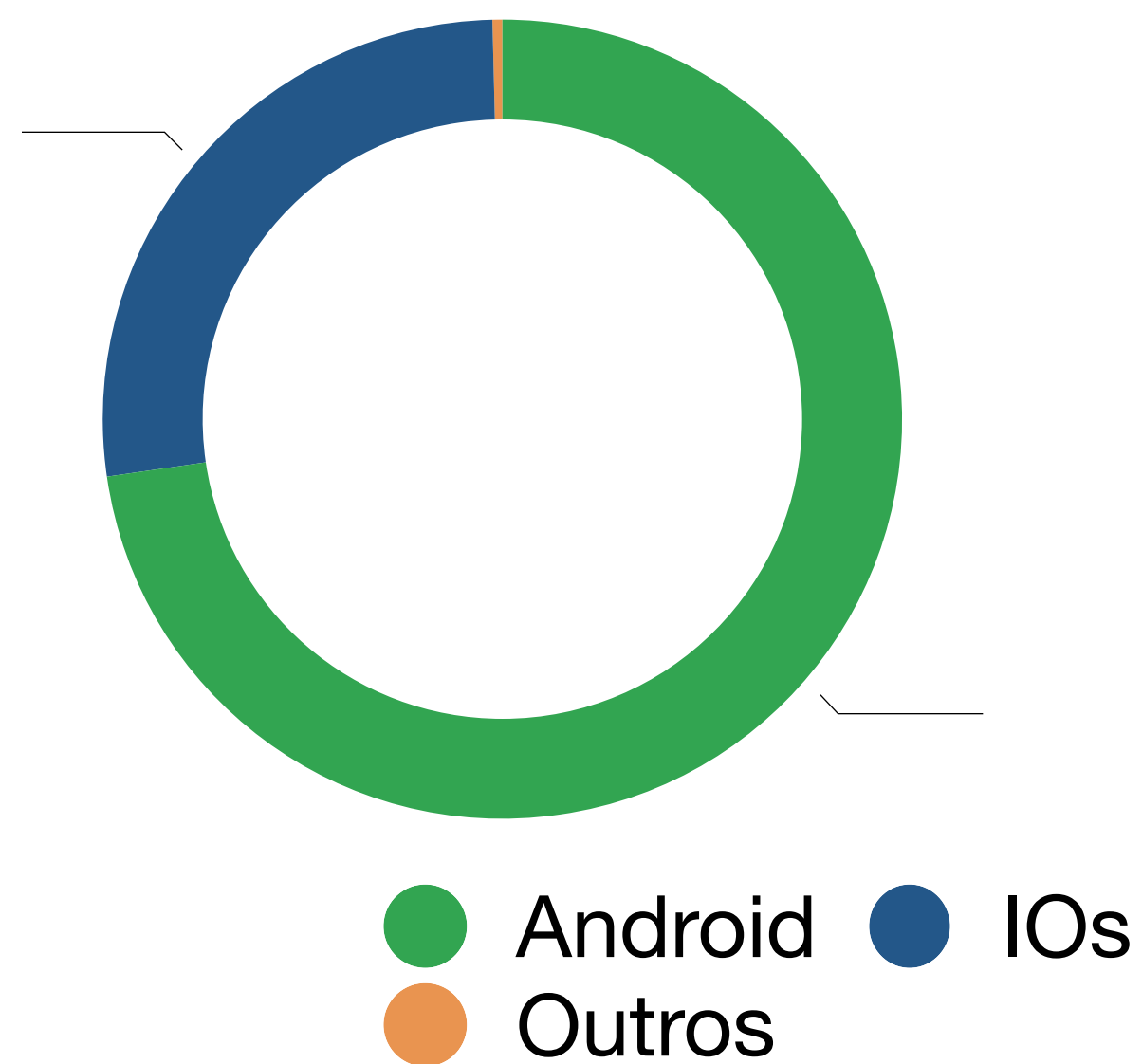
**Muitas pessoas não possuem
smartphone. Aproximadamente
1.1 bilhões de pessoas sequer
tem acesso a energia elétrica.**

