


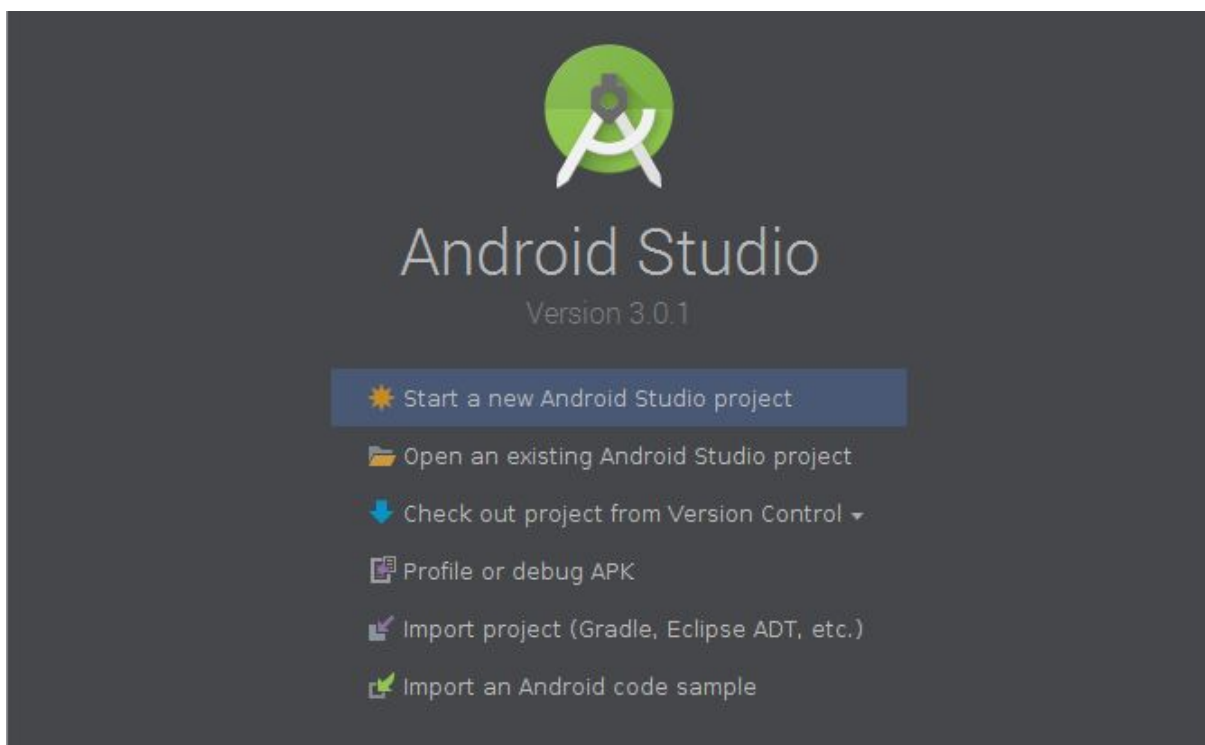
Tutorial Inicialização do Ambiente de Desenvolvimento e Primeiro Aplicativo

Para desenvolvimento de aplicações Web baseadas em Android, existem algumas opções à escolha do desenvolvedor, de acordo com o grau de complexidade pretendido pelo aplicativo. Uma boa maneira de começar a estudar as características desse ambiente é utilizar a ferramenta disponibilizada pelo próprio fabricante do sistema operacional, o ambiente de desenvolvimento (IDE) Android Studio.

