

Programação para Dispositivos Móveis

Introdução ao Desenvolvimento para Dispositivos Móveis
Sistema Operativo Android

2º Ano 1º Semestre
2020-2021



Sumário

- Introdução ao Desenvolvimento para Dispositivos Móveis
 - História
 - Vantagens / Desvantagens
 - Aplicações Nativas / Web / Híbridas
- Sistema Operativo Android
 - Android Studio
 - Instalação: versão atual: **4.0.1**
 - Execução
 - Criação do 1.º Projeto
 - Anatomia
 - SDK Manager
 - AVD Manager
 - Editor de Temas
 - Explorar



História



“Evolução” do telemóvel



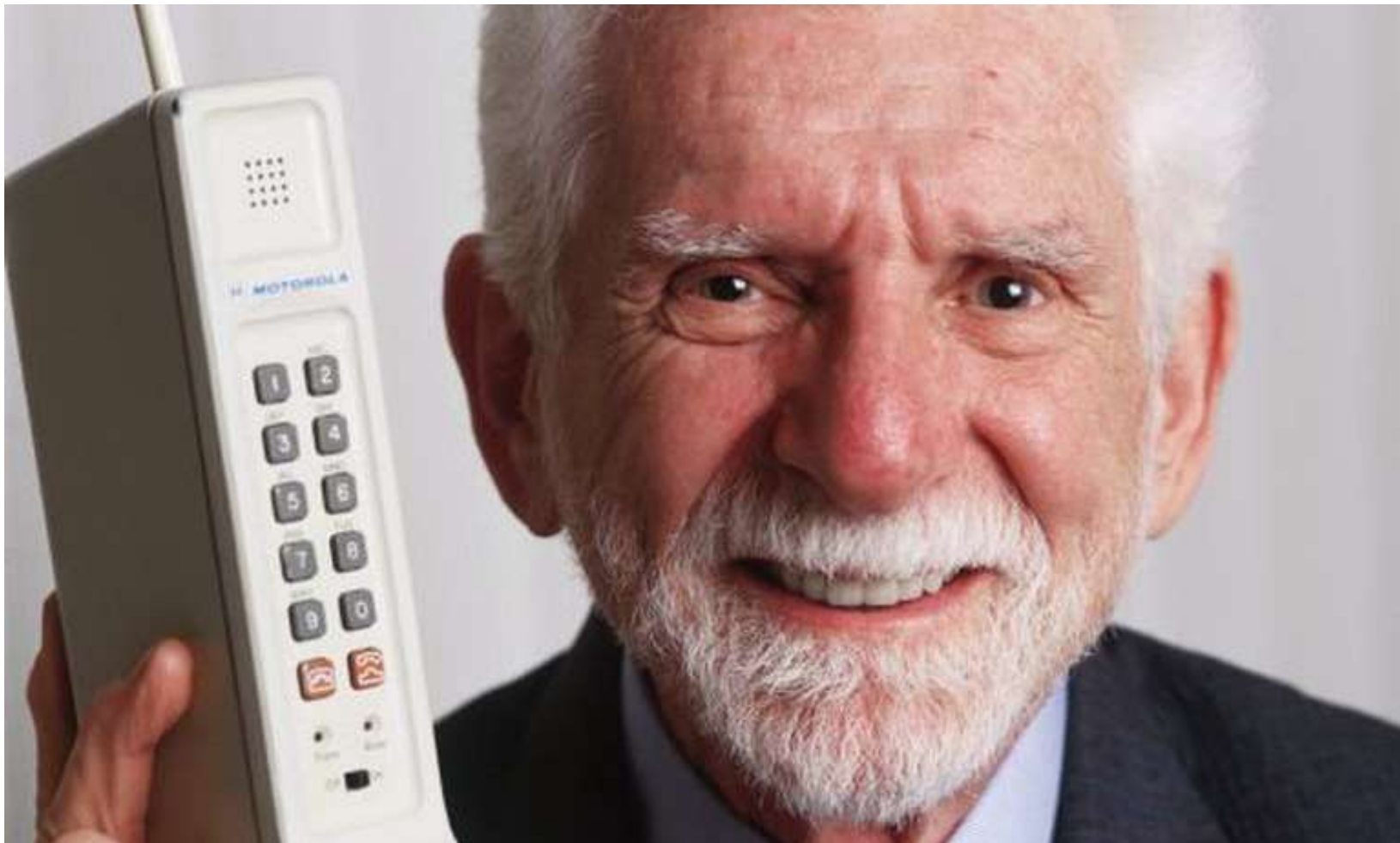
História

chamadas

- Era do “tijolo” (1974-1984)
 - Era portátil (mas não muito)
 - Mais caro do que usar telefones de cartão pré-pagos (payphones)
 - Baterias gigantescas
 - Principais utilizadores:
 - Vendedores, corretores de bolsa, ...
 - Passado algum tempo passaram a existir mais torres de rede celular e... os dispositivos foram ficando (um pouco) mais pequenos!



História



Martin Cooper



História

chamadas

SMS

- Era do “Candy Bar” (1984-1993)
 - Rede 2G network : GSM, CDMA, TDMA, iDEN
 - Alargamento da rede de torres celulares:
 - Menores necessidades de energia
 - Melhor qualidade de voz
 - Surge o SMS
 - Devido à prosperidade económica todos desejam possuir um telemóvel: EU, USA, e JP



História

chamadas

SMS

música & fotos

- Era do “Telemóvel com extras” (1991-2008)
 - Rede 2.5G: GPRS
 - Câmara
 - MMS
 - Capacidades de transmissão limitada de dados
 - Internet no telemóvel (fraca qualidade):
 - Preços elevados
 - Pouco consistente em termos de visualização de informação



História

chamadas

SMS

música & fotos

- Era do “Smartphone” (2001-2010)
 - 3G, HSDPA, WI-FI
 - Como um telefone “com extras”, mas que imitava um PC
 - Com o seu próprio sistema operativo (ex. Symbian)
 - Grandes écrans (usavam “stylus”)
 - A plataforma Mobile começa a ganhar alguma relevância
 - Email como aplicação principal



História

chamadas

SMS

música & fotos

apps

- Era do “Touch” (2007-agora)
 - 3G, 4G
 - Acelerómetros
 - GPS/Baseados na localização
 - Design centrado no utilizador
 - Impacto na utilização diária
 - Interfaces ricos
 - Plataforma multimédia pessoal
 - Web móvel – uso comum e diário



Vantagens e desvantagens

- Vantagens

- Experiência rápida e agradável por parte do utilizador
- Acesso ao conteúdo em modo *offline*
- Menor custo de acesso, dado que a grande parte da interface já se encontra instalada no dispositivo
- Acesso a recursos do próprio dispositivo móvel (GPS, Calendário, Contactos, Câmara Fotográfica, etc.)

- Desvantagens

- Existem demasiadas atualizações de versões
- Existem várias plataformas (IOS, Android, etc.), o que requer desenvolver a mesma aplicação para cada plataforma

Aplicações Nativas / Web / Híbridas

○ Aplicações Nativas

- Desenvolvidas para um sistema específico (Android, IOS, etc.)
- Descarregadas a partir de uma Loja de Aplicações (Play Store, iStore, etc.)
- Linguagens de Programação específicas (Java, Swift, etc.)
- Maior rapidez e oferecem uma experiência de utilização otimizada ao dispositivo
- Utilização dos recursos existentes no dispositivo (câmara fotográfica, GPS, serviço de notificações, calendário, etc.)
- Desenvolver aplicação para cada sistema. Aguardar aprovação da Loja de Aplicações.
- Cada atualização, será necessário atualizar para todos os sistemas

Aplicações Nativas / Web / Híbridas

- Web
 - Acesso através de um *browser*, não requerendo a sua instalação no dispositivo móvel
 - A maioria é desenvolvida em *JavaScript*, CSS e HTML5 (Linguagens Web)
 - Apresentação de dados
 - Não permitem a utilização de recursos do dispositivo móvel
 - Desenvolvimento mais curto e menos dispendioso
 - Difíceis de encontrar, não existindo uma loja de aplicações associada a estes tipos de aplicações
 - Necessário ligação à Internet

Aplicações Nativas / Web / Híbridas

- Híbridas
 - Aplicações com comportamento nativo, mas desenvolvidas com tecnologias web
 - Desenvolvidas com recursos a Frameworks específicas, como o **React Native** (<https://facebook.github.io/react-native/>)
 - Desenvolvimento mais curto e com menos custos que as aplicações nativas
 - Acesso pelas Lojas de Aplicações, sendo invisível a diferença entre este tipo de aplicações das aplicações nativas
 - Permite usar os recursos do dispositivo, mas de uma forma menos eficiente que as aplicações nativas

Web vs Nativas

Web

- Acesso aos recursos do dispositivo é limitado. Embora haja APIs para o GPS e outras, a flexibilidade é menor.
- Mais complexidade na manutenção do estado da aplicação (modo *offline*, *multitasking*).

Nativas

- O acesso aos recursos é efetuada de uma forma rápida, aumentando o leque de oportunidades e possibilidades.
- Manutenção do estado da aplicação é muito mais fácil e versátil.

Web vs Nativas

Web

- Qualquer dispositivo (móvel ou desktop) com um browser pode aceder.
- Qualquer lançamento efetuado fica disponível logo de imediato.

Nativas

- Apenas destinados a um sistema. Qualquer aplicação terá que ser desenvolvida para cada sistema.
- O processo de *deployment* pode durar algum tempo e até pode ser recusado pela Loja.

Web vs Nativas

Web

- Tempo de desenvolvimento menor, tanto no lançamento como nas atualizações. As alterações são rápidas.
- Usam tecnologias standard e já conhecidas (html, css, javascript).
- **Imensos dispositivos e *browsers*, sendo um ecossistema aberto e instável**

Nativas

- **Menos programadores de aplicações nativas. Terá que haver programador para cada tipo de programação. Alterações (mesmo pequenas), terá que haver um *deploy*.**
- Usam tecnologias nativas, o que terá que haver um período de aprendizagem.
- São oferecidos SDK, frameworks e sistemas de base aos programadores.

Web vs Nativas

Web

- É preciso implementar um sistema de pagamento se quiser criar uma aplicação paga.
- Usabilidade complexa e difícil de gerir

Nativas

- Havendo uma loja, já existe um sistema de pagamento integrado
- Velocidade superior, pois em código nativo não há download da interface.