Fonte: <https://explodingtopics.com/blog/smartphone-stats>

Introdução

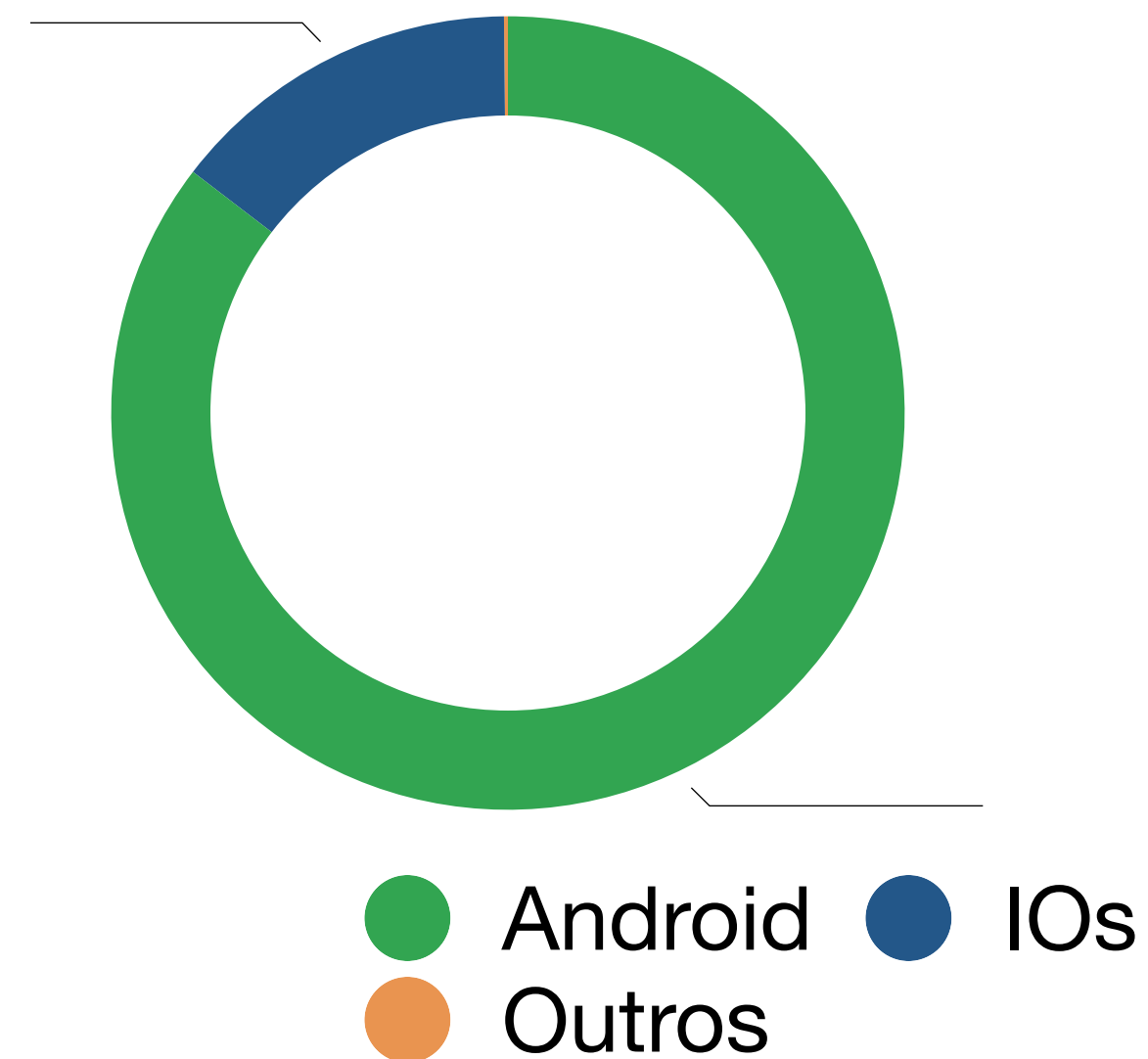
A balha entre sistemas operacionais

Mercado global de sistemas operacionais



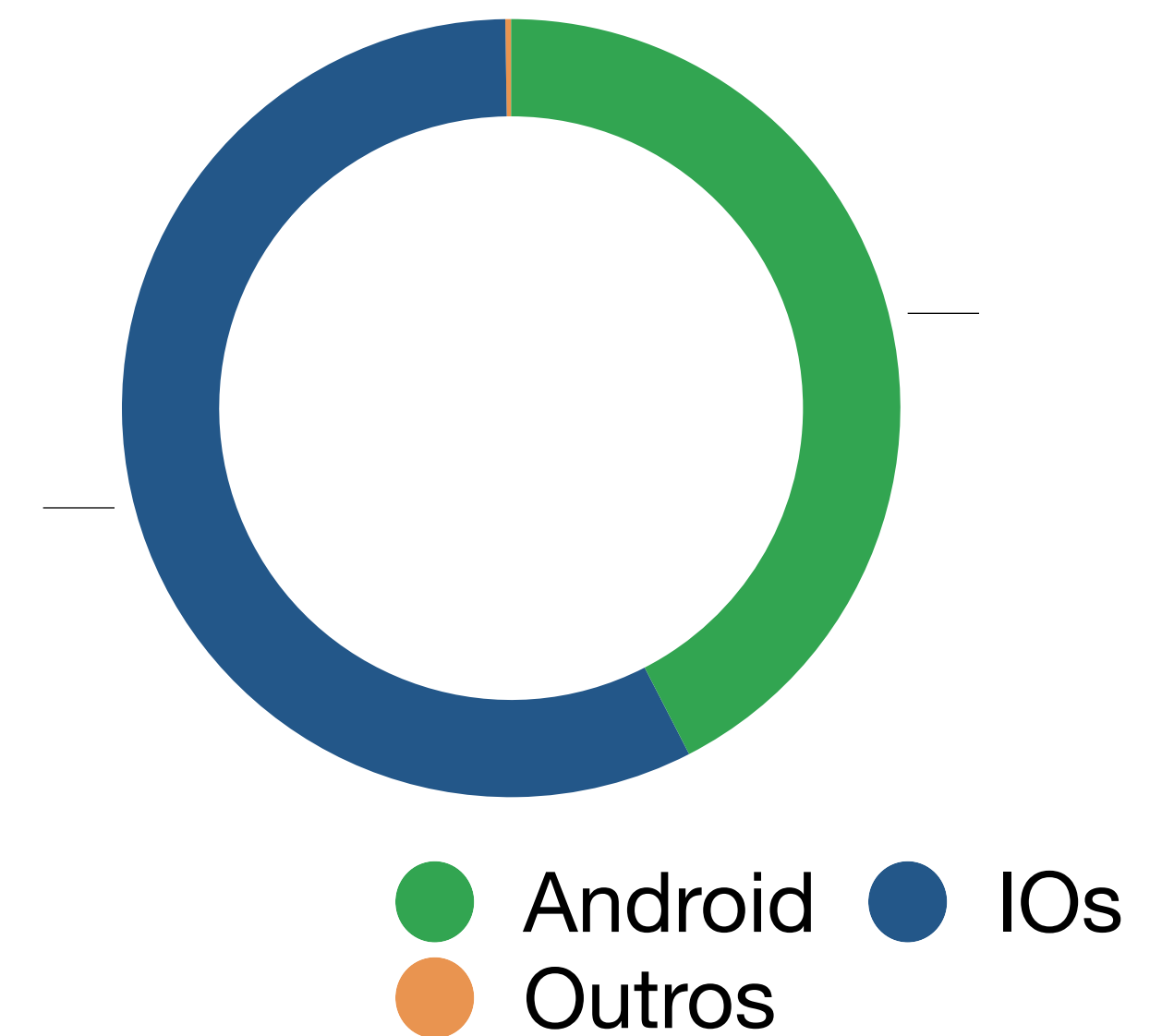
Fonte: [StatsCounter](#)

Mercado global de sistemas operacionais no Brasil



Fonte: [StatsCounter](#)

Mercado global de sistemas operacionais nos EUA



Fonte: [StatsCounter](#)

Introdução

A balha entre sistemas operacionais



Fonte: <https://rodrigoesilva.wordpress.com/2013/01/27/r-i-p-enfim-o-symbian-morre/>



Fonte: <https://x.com/SergioCabrales/status/382946575988711424?lang=ar-x-fm>

Smartphone OS sales market share evolution

A evolução dos celulares

A evolução dos celulares

De um tijolo bege com antena (1973) a um supercomputador de bolso (2020s)

- Primeira ligação móvel (1973)
 - Martin Cooper (Motorola) realizou a primeira chamada de um telefone portátil, ligando para Joel Engel (Bell Labs).
 - O aparelho era grande, pesado, sem tela e com antena
- Anos 1980 – Primeiros consumidores
 - Telefones ainda eram caros e volumosos, serviam apenas para chamadas.
 - Exemplo: um Motorola custava cerca de £300 (US\$ 500).



A evolução dos celulares

De um tijolo bege com antena (1973) a um supercomputador de bolso (2020s)

- Anos 1990–2000 – Popularização e funções extras
 - Tornaram-se menores e mais acessíveis.
 - Ganharam funções como SMS, rádio, câmera e internet lenta.
 - Nokia N95 (2006) foi um marco antes da chegada do iPhone.



Fonte: <https://fonewallah.com/products/nokia-n95>

A evolução dos celulares

De um tijolo bege com antena (1973) a um supercomputador de bolso (2020s)

- 2007 em diante – Smartphones
 - O iPhone revolucionou o mercado ao integrar várias funções em um só dispositivo.
 - Celulares deixaram de ser apenas telefones e passaram a ser mini-computadores portáteis.
 - Avanços tecnológicos
 - Chips menores e mais poderosos permitiram adicionar sensores e novas funções:
 - GPS, câmeras avançadas, acelerômetros, giroscópios, barômetros, sensores de impressão digital e até LiDAR.
 - Conexões móveis evoluíram (3G → 4G → 5G, já se discute o 6G).



Fonte: <https://www.wired.com/story/iphone-history-dogfight/>

The Evolution of Smartphones



1984

Motorola
DynaTAC 8000X



1992

The First
Smartphone
IBM Simon



1997

Nokia 6110



2000

Sharp J-SH04



2007

The First
Apple iPhone



2008

The First
Android
HTC Dream



2022+

Modern
Smartphones

A evolução dos celulares

De um tijolo bege com antena (1973) a um supercomputador de bolso (2020s)

- 1973 – Primeira ligação de celular
➡ Martin Cooper (Motorola) faz a 1ª chamada móvel.
- 1984 – Primeiro celular portátil
➡ Motorola DynaTAC 8000X (“The Brick”), caro, pesado, bateria curta.



Fonte: <https://www.techradar.com>

A evolução dos celulares

De um tijolo bege com antena (1973) a um supercomputador de bolso (2020s)

- 1992 – Primeiro smartphone (IBM Simon)
 - ➡ Tela touch com stylus, agenda, e-mail.
- 1994 – Primeiros apps móveis
 - ➡ Agenda, calendário, notas no IBM Simon.
- 1997 – Início dos jogos móveis
 - ➡ Nokia 6110 com Snake.
 - ➡ Ericsson cria o termo smartphone.