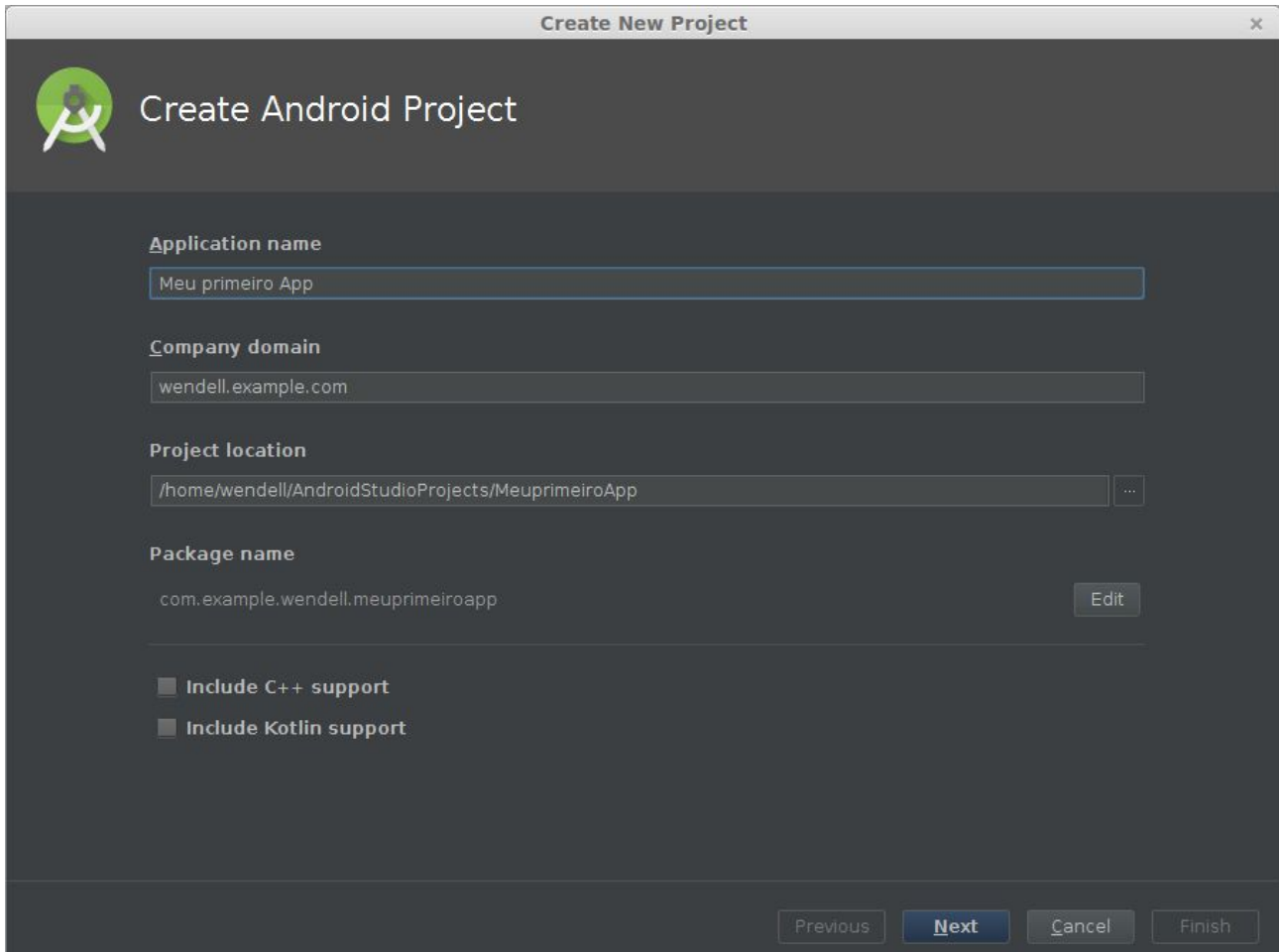
O Android Studio é uma solução bastante completa, fornecendo um bom editor de código, com capacidade de autocomplemento de código, auto formatação e outras facilidades. Também inclui um gestor de projetos e ferramentas de automação de geração de pacotes e até mesmo integração com plataformas de controle de versão. Também há ferramentas de depuração e até mesmo um emulador de dispositivos, onde é possível pré-visualizar o funcionamento do aplicativo.

Vamos começar abrindo o ambiente para conhecê-lo. Clique no ícone  do Android Studio para iniciá-lo.

Ele deve abrir a tela de boas vindas, onde encontraremos uma lista dos projetos disponíveis, caso algum exista e uma lista de opções para criar ou abrir projetos, além de outras tarefas comuns.