Desenvolvimento para Dispositivos Móveis

- Qual usar ?
- Para a melhor escolha é necessário ter em conta vários fatores:
 - Objetivos
 - Funcionalidades
 - Público-alvo
 - Orçamento
 - Tempo de desenvolvimento

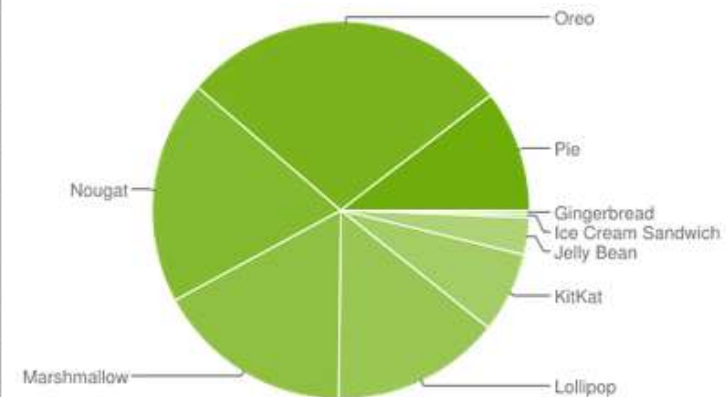


Android – Conceito e Evolução

- Sistema operativo mobile baseado em Linux
- Criado pela Android Inc. (suportada pelo Google)
 - Android Inc. comprada pelo Google em 2005
- Revelado em 2007 no contexto do Open Handset Alliance
- Open-source sob a licença Apache
- Out 2012: 700,000 apps

Android – Conceito e Evolução

Versão ↕	Codinome ↕	Data de lançamento ↕	Nível API ↕	Distribuição ↕
10	Android 10 (Q)	3 de setembro de 2019	29	16,12%
9	Pie	6 de agosto de 2018	28	37,4%
8.1	Oreo	15 de dezembro de 2017	27	11,29%
8.0		21 de agosto de 2017	26	7,37%
7.1.x	Nougat	5 de dezembro de 2016	25	4,3%
7.0.x		22 de agosto de 2016	24	6,24%
6.0.-6.0.1	Marshmallow	05 de Outubro de 2015	23	8,72%
5.1-5.1.1	Lollipop	10 de março de 2015	22	4,84%
5.0-5.0.2		12 de novembro de 2014	21	1,21%
4.4.4.4.4	KitKat	31 de outubro de 2013	19	1,87%
4.3.x	Jelly Bean	24 de julho de 2013	18	menos de 0,1%
4.2.x		13 de novembro de 2012	17	menos de 0,1%
4.1.x		9 de julho de 2012	16	menos de 0,1%
4.0.3-4.0.4	Ice Cream Sandwich	16 de dezembro de 2011	15	menos de 0,1%
2.3.3-2.3.7	Gingerbread	9 de fevereiro de 2011	10	menos de 0,1%
2.2-2.2.3	Froyo	20 de maio de 2010	8	menos de 0,1%
2.0-2.1	Eclair	26 de outubro de 2009	5	menos de 0,1%
1.6	Donut	15 de setembro de 2009	4	menos de 0,1%
1.5	Cupcake	27 de abril de 2009	2	menos de 0,1%
1.0-1.1	Petit Four ou Beta	23 de setembro de 2008	1	não há registros



Fonte: <https://pt.wikipedia.org/wiki/Android/>

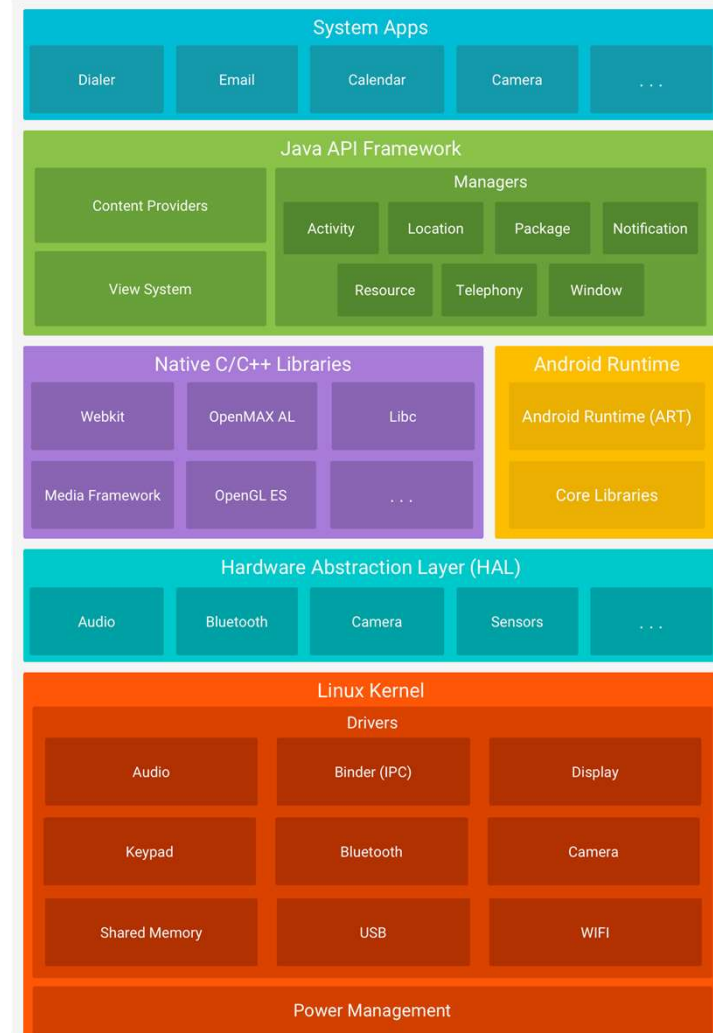


Android – Conceito e Evolução

- Sistema Operativo Móvel.
- Fornece uma estrutura de aplicativo avançado, permitindo a criação de aplicativos e jogos inovadores para dispositivos móveis.
- Ambiente de Linguagem Java.
- Estrutura de aplicativo adaptativa, permitindo fornecer recursos consoante as configurações dos dispositivos.
 - É possível criar diferentes arquivos XML de layout para vários tipos de tamanhos de ecrã. Assim, o sistema determina qual dos arquivos XML deverá aplicar tendo em conta o ecrã.

Android - Arquitetura

- Pilha de software com base em Linux de código aberto criada para diversos dispositivos.
- 6 Componentes (camadas):
 - Kernel do Linux
 - Camada de Abstração de Hardware (HAL)
 - Android Runtime
 - Bibliotecas C/C++ nativas
 - Estrutura da Java API
 - Aplicativos do sistema



Fonte: <https://developer.android.com/guide/platform/>



Android - Arquitetura

- Kernel do Linux:
 - Aproveitar os **recursos de segurança** e que os fabricantes dos dispositivos desenvolvam drivers de hardware para um kernel conhecido.
- Camada de Abstração de Hardware (HAL):
 - módulos de biblioteca que implementam uma interface para um tipo específico de componente de hardware (ex: módulo de câmera ou Bluetooth)
- Android Runtime:
 - cada aplicativo executa o próprio processo com uma instância própria do Android Runtime (ART) – dispositivos a partir da versão 5.0 do Android (API nível 21)
- Bibliotecas C/C++ nativas:
 - A plataforma Android fornece as Java Framework APIs para expor a funcionalidade de algumas dessas bibliotecas nativas aos aplicativos.
- Estrutura da Java API:
 - O conjunto completo de recursos do SO Android está disponível pelas APIs programadas na linguagem Java.
- Aplicativos do sistema:
 - Os aplicativos do sistema funcionam como aplicativos para os utilizadores. Assim, o programador pode usar, por exemplo, o aplicativo de SMS para enviar mensagem SMS, não sendo necessário desenvolver esse aplicativo.



Android - Funcionalidades

- Ambiente de aplicações que permite a reutilização e substituição de componentes
- Máquina virtual Dalvik otimizada para dispositivos móveis (substituída no Lollipop, pelo ART)
- Browser integrado (inicialmente) baseado no projecto open-source Webkit
- Gráficos baseados em biblioteca gráfica; gráficos 3D baseados em OpenGL
- SQLite para armazenamento de informação estruturada
- Suporte para diversos formatos de áudio, vídeo e imagem (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF)
- Telefonia GSM (dependente do hardware)
- Bluetooth, EDGE, 3G, e WiFi (dependente do hardware)
- Camera, GPS, bússola, e acelerómetro (dependente do hardware)
- Ambiente de desenvolvimento rico que inclui um emulador de dispositivos, ferramentas de debug, profiling de memória e de desempenho, e um plugin para o IDE Eclipse (actualmente o Android Studio, baseado no IntelliJ)

Como começar ?

- Android Developers



- <https://developer.android.com/guide/>

- Xamarin Developers



- <https://developer.xamarin.com/pt-br/guides/>

- Xamarin Forms
 - Cross-Plataform
 - Android
 - iOS
 - Mac
 - ...



Android Studio

- Plataforma de desenvolvimento para **Android**.
 - Criada pela **Google**.
 - Baseada no IDE IntelliJ IDEA da JetBrains, semelhante ao popular **Eclipse** ou ao **Netbeans**.
 - Linguagem de Programação **Java**.
 - Versão atual:



Android Studio provides the fastest tools for building apps on every type of Android device.

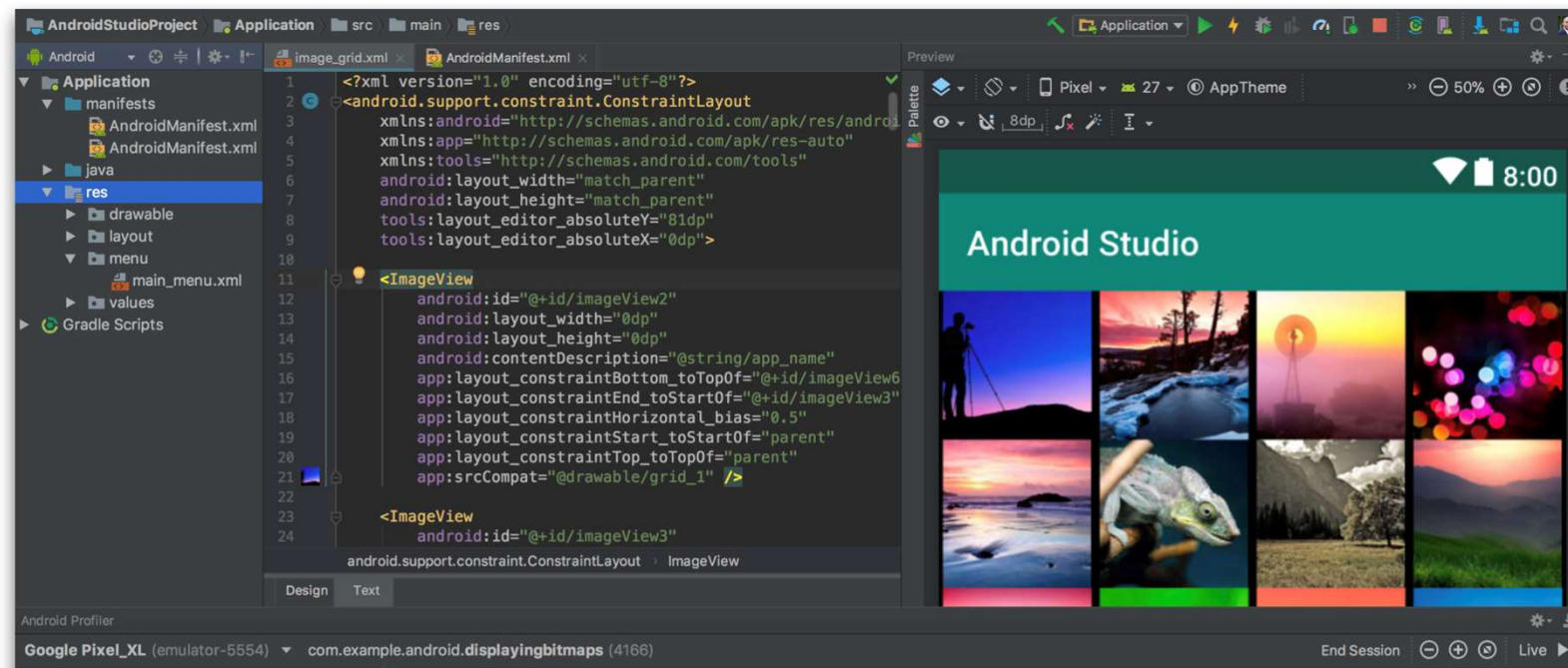
DOWNLOAD ANDROID STUDIO

4.0.1 for Windows 64-bit (871 MB)