Fonte: <https://www.oxygen.ie/simon-first-smartphone-ever/>



Fonte: <https://www.pocketgamer.biz/finnish-mobile-game-industry-part-1/>

A evolução dos celulares

De um tijolo bege com antena (1973) a um supercomputador de bolso (2020s)

- 2000 – Primeiro celular com câmera
➡ Sharp J-SH04 (Japão).
- 2001 – Internet nos celulares
➡ Conexão 3G → e-mails e videoconferências.
- 2007 – Revolução com o iPhone
➡ Touchscreen, iPod, câmera, internet completa.
➡ App Store → milhões de aplicativos.



Fonte: <https://en.wikipedia.org/wiki/J-SH04>

A evolução dos celulares

De um tijolo bege com antena (1973) a um supercomputador de bolso (2020s)

- 2008 – Primeiro Android
➡ HTC Dream (T-Mobile G1) → estreia do Android Market.
- 2020 – Smartphones modernos
➡ Câmeras profissionais, redes sociais, streaming, apps em massa.
- 2023+ – O futuro
➡ Dobráveis, IA, 5G/6G, tradução em tempo real, aparelhos sem botões.



Fonte: https://pt.wikipedia.org/wiki/HTC_Dream



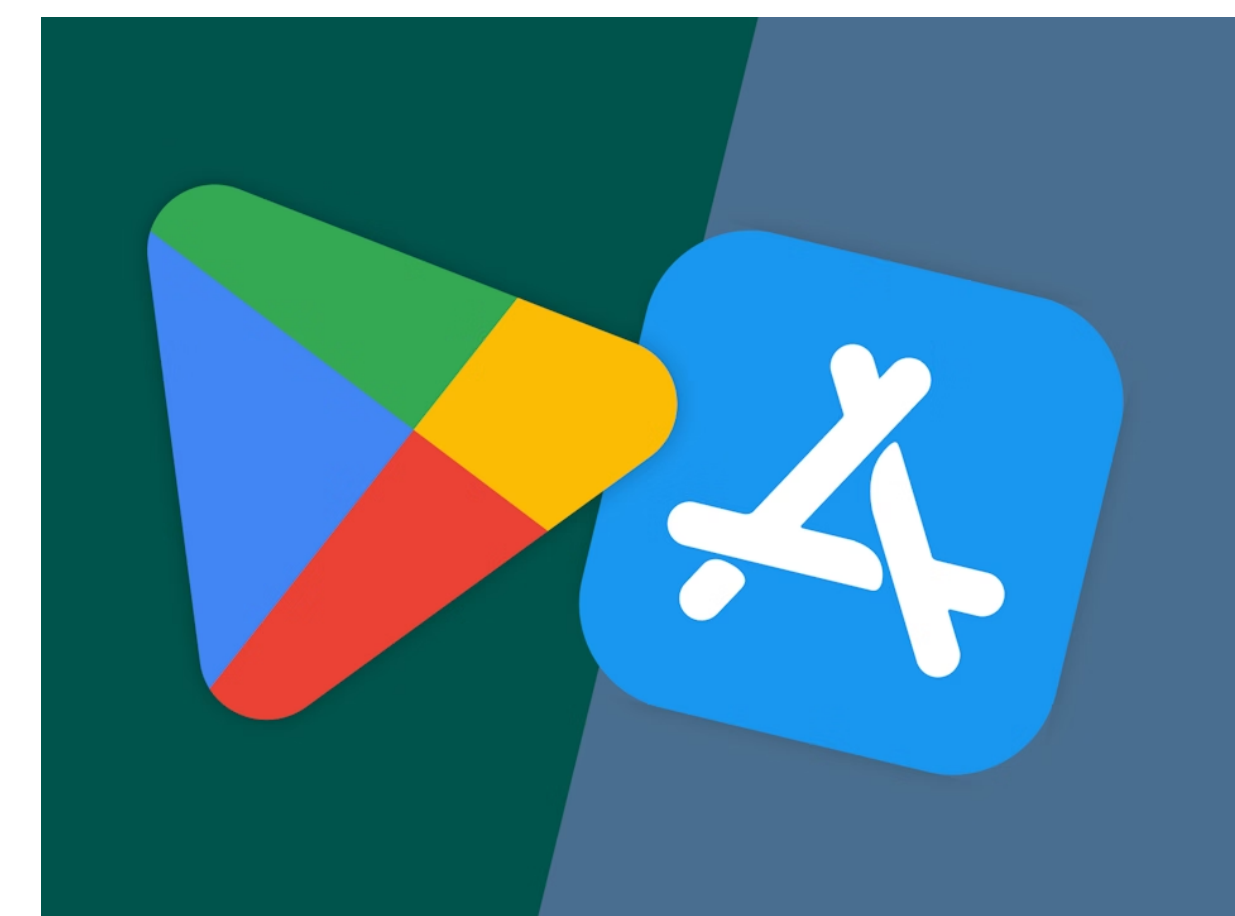
Fonte: <https://technologysage.net/foldable-phones-innovation-or-gimmick/>