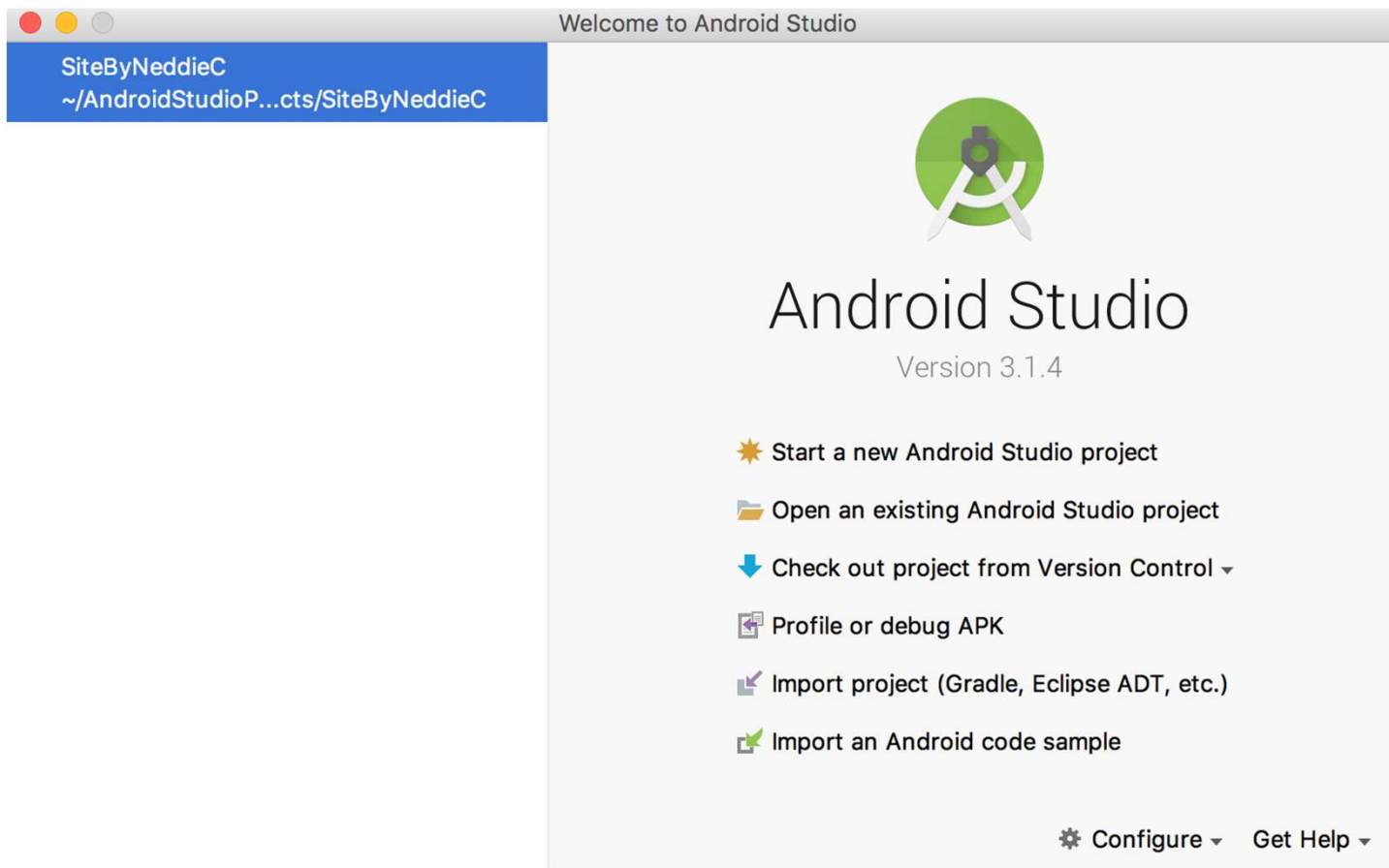
Android Studio

- Instalação grátis
 - <https://developer.android.com/studio/>



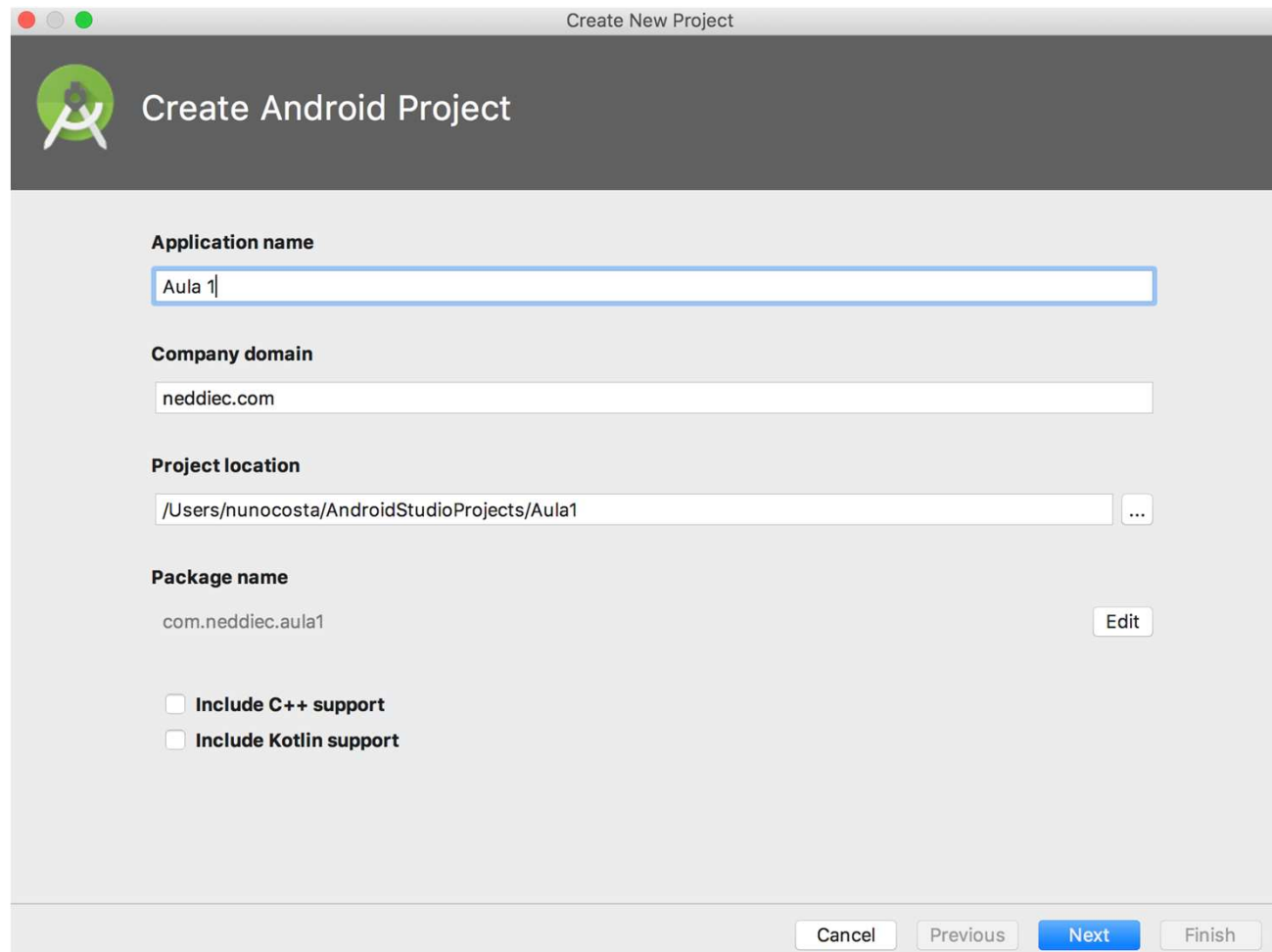
Android Studio

○ Executar



Android Studio

- Criação do primeiro projeto



The screenshot shows the 'Create New Project' dialog box in Android Studio. The dialog has a title bar with standard window controls and the text 'Create New Project'. Below the title bar is a header area with the Android logo and the text 'Create Android Project'. The main area contains several input fields and checkboxes:

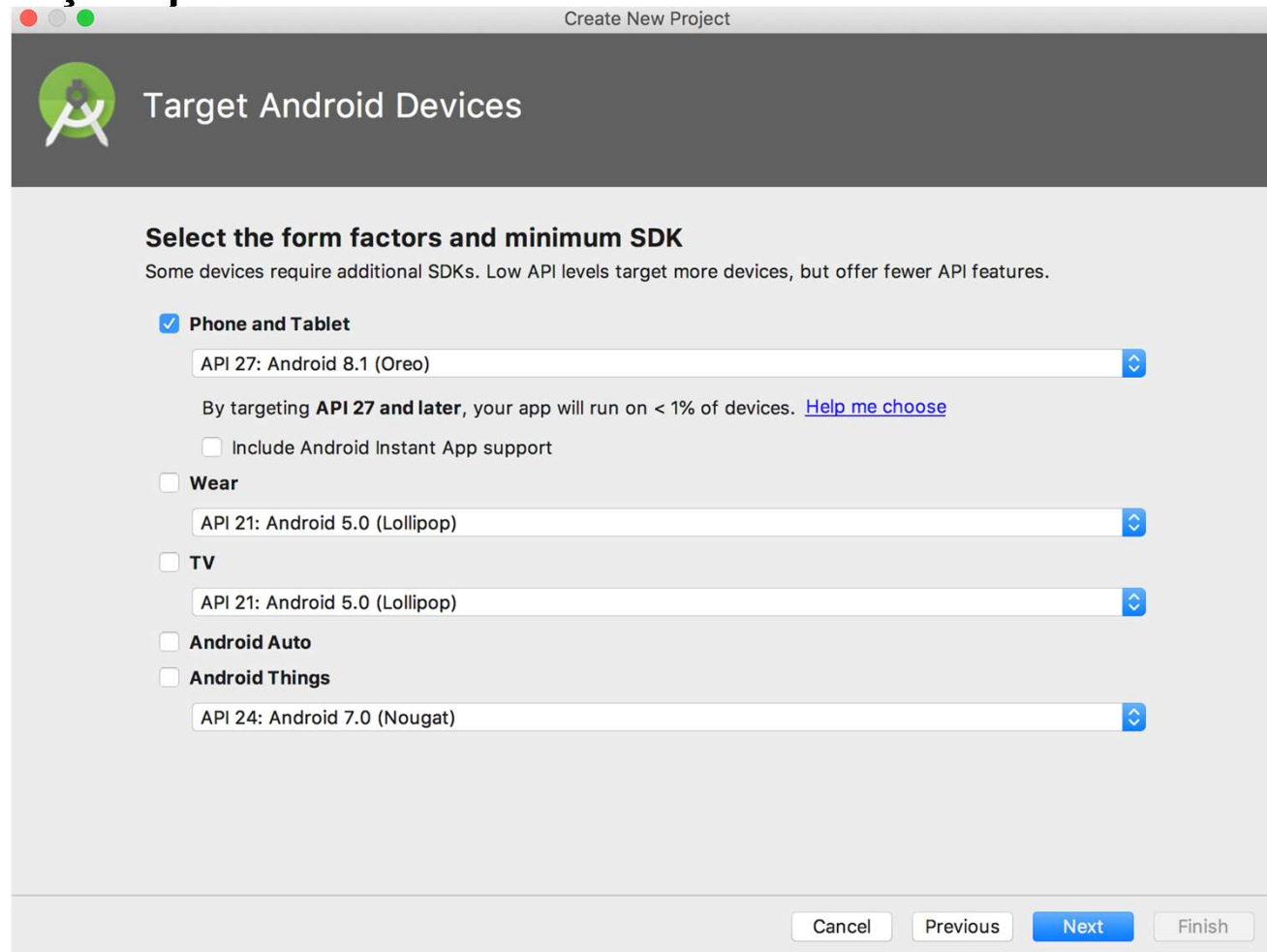
- Application name:** A text field containing 'Aula 1'.
- Company domain:** A text field containing 'neddiec.com'.
- Project location:** A text field containing '/Users/nunocosta/AndroidStudioProjects/Aula1' with a browse button ('...') to its right.
- Package name:** A text field containing 'com.neddiec.aula1' with an 'Edit' button to its right.
- Include C++ support:** An unchecked checkbox.
- Include Kotlin support:** An unchecked checkbox.

At the bottom of the dialog are four buttons: 'Cancel', 'Previous', 'Next' (highlighted in blue), and 'Finish'.



Android Studio

- Aplicação para ser executada em



The screenshot shows the 'Create New Project' dialog in Android Studio. The title bar says 'Create New Project'. The main header is 'Target Android Devices' with the Android logo. Below this, the section is titled 'Select the form factors and minimum SDK'. A note states: 'Some devices require additional SDKs. Low API levels target more devices, but offer fewer API features.'

The following options are listed:

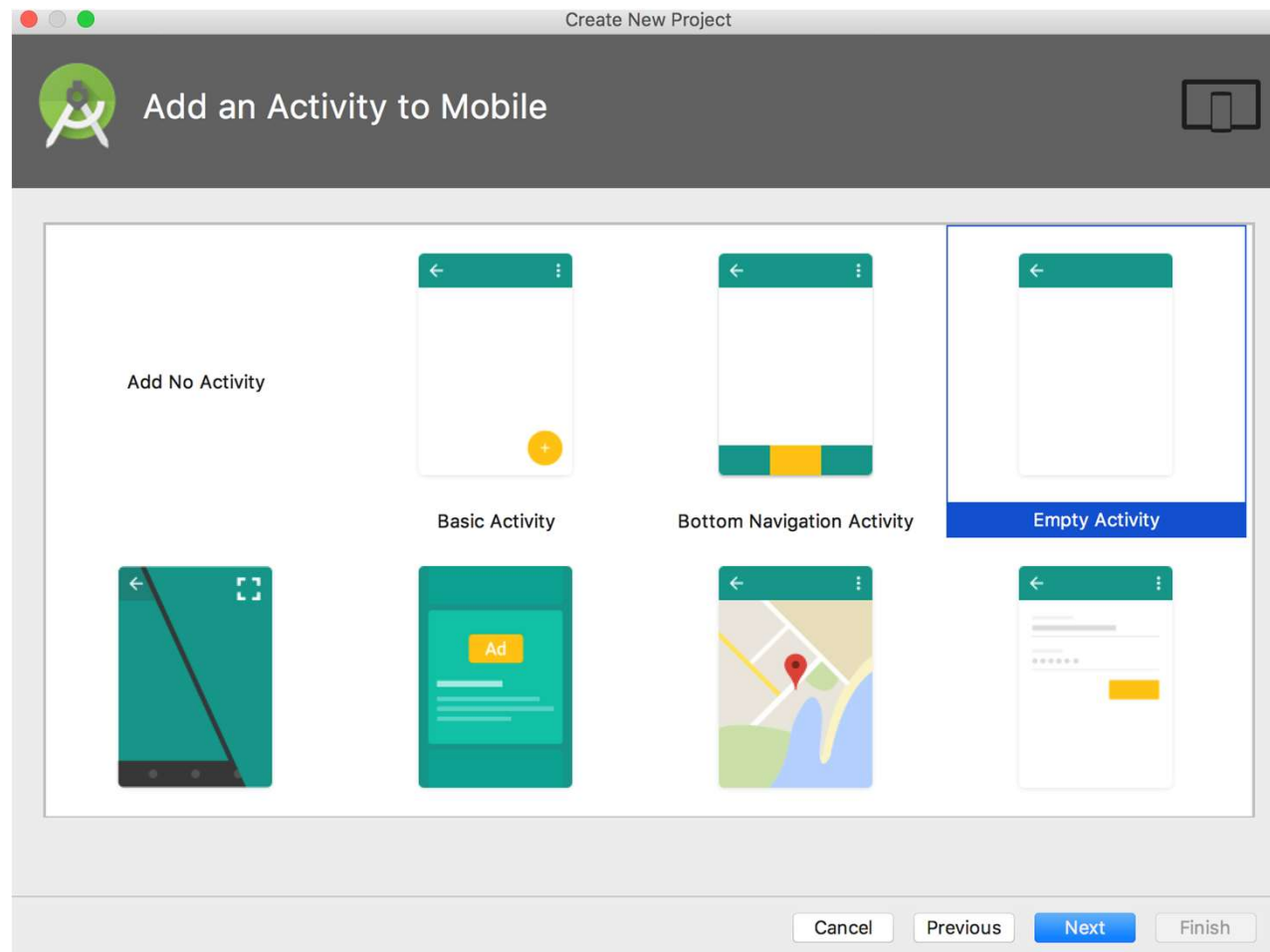
- ☒ **Phone and Tablet**
API 27: Android 8.1 (Oreo)
By targeting **API 27 and later**, your app will run on < 1% of devices. [Help me choose](#)
☐ Include Android Instant App support
- ☐ **Wear**
API 21: Android 5.0 (Lollipop)
- ☐ **TV**
API 21: Android 5.0 (Lollipop)
- ☐ **Android Auto**
- ☐ **Android Things**
API 24: Android 7.0 (Nougat)

At the bottom, there are four buttons: 'Cancel', 'Previous', 'Next' (highlighted in blue), and 'Finish'.