O mercado de aplicativos

O mercado de aplicativos

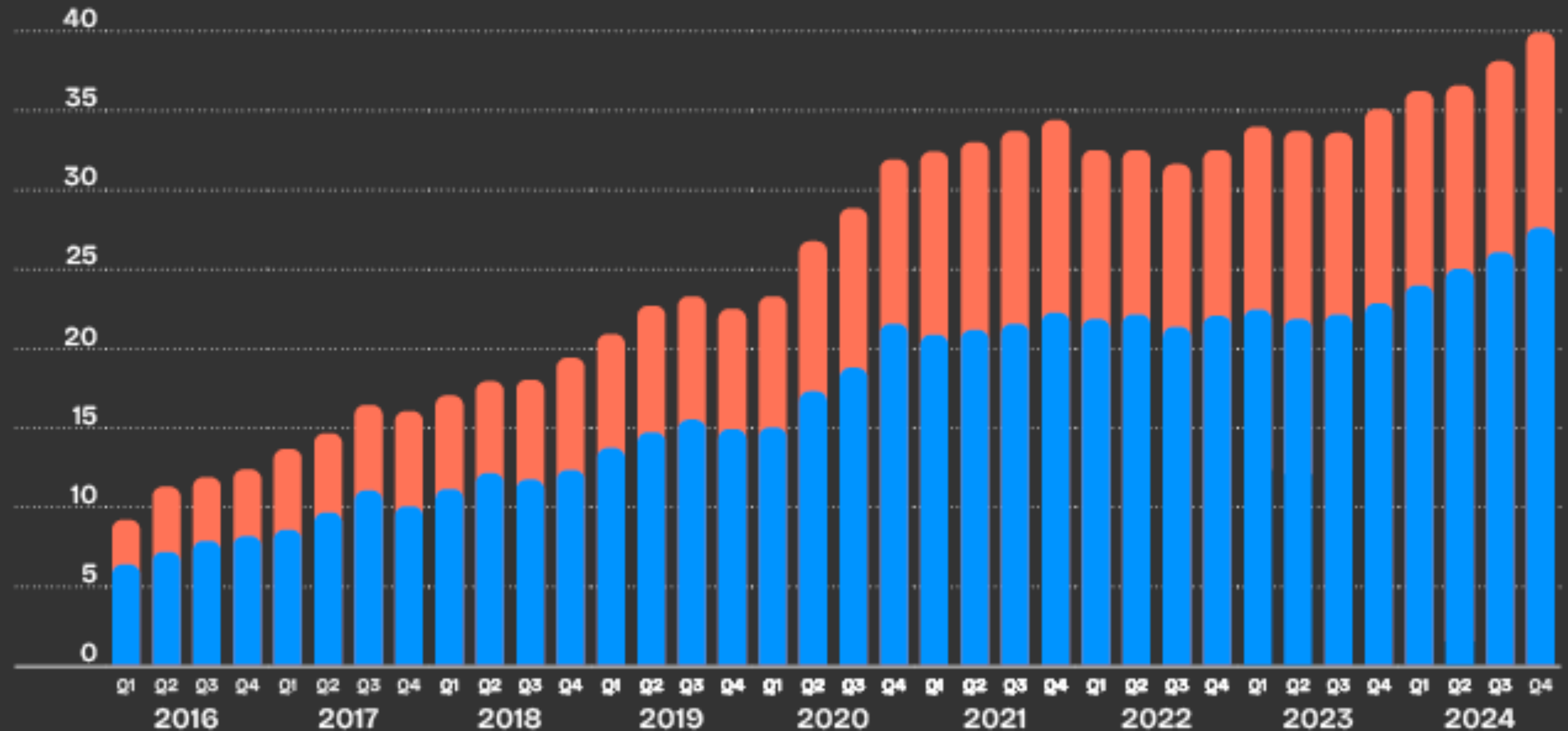
Panorama geral

- 🌐 Downloads (Q1 2024): quase 34 bilhões somados (Play + App Store)
 - Mercado bilionário, com crescimento contínuo
- Google Play Store (Android)
 - 1,68 milhão de apps disponíveis (Jun/2024)
 - Maioria gratuita, mas bilhões em receita via jogos e compras internas
- Apple App Store (iOS)
 - Menor em número de apps, mas maior em receita
 - US\$ 25 bilhões gastos por usuários no 2º trimestre de 2024



Fonte: <https://www.apptweak.com/>

Apple App Store and Google Play Revenues 2016 to 2024 (\$bn)



O mercado de aplicativos

Comparativo Android x iOS

- Outros App Stores
 - Amazon Appstore, Samsung Galaxy Store → Android apps
 - Aptoide (portuguesa) → distribui apps Android
 - Epic Games Store → Fortnite, Fall Guys, Rocket League



Fonte: <https://www.samsung.com/>

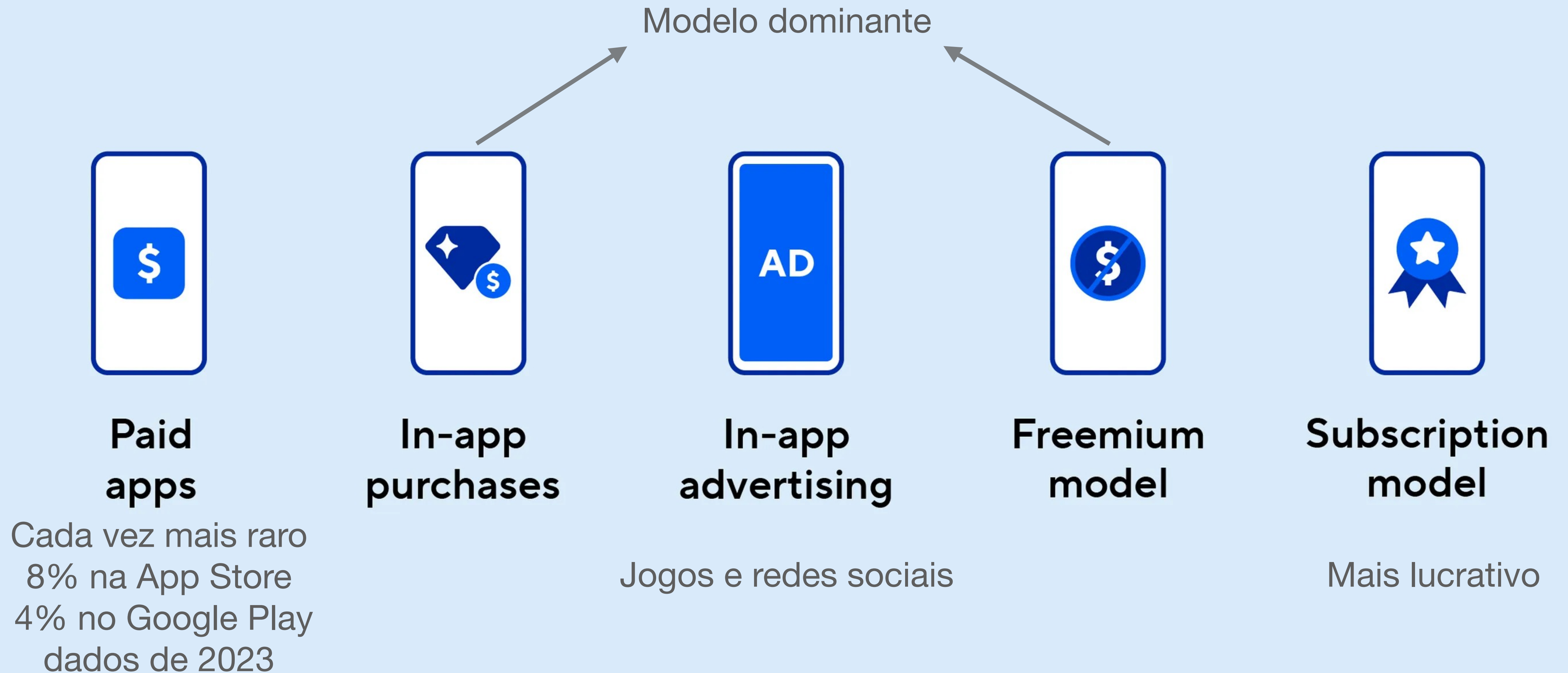


Fonte: <https://aptoide.br.aptoide.com>



Fonte: <https://store.epicgames.com>

Estratégias de monetização



O mercado de aplicativos

Distribuição

- Por padrão, app só pode ser instalado via loja de aplicativos
 - No iOS, única alternativa é fazer jailbreak do aparelho
- Loja **analisa o aplicativo antes de publicar**, o que pode levar dias ou semanas
- Desenvolvedores pagam assinatura para publicar nas lojas
 - **US\$ 100/ano** (Apple App Store), **US\$ 25 total** (Google Play)
 - No iOS, **Apple detém exclusividade na venda de apps** (App Store)

Plataformas de desenvolvimento

Plataformas de desenvolvimento

- Do ponto de vista do usuário, há duas opções:
 - Sistemas web (web app)
 - São executados no navegador
 - Não são distribuídos nas lojas de aplicativos
 - App (nativo): roda no sistema operacional

Plataformas de desenvolvimento

Estratégias de desenvolvimento

- O desenvolvedor pode criar uma aplicação de diversas formas:
 - Desenvolvimento nativo
 - Desenvolvimento web
 - Desenvolvimento híbrido
 - Desenvolvimento nativo multiplataforma

Plataformas de desenvolvimento

Desenvolvimento nativo

- Abordagem em que um aplicativo específico para uma plataforma é criado
 - Usam um **SDK (Software Development Kit)** oficial da plataforma no desenvolvimento
- Android:
 - **Java** ou **Kotlin**, Android SDK (~~View API~~ ou com **Jetpack Compose**)
- iOS:
 - ~~Objective C~~ ou **Swift**, iOS SDK, usando Cocoa Touch ou SwiftUI.

Plataformas de desenvolvimento

Desenvolvimento nativo

- Características:

 Alto desempenho: uso otimizado do hardware e software da plataforma

 Melhor experiência de usuário – segue as diretrizes oficiais de design

 Acesso completo aos recursos do dispositivo – câmera, GPS, sensores etc.