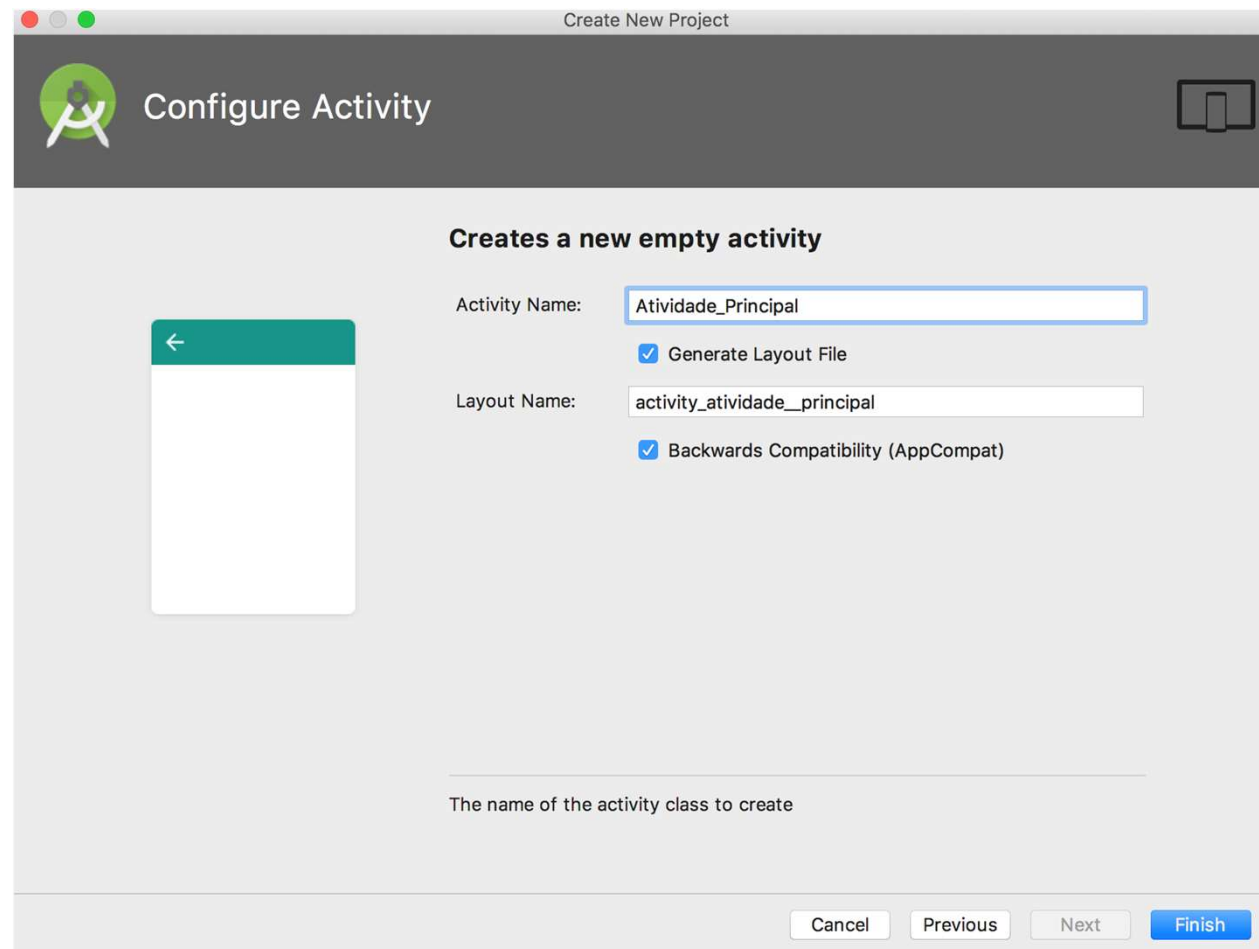
Android Studio

- Adicionar Atividades pré-definidas



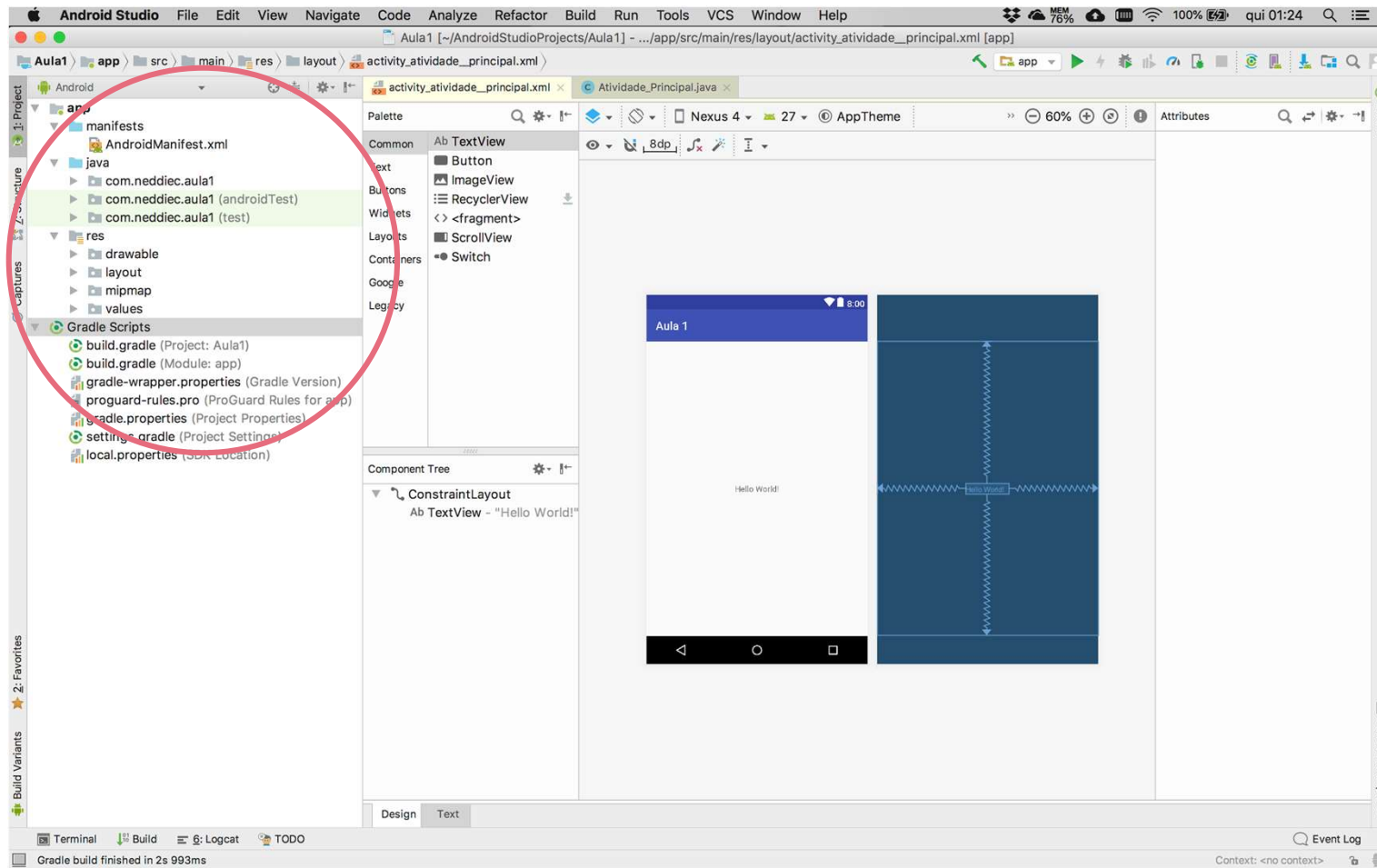
Android Studio

- Criar uma Atividade vazia



Android Studio

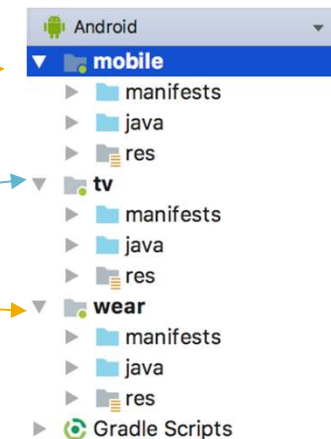
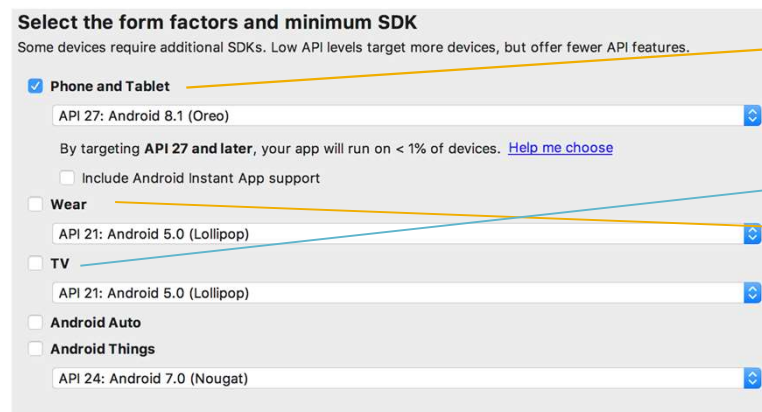
○ Anatomia do Android Studio



Android Studio

○ Anatomia do Android Studio

- Um projeto Android é constituído por **módulos**.
 - Aplicações
 - Biblioteca
 - Testes
- Quando existe apenas uma plataforma de execução e selecionado a opção “Phone and Tablet” é criado a **app** (módulo de aplicação único)