 Funcionalidade offline – pode rodar sem internet

 Visibilidade nas App Stores – fácil publicação e promoção na lojas de aplicativos

 SDK é atualizado simultaneamente com lançamento de nova versão do SO

Plataformas de desenvolvimento

App nativo

✓ Vantagens:

- Melhor desempenho e uso total dos recursos do dispositivo
- Experiência de usuário otimizada (UI/UX nativo)
- Maior suporte da própria plataforma

✗ Desvantagens:

- Maior custo (dois códigos diferentes: Android e iOS)
- Curva de aprendizado específica para cada plataforma

Plataformas de desenvolvimento

Desenvolvimento web

- Originalmente era muito limitado (não acessava câmera, não funciona offline...)
- **Progressive web apps** utilizam tecnologias que nos permite uma experiência próxima da oferecidas por aplicações nativas
- Por se tratar de um app da Web, melhor distribuição, qualquer dispositivo tendo uma única base de código
- Depois de instalado, um PWA se parece com qualquer outro app:
 - Possui um ícone na tela inicial
 - Ele aparece na pesquisa de apps no dispositivo, no acesso rápido, etc ...
 - É aberta em uma janela independente, separada da interface do navegador



Fonte: [Wikipedia](#)

Plataformas de desenvolvimento

PWA

Recurso	Website	Progressive Web App	Nativo
Usar quando estiver offline	Não	Sim	Sim
Precisa baixar em apps store	Não	Não	Sim
Implementa Design Responsivo	Sim	Sim	Sim
Indexável por mecanismos de busca	Sim	Sim	Sim
Notificações locais	Não	Sim	Sim
Notificações Push	Não	Sim	Sim
Precisa baixar para instalar	Não	Não	Sim
Atualizações rápidas e on-demand	Sim	Sim	Não

Plataformas de desenvolvimento

PWA

✓ Vantagens:

- Mais barato e rápido de desenvolver (base única de código)
- Não depende de loja de aplicativos para distribuição
- Funciona em qualquer dispositivo*

✗ Desvantagens:

- Recursos limitados de hardware (GPS, câmera, sensores, etc.)
- Menor engajamento (nem sempre aparece nas lojas oficiais)
- Pode ter performance inferior em comparação aos nativos

Plataformas de desenvolvimento

App híbrido

- App construído com tecnologias web, rodando dentro de um contêiner nativo
- ➡ Acesso a recursos do dispositivo via plugins
 - GPS, câmera, Bluetooth, biometria etc.
- ♻ Um único código pode rodar em iOS e Android



Plataformas de desenvolvimento

App híbrido

✓ Vantagens:

- Uma única base de código
- Reutilização total de código web
- Acesso a quase todos os recursos via plugins

✗ Desvantagens:

- Performance fraca para apps complexos (usa WebView)
- UI menos fluida em relação ao nativo ou cross moderno

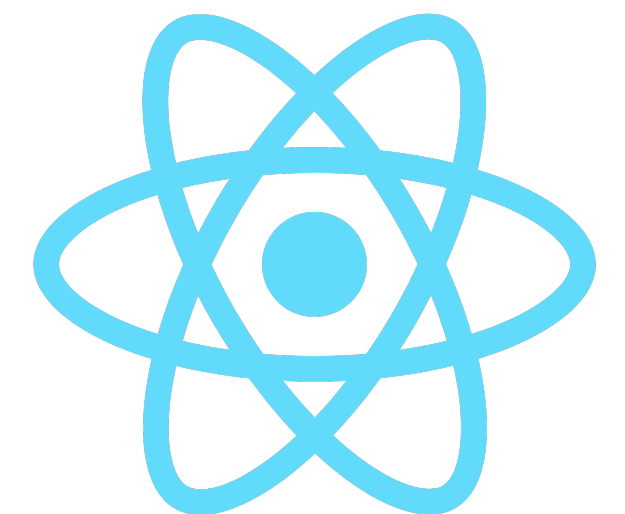
Plataformas de desenvolvimento

Desenvolvimento nativo multiplataforma

- Permite a criação de aplicativos para múltiplas plataformas usando um único código-base
- Principais frameworks cross-platform
 - **Flutter**, usa Dart
 - **React Native**, usa JavaScript
 - **Kotlin Multiplatform**, usa Kotlin

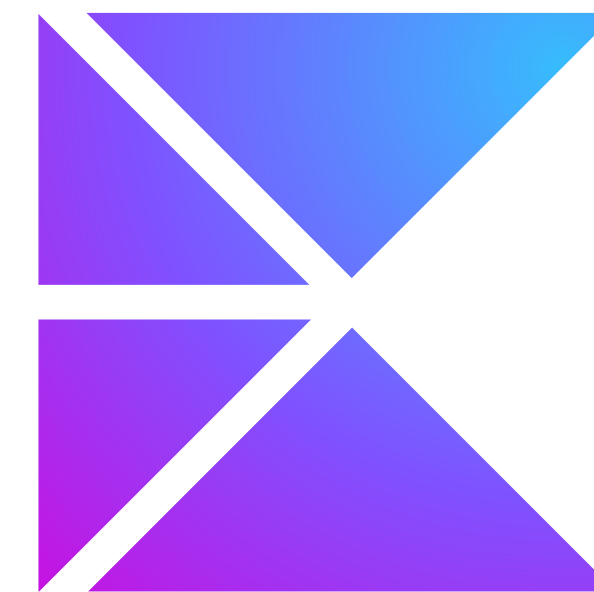


Fonte: <https://flutter.dev/brand>



React Native

Fonte: <https://worldvectorlogo.com>



Fonte: <https://www.jetbrains.com>

Plataformas de desenvolvimento

Flutter

- Criado pelo Google (2017)
- Usa a linguagem Dart
- Hot Reload
- Suporte nativo ao Material Design
- Renderização própria (não depende de WebViews)
- Suporte a mobile, web e desktop a partir de um único código
- Comunidade ativa e em crescimento constante



Dart

Plataformas de desenvolvimento

Flutter

✓ Vantagens

- Desenvolvimento rápido com Hot Reload
- UI rica e personalizável com widgets prontos
- Desempenho próximo ao nativo
- Grande adoção no mercado e suporte da Google

✗ Desvantagens

- Código em Dart (menos popular que JavaScript, Kotlin, etc.).
- Tamanho dos apps maior em comparação a nativos.
- Algumas limitações em bibliotecas específicas para recursos nativos.
- Menor maturidade em desktop e web comparado ao mobile.

Plataformas de desenvolvimento

React Native

- Criado pelo Facebook (Meta) em 2015
- Baseado na biblioteca React (JavaScript)
- Usa componentes que são renderizados como nativos
- Fast Refresh: permite visualizar alterações em tempo real
- Suporte a Android, iOS e integração com web via frameworks adicionais
- Grande comunidade global e vasto ecossistema de pacotes

Plataformas de desenvolvimento

Flutter

✓ Vantagens

- Baseado em JavaScript, linguagem popular e acessível
- Grande comunidade e muitos pacotes prontos
- UI próxima ao nativo e altamente customizável
- Usado por grandes empresas (Facebook, Microsoft, Instagram, etc.)

✗ Desvantagens

- Atualizações do framework podem quebrar compatibilidade.

Plataformas de desenvolvimento

Kotlin Multiplatform

- Criado pela JetBrains.
- Permite **compartilhar código** entre Android, iOS, Web, Desktop e Backend
- Mantém a possibilidade de escrever código nativo quando necessário
- Usa a linguagem Kotlin (popular no ecossistema Android)
- Suporte ao Compose Multiplatform para criação de UIs declarativas
- Integração flexível em projetos já existentes
- Comunidade em crescimento, promovida a Stable em 2023

Plataformas de desenvolvimento

Kotlin Multiplatform

✓ Vantagens

- Reaproveitamento de código entre múltiplas plataformas
- Integração nativa: mantém acesso a APIs específicas de cada sistema
- Ideal para equipes que já usam Kotlin no Android
- Flexibilidade: compartilhar apenas o que for conveniente (não exige 100%)
- Suporte oficial da JetBrains

✗ Desvantagens

- Mais jovem que Flutter e React Native (menor maturidade).
- Menor número de bibliotecas prontas comparado a outras opções.
- Requer conhecimento de Kotlin
- Curva de aprendizado maior para equipes sem experiência em Android/Kotlin

Plataformas de desenvolvimento

Desenvolvimento nativo multiplataforma

✓ Vantagens:

- Código único para várias plataformas
- Reduz tempo e custo de desenvolvimento
- Grande comunidade e bibliotecas prontas*

✗ Desvantagens:

- Desempenho inferior ao nativo em alguns casos
- Dependência de atualizações do framework
- Dependência de bibliotecas de terceiros para acesso a recursos nativos
- Integração com recursos avançados de hardware pode exigir código nativo adicional

Plataformas de desenvolvimento

- Quem é seu público alvo?
- Quanto pode ser gasto?
- Quais os objetivos do projeto?



A evolução do Android

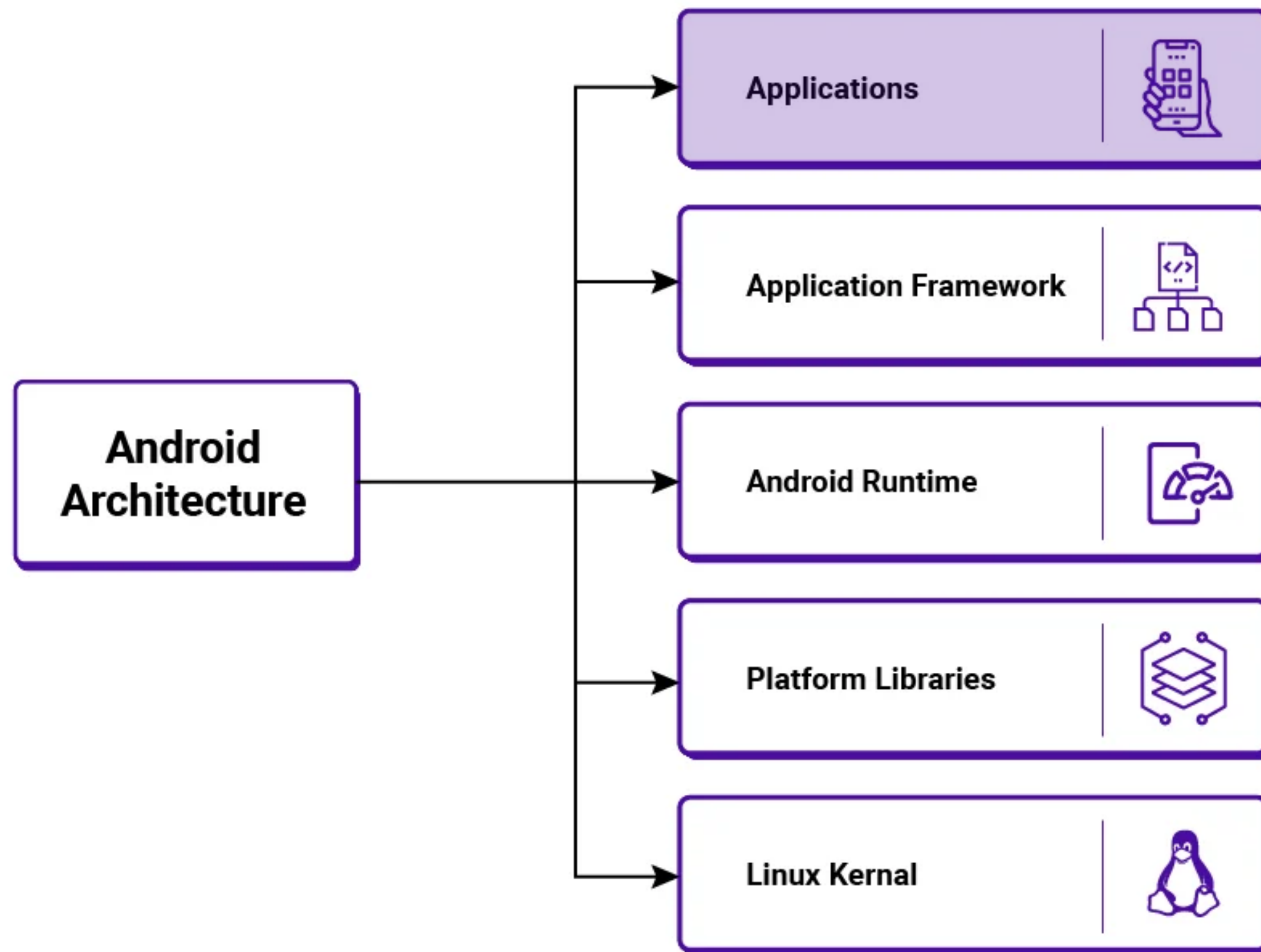
A evolução do Android

No começo era tudo mato

- A Android Inc. foi fundada em outubro de 2003 em Palo Alto, Califórnia, por Andy Rubin, Rich Miner, Nick Sears e Chris White.
- Inicialmente, visava aprimorar os sistemas operacionais de câmeras digitais
- Mudou seu foco para a criação de um S.O. móvel robusto para rivalizar com o Symbian e o Windows Mobile
- Como decisão estratégica, o Google adquiriu a Android Inc. em 2005
 - Manteve Android como um projeto de código aberto, aumentando sua aceitação e promovendo a inovação.

A evolução do Android

A arquitetura do Android



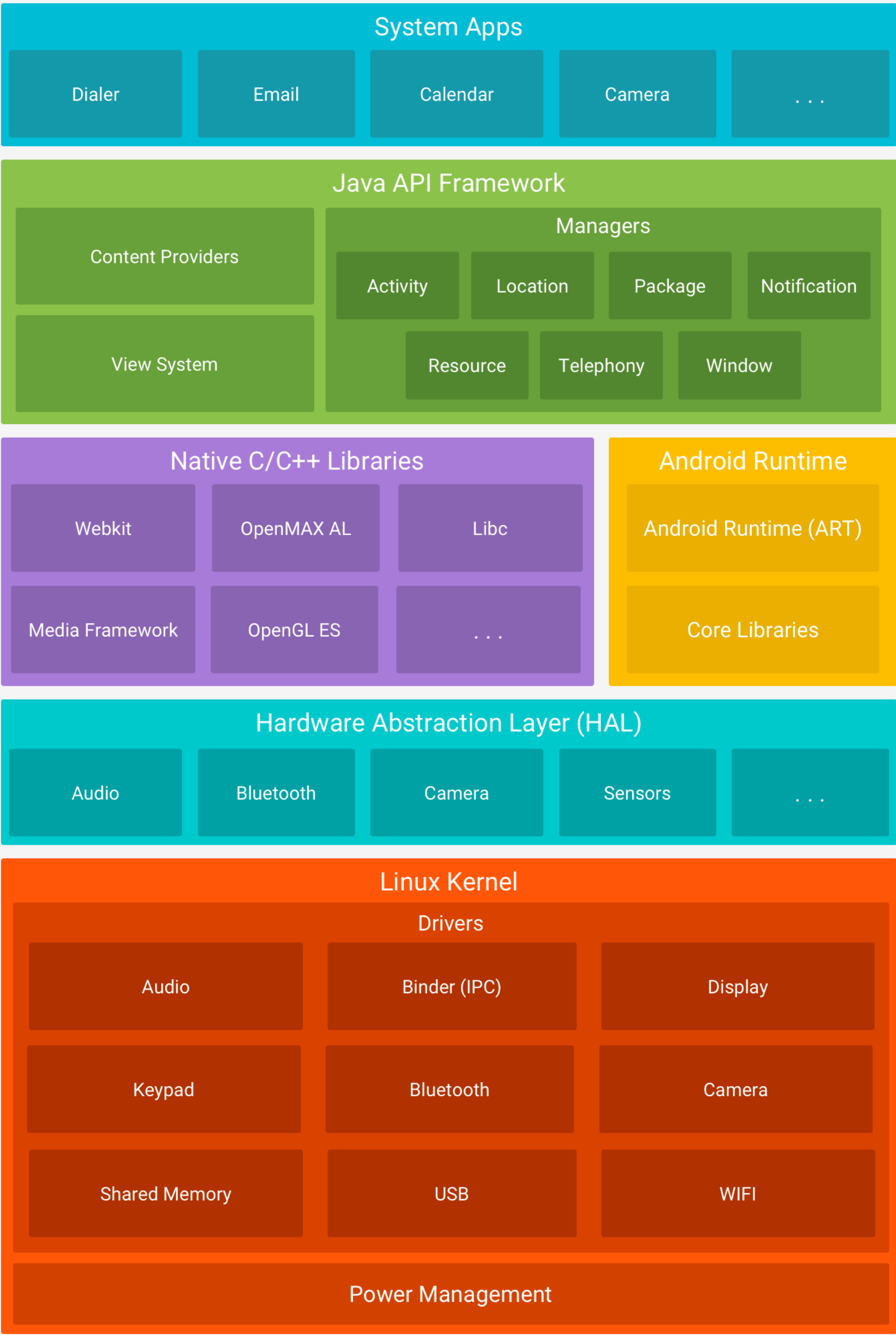
Aplicativos instalados pelo usuário e aplicativos do sistema