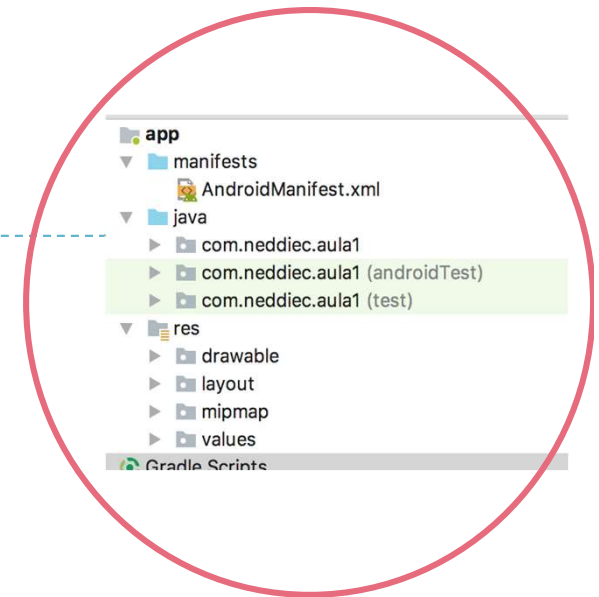


Android Studio

○ Anatomia do Android Studio

○ Ficheiro de Manifesto

- Descrição dos componentes da aplicação
 - Exemplos: Atividades, Serviços
- Definição de permissões de acesso a recursos
 - Exemplos: GPS, Internet



```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.neddiec.aula1">

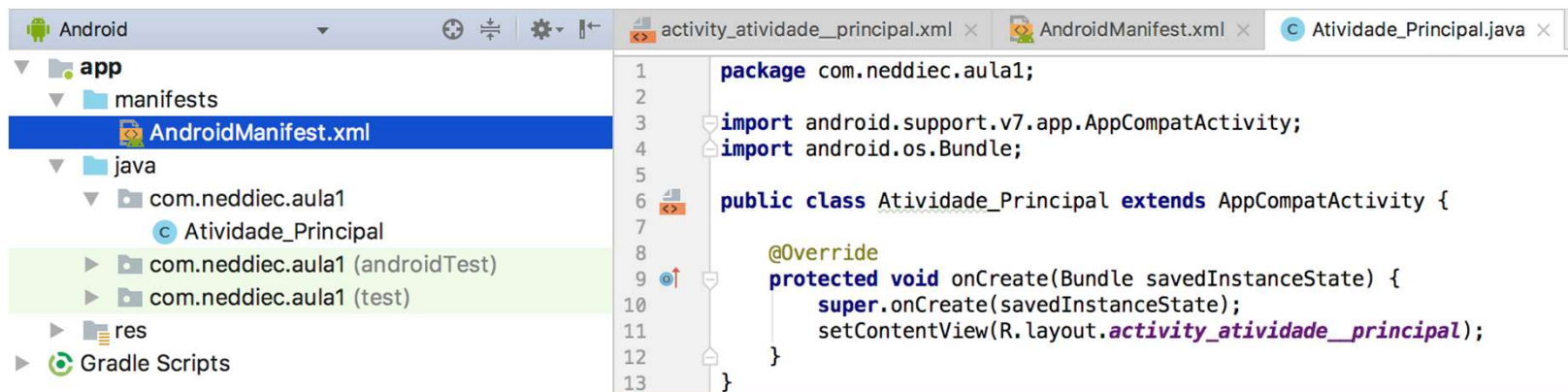
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="Aula 1"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".Atividade_Principal">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

Android Studio

- Anatomia do Android Studio
 - Código Fonte JAVA
 - Separação do código principal da aplicação em relação aos ficheiros de testes.



- Código gerado automaticamente pelo Android Studio



Android Studio

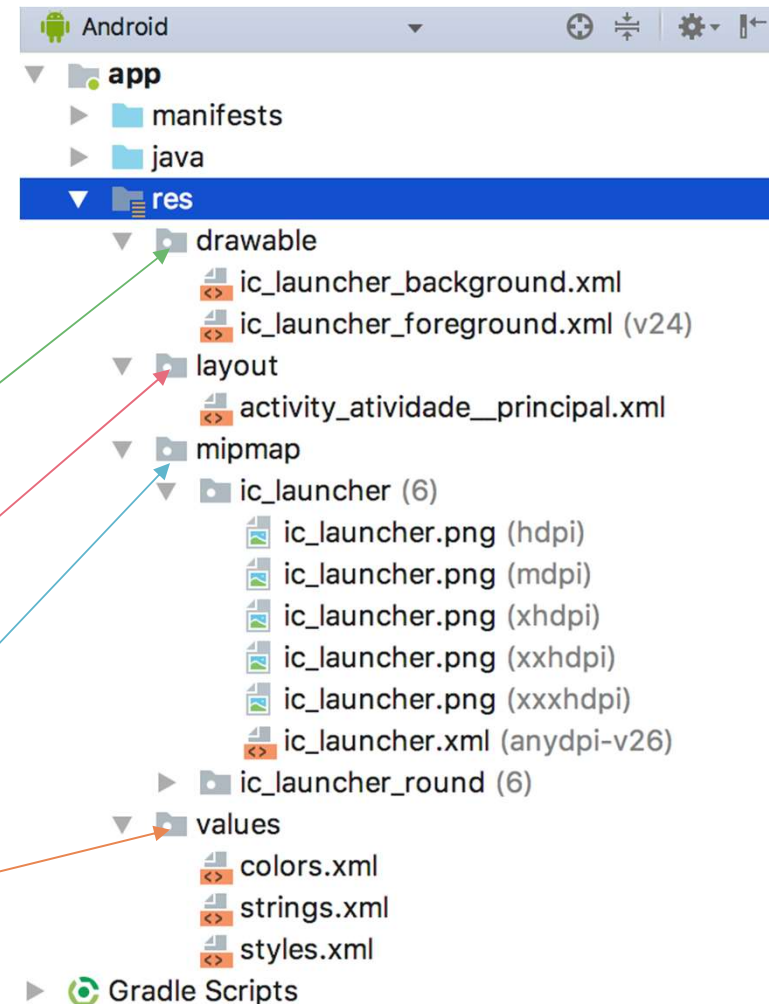
- Anatomia do Android Studio
 - Recursos