Contém os *buildingBlocks* necessários para criar aplicativos

A Android Runtime (ART) criada para executar várias máquinas virtuais em dispositivos com pouca memória

Provém acesso a uma série de bibliotecas como WebKit, SQLite, SSL, Libc.

Abstrações de hardware, drives e etc ...



A evolução do Android

Marcos no desenvolvimento de aplicações Android

- 2009 - Android 1.5 Cupcake (API 3)
 - Suporte a Widgets
- 2010 - Android 2.3 Gingerbread (API 9)
 - Suporte a Near Field Communication (NFC)
- 2011 - Android 3.0 Honeycomb (API 11)
 - Primeira versão apenas para tablet
 - Introduziu a API de Fragments
 - Adição da System bar, remoção de botões físicos



Fonte: <https://logos.fandom.com>



Fonte: <https://logos.fandom.com>



Fonte: <https://logos.fandom.com>

A evolução do Android

Marcos no desenvolvimento de aplicações Android

- 2013 - Android KitKat 4.4 (API 19 e API 20)
 - Suporte a wearables
- 2014 Android Lollipop (API 2021)
 - Sai Dalvik (JIT) entra ART (AOT)
 - Com *Just In Time*, o código Java era compilado para bytecode e, em seguida, interpretado em código de máquina no momento da execução
 - Com *Ahead of Time (AOT)*, todo o bytecode do aplicativo é traduzido para código de máquina nativo no momento da instalação
 - Introduziu o Material Design, Android Studio foi lançado



Fonte: <https://logos.fandom.com>



Fonte: <https://logos.fandom.com>

A evolução do Android

Marcos no desenvolvimento de aplicações Android

- 2015 - Android Marshmallow 6.0 (API 23)
 - Introduziu a concessão de permissões em tempo de execução
- 2016 - Android Nougat 7.0 (API 24)
 - Kotlin passou a ser suportado
- 2017 - Android Oreo 8.0 (API 26)
 - Introduziu os canais de notificação
 - Outra melhorias em notificações
 - Kotlin passou a ser a linguagem oficial



Fonte: <https://logos.fandom.com>



Fonte: <https://logos.fandom.com>

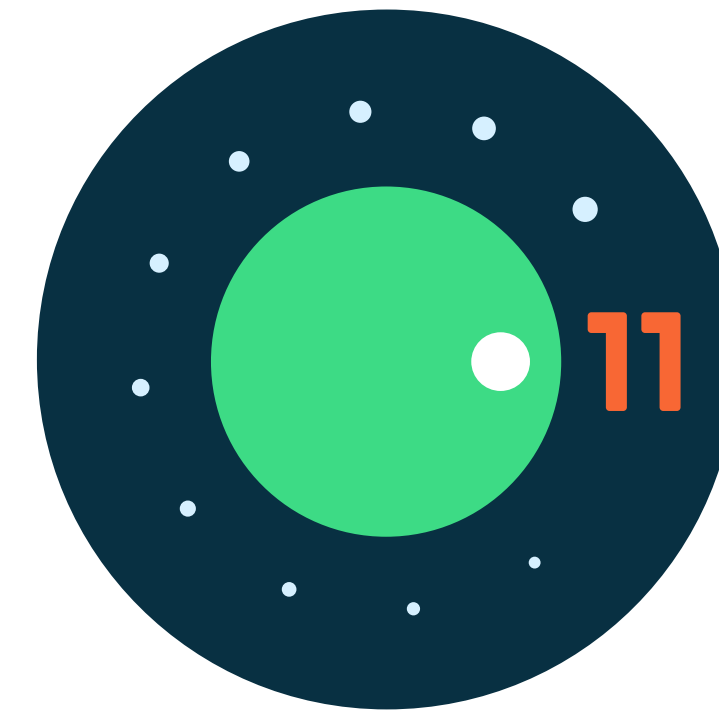


Fonte: <https://logos.fandom.com>

A evolução do Android

Marcos no desenvolvimento de aplicações Android

- 2020 - Android 11 (API 30)
 - Novo controle de permissões: One time permission
 - Jetpack Compose Alpha foi apresentado
- 2021 - Android 12 (API 31)
 - Material You (m3) "substitui" o Material Design
 - Permitir que o usuário controle total da paleta de cores do S.O aos aplicativos
 - Primeira versão oficial do JetPack compose foi lançada