 - Separação entre a atividade e os recursos => não juntar a lógica do negócio com a interface gráfica

Ficheiros bitmap (png, gif, jpeg), imagens e ficheiros XML que descrevam formas e objetos

Ficheiros XML que definem layouts gráficos. Existe a possibilidade de alterar de forma visual (design) ou por texto (XML)

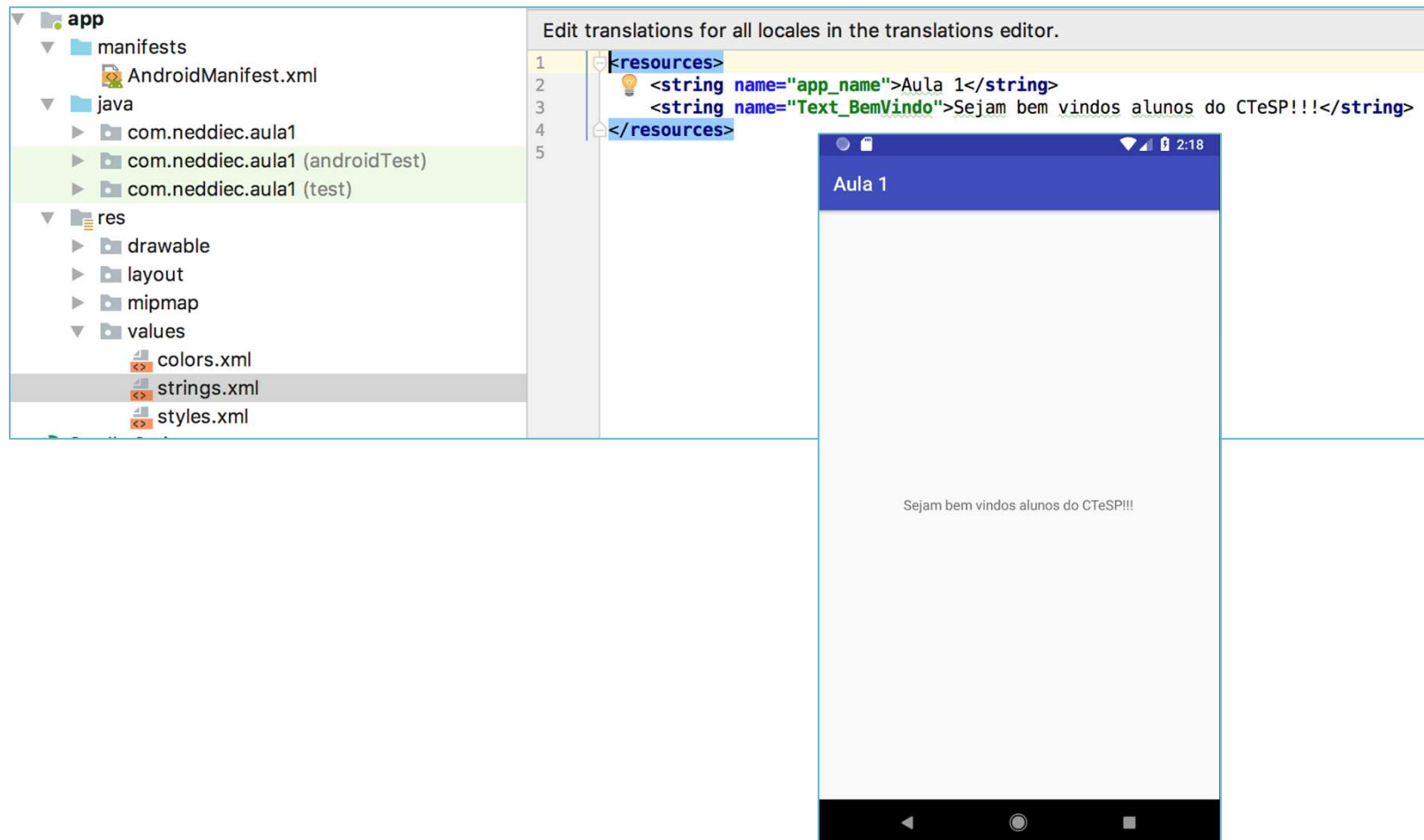
Ícones em diferentes densidades. Possibilidade de escolher o ícone dependendo da resolução do dispositivo

Ficheiros XML que definem valores usados na aplicação. Por exemplo, variáveis das cores, dimensões das margens de ecrã, texto usado e temas gráficos.



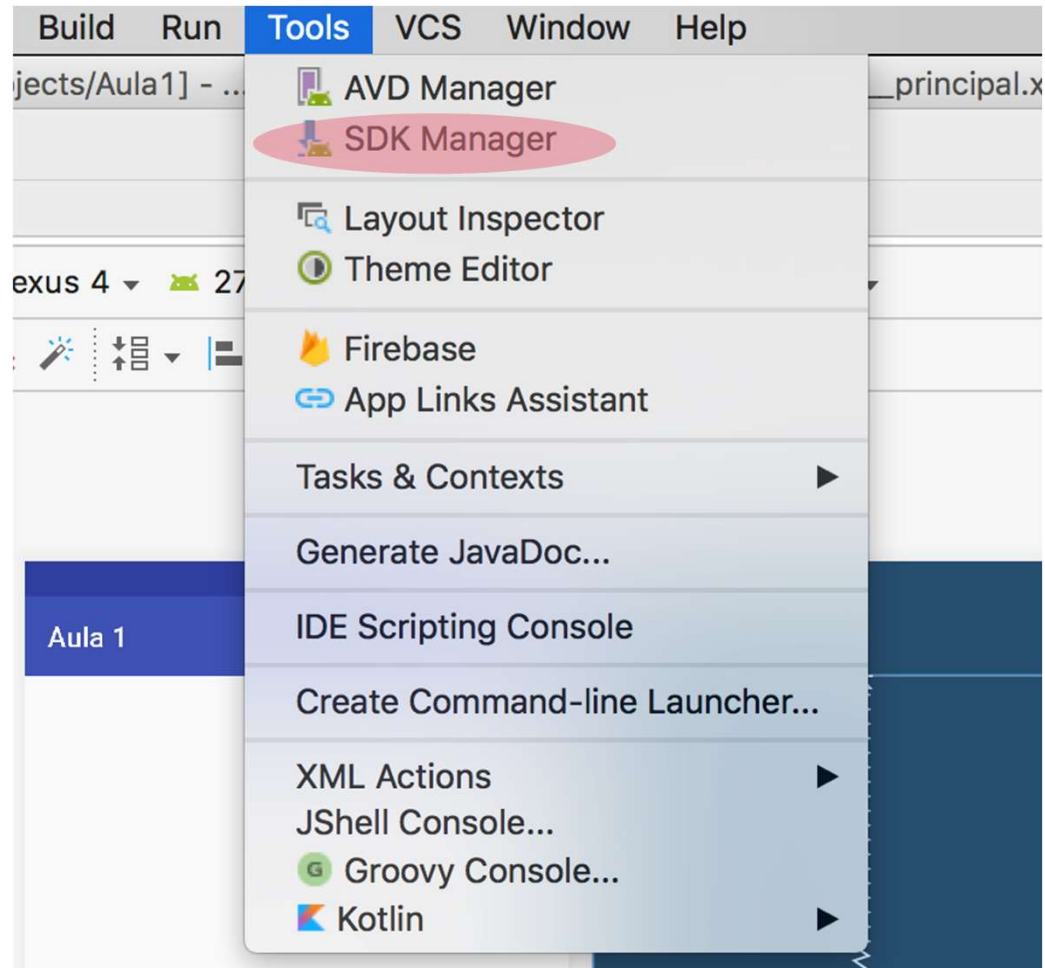
Android Studio

- Alterar o texto, usando “resource”



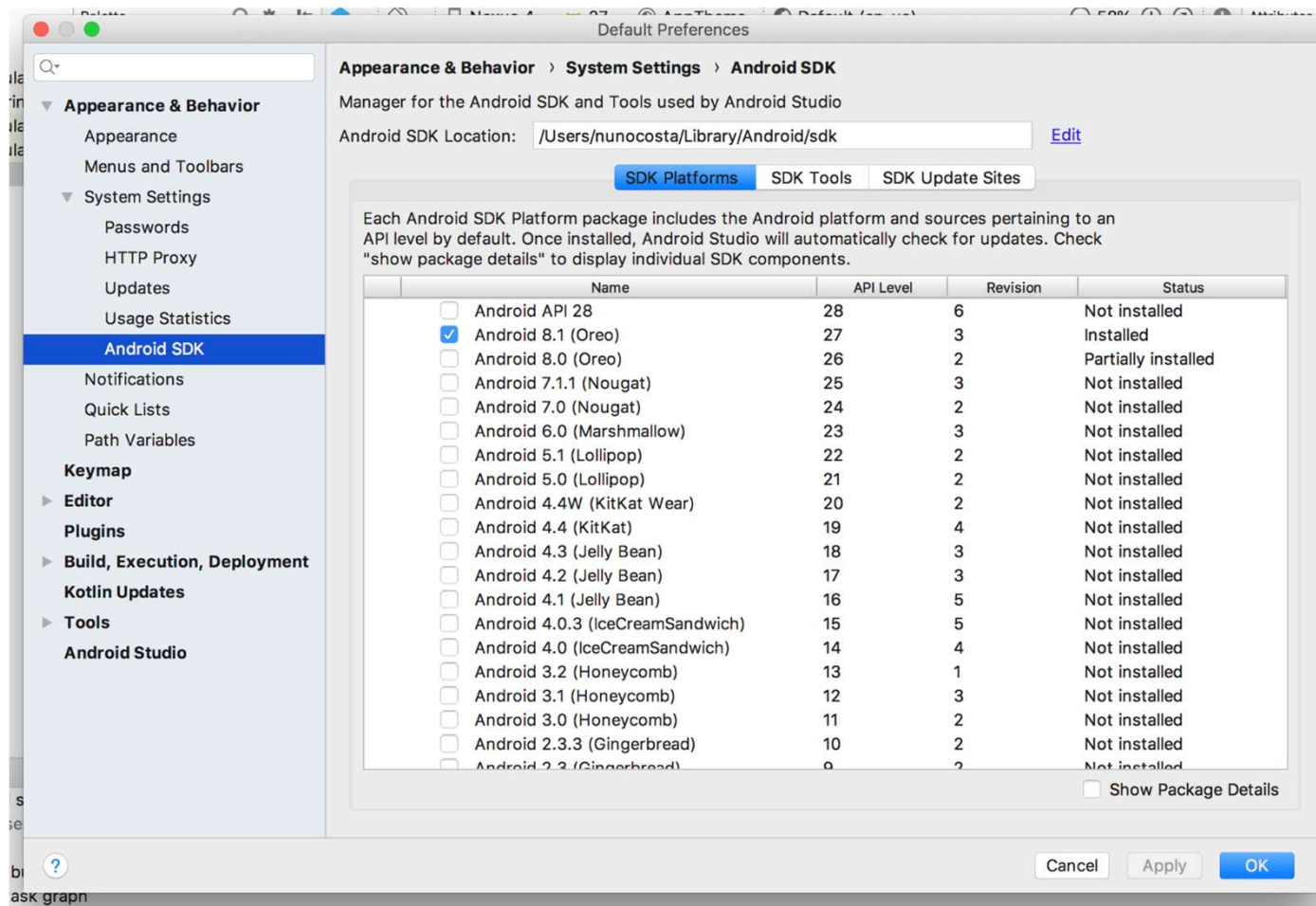
Android Studio

- SDK Manager
 - Separa ferramentas, plataformas e outros componentes em pacotes que podem ser descarregados individualmente.



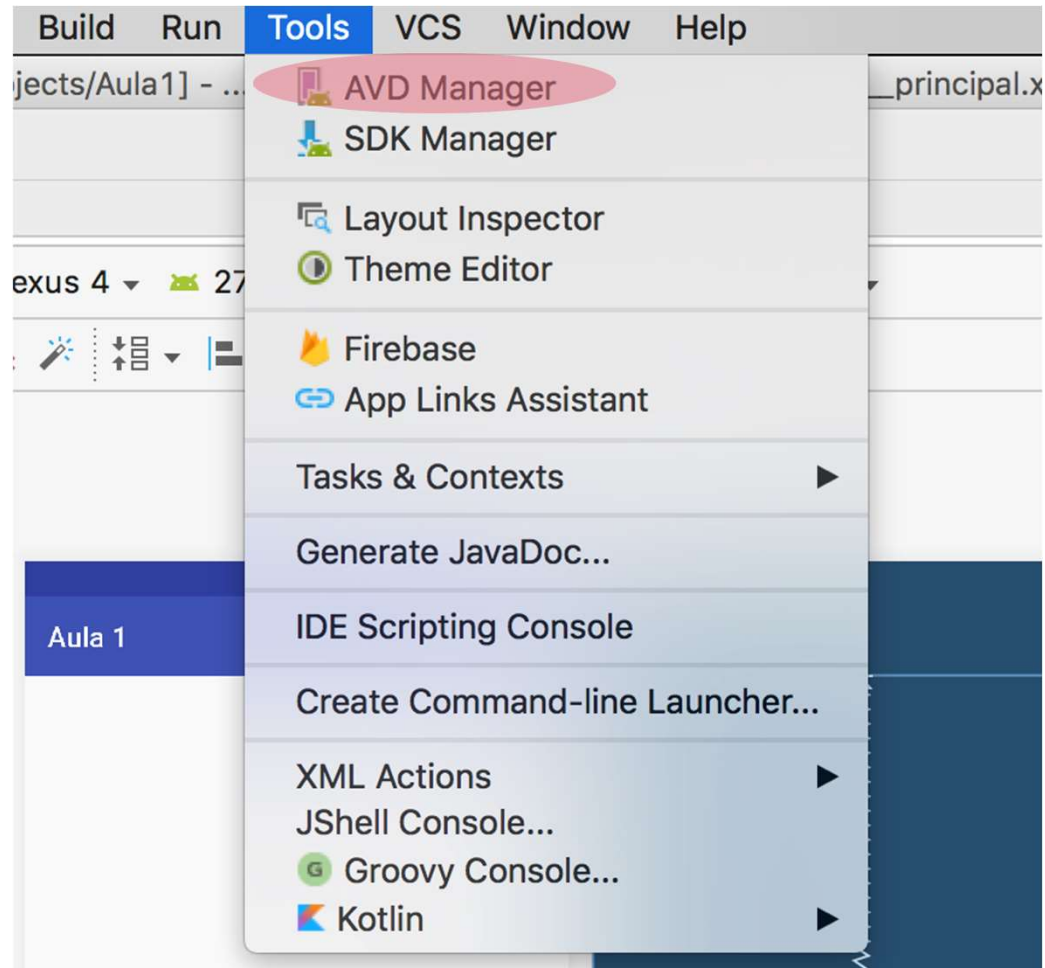
Android Studio

○ SDK Manager



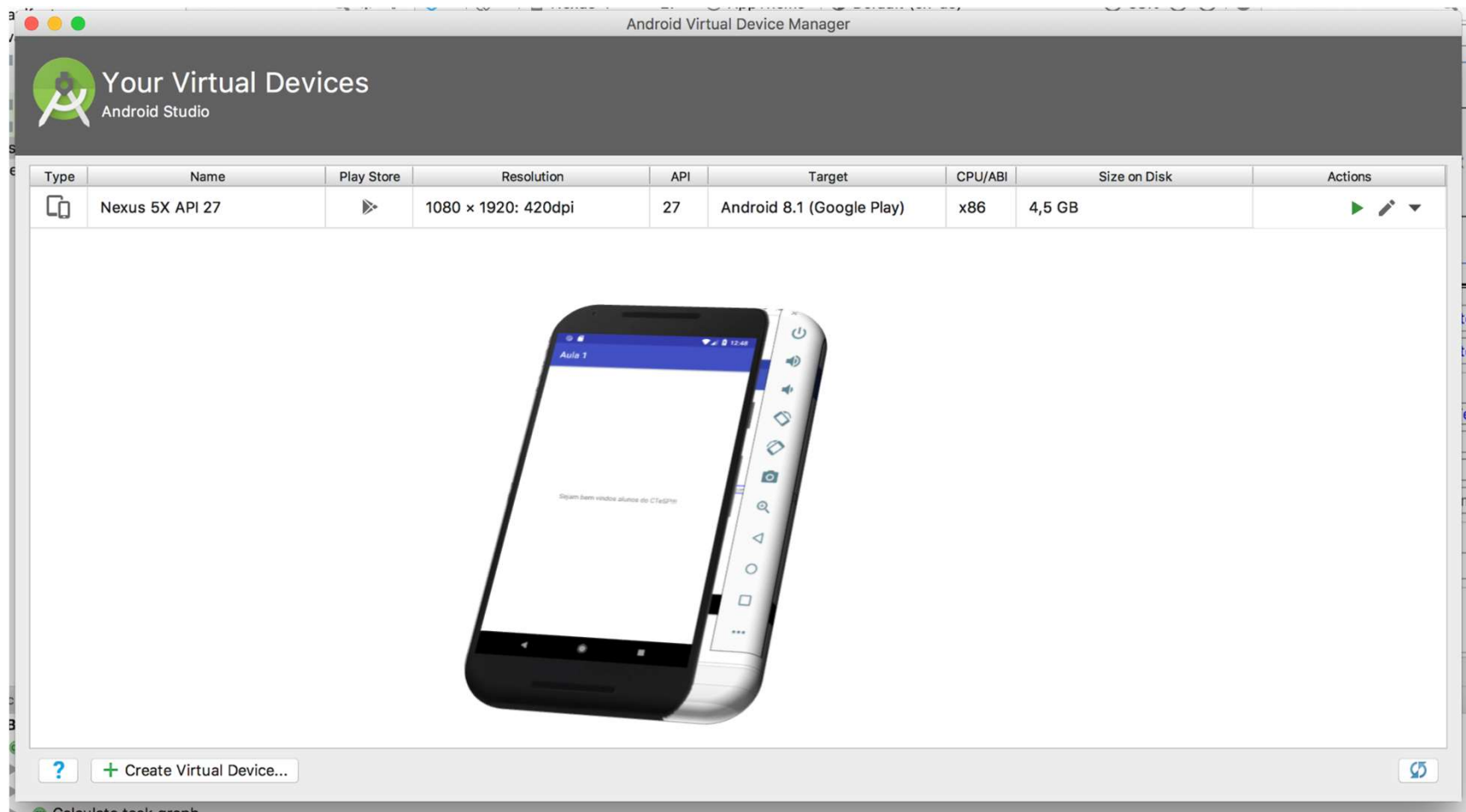
Android Studio

- AVD Manager
 - *Android Virtual Device*
 - Emulador para testar aplicações Android sem a necessidade de instalar a aplicação num dispositivo físico
 - Um AVD pode ser configurado para emular uma variedade de recursos de hardware:
 - Tamanho do ecrã
 - Memória
 - Câmara
 - GPS
 - Acelerómetro ...



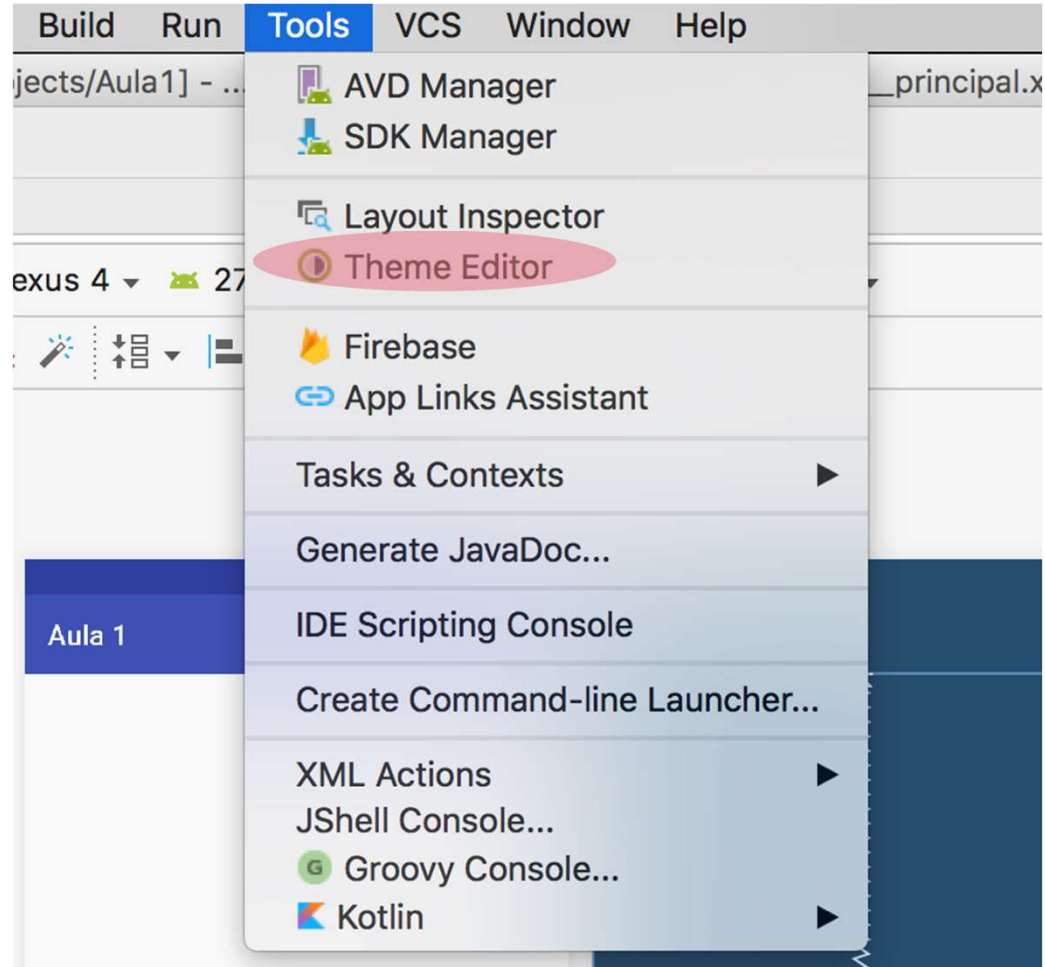
Android Studio

○ AVD Manager



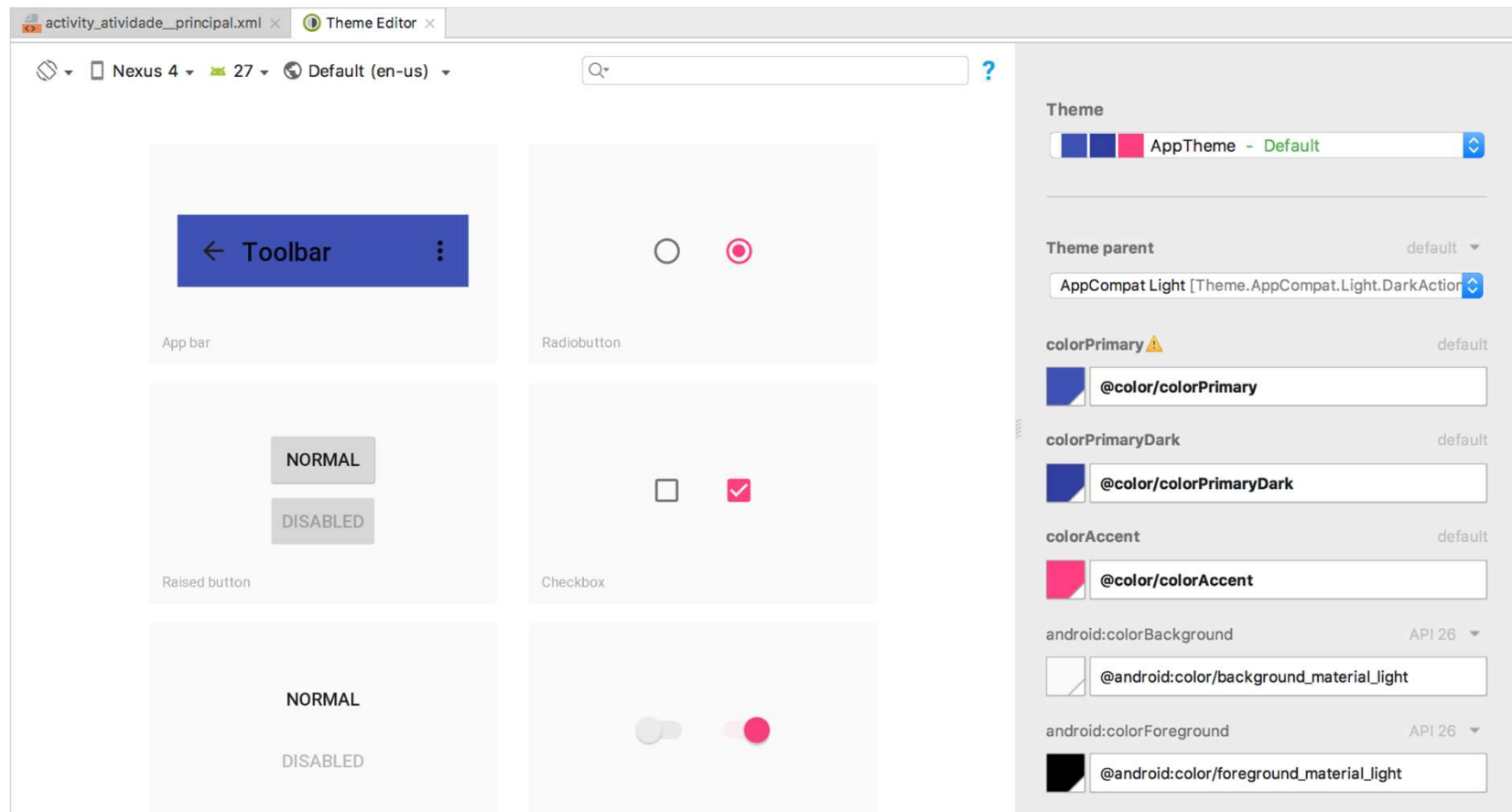
Android Studio

- Editor de Temas
 - Estilo de interface gráfica que determina a aparência gráfica da aplicação
 - O estilo é associado no ficheiro de manifesto do projeto

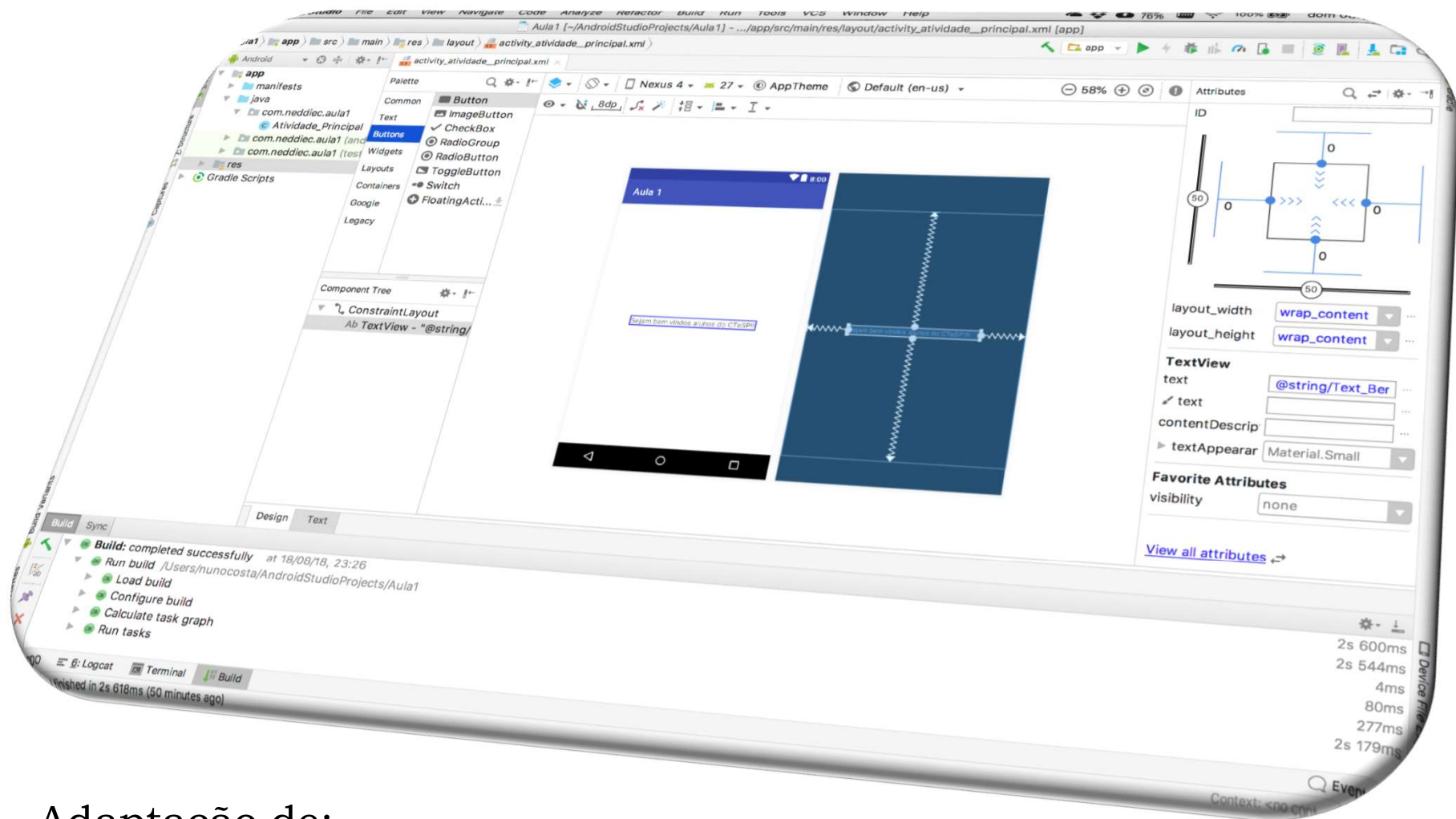


Android Studio

○ Editor de Temas



Vamos explorar o Android Studio



Adaptação de:
Nuno Costa