Fonte: <https://logos.fandom.com>



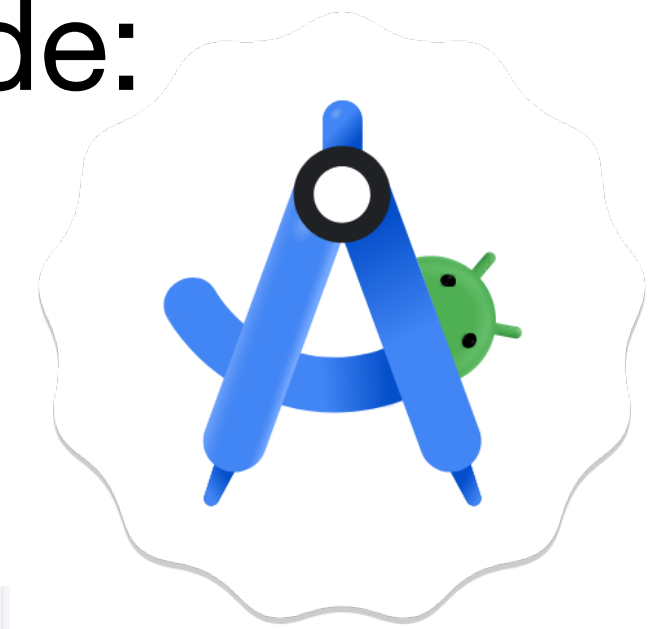
Fonte: <https://logos.fandom.com>



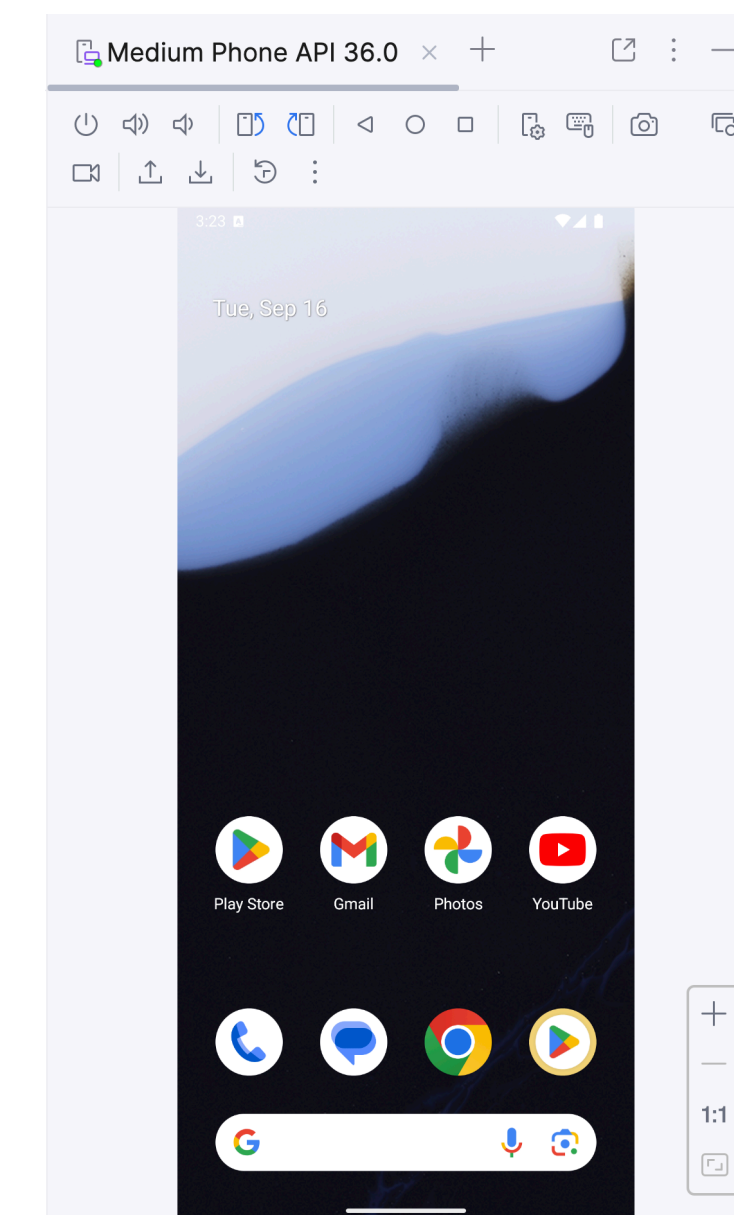
Hello World

Hello World

- Para desenvolver uma aplicativo Android você precisa no mínimo de:
 - A IDE oficial para o desenvolvimento Android: **Android Studio**
 - Pelo menos 8 GB de ram
 - Um dispositivo alvo
 - Emulador Android (AVD - Android Virtual Device)
 - Dispositivo real
 - Uma versão do **SDK do Android instalada**



Fonte: <https://logos.fandom.com>



Hello World

- O Android Studio traz tudo que é necessário e outras coisas mais:
 - Android Emulator
 - Google Play Services (APIs de mapas, Google Play etc.)
 - Android SDK build tools (ex.: dx, para converter .class para .dex)
 - Android SDK Platform-tools (ex.: adb, que gerencia dispositivos)
 - Android SDK Tools (ex.: ddms, para debugging)



Fonte: <https://logos.fandom.com>

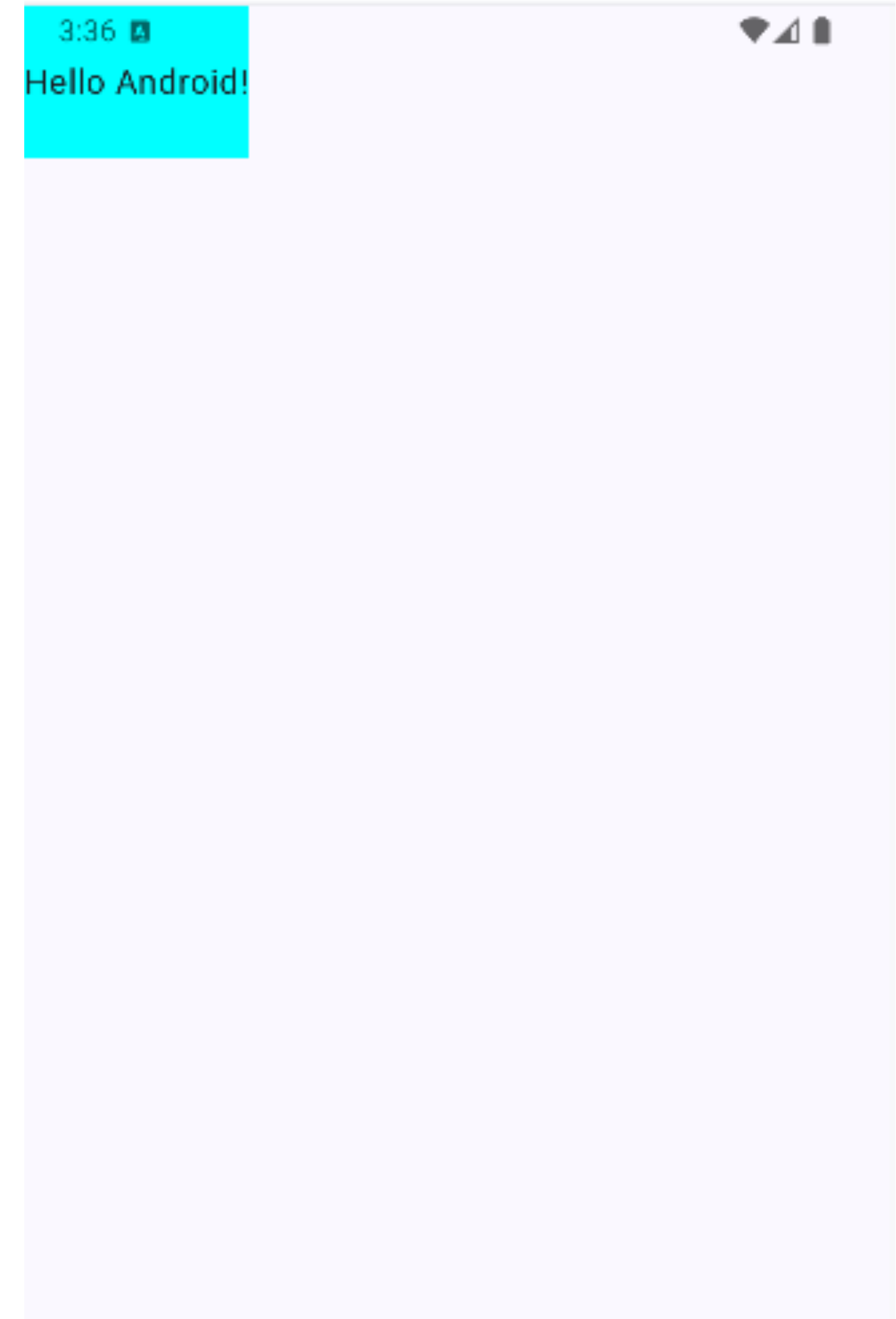
Hello World

- Sigam os seguintes [Codelabs](#) para criar seu primeiro aplicativo Android:
 - [Baixar e instalar o Android Studio](#)
 - [Criar seu primeiro app Android](#)
- Escolha uma das opções abaixo:
 - [Executar seu primeiro app no Android Emulator](#)
 - [Como conectar o dispositivo Android](#)



Hello World

- Ao final você deve ter terminal com algo similar a imagem ao lado



Bibliografia

- [How Many People Own Smartphones? \(2025-2029\)](#)
- [Smartphone OS sales market share evolution](#)
- [The surprising ways cellphones have changed our lives](#)
- [App stores - statistics & facts](#)
- [App Data Report 2025](#)
- [Progressive Web Apps](#)
- [O que é PWA \(Progressive Web Apps\) e porque usar?](#)

Bibliografia

- [Native App Development: Everything You Need to Know](#)
- [What is Hybrid Mobile App Development?](#)
- [Android version history](#)
- [Introduction to Android Development: History and Evolution of Android](#)
- [What is Android Operating System \(OS\): Features, and How it Works?](#)
- [The Evolution of Android Development: Key Changes from 2016 to 2024](#)
- [Android version history](#)

Por hoje é só