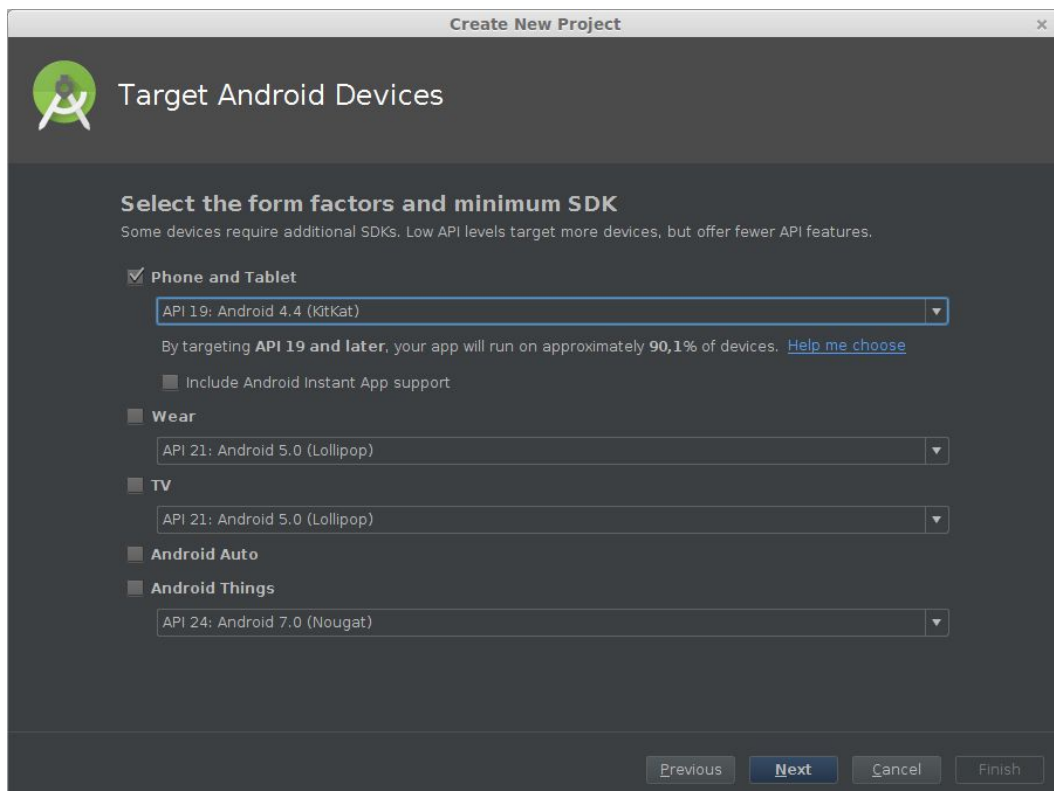


Vamos escolher a primeira opção, *Start a new Android Studio project*, que é para criar um novo projeto. Na próxima tela, deve-se indicar algumas informações básicas do projeto: o nome do aplicativo, o domínio da companhia e o local no diretório de arquivos onde queremos salvar os arquivos do projeto. Para o nome, use “Meu primeiro App”. No campo *Company Domain*, deixe o valor que vem indicado, por enquanto. Mais tarde, caso decida registrar uma conta de desenvolvedor, será possível usar seu próprio domínio. O domínio é um identificador único usado para marcar um desenvolvedor e seus aplicativos dentro da loja do Android. Também é usado para identificação internamente pelo sistema. Repare que o nome do pacote é o domínio acrescido do nome do aplicativo.

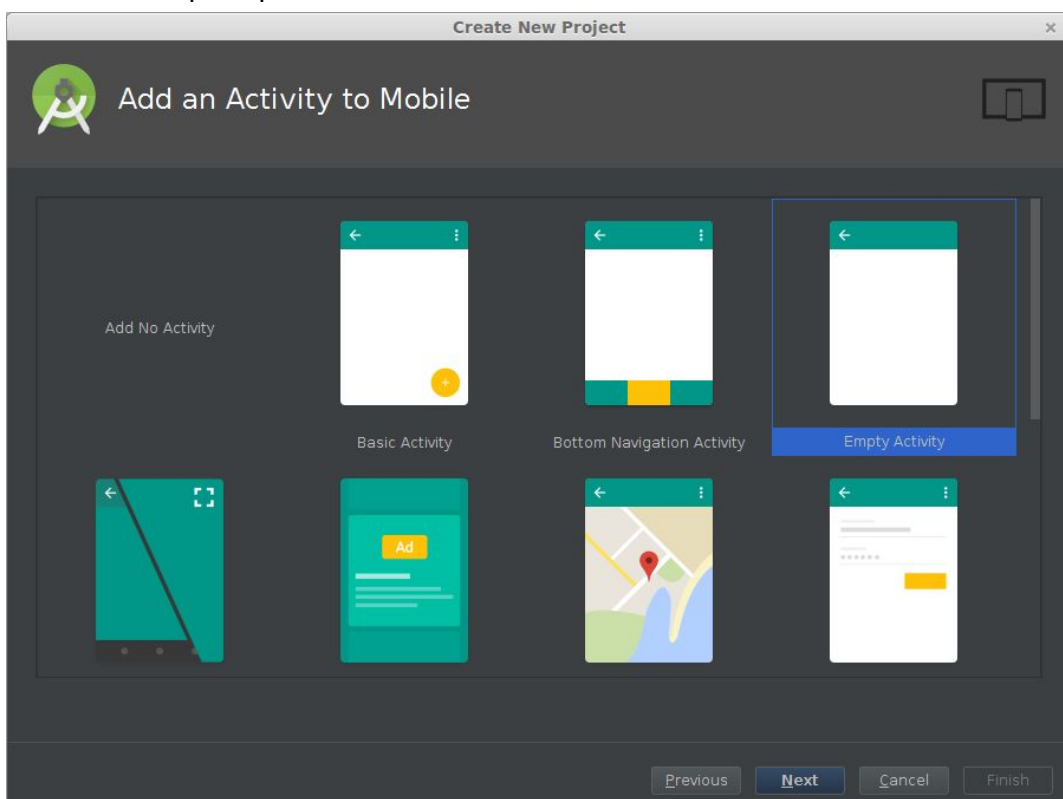


The screenshot shows the 'Create New Project' dialog in Android Studio. The dialog is titled 'Create New Project' and features the Android logo. The main heading is 'Create Android Project'. Below this, there are four input fields: 'Application name' (containing 'Meu primeiro App'), 'Company domain' (containing 'wendell.example.com'), 'Project location' (containing '/home/wendell/AndroidStudioProjects/MeuprimeiroApp'), and 'Package name' (containing 'com.example.wendell.meuprimeiroapp'). There are also two checkboxes: 'Include C++ support' and 'Include Kotlin support', both of which are unchecked. At the bottom of the dialog, there are four buttons: 'Previous', 'Next', 'Cancel', and 'Finish'.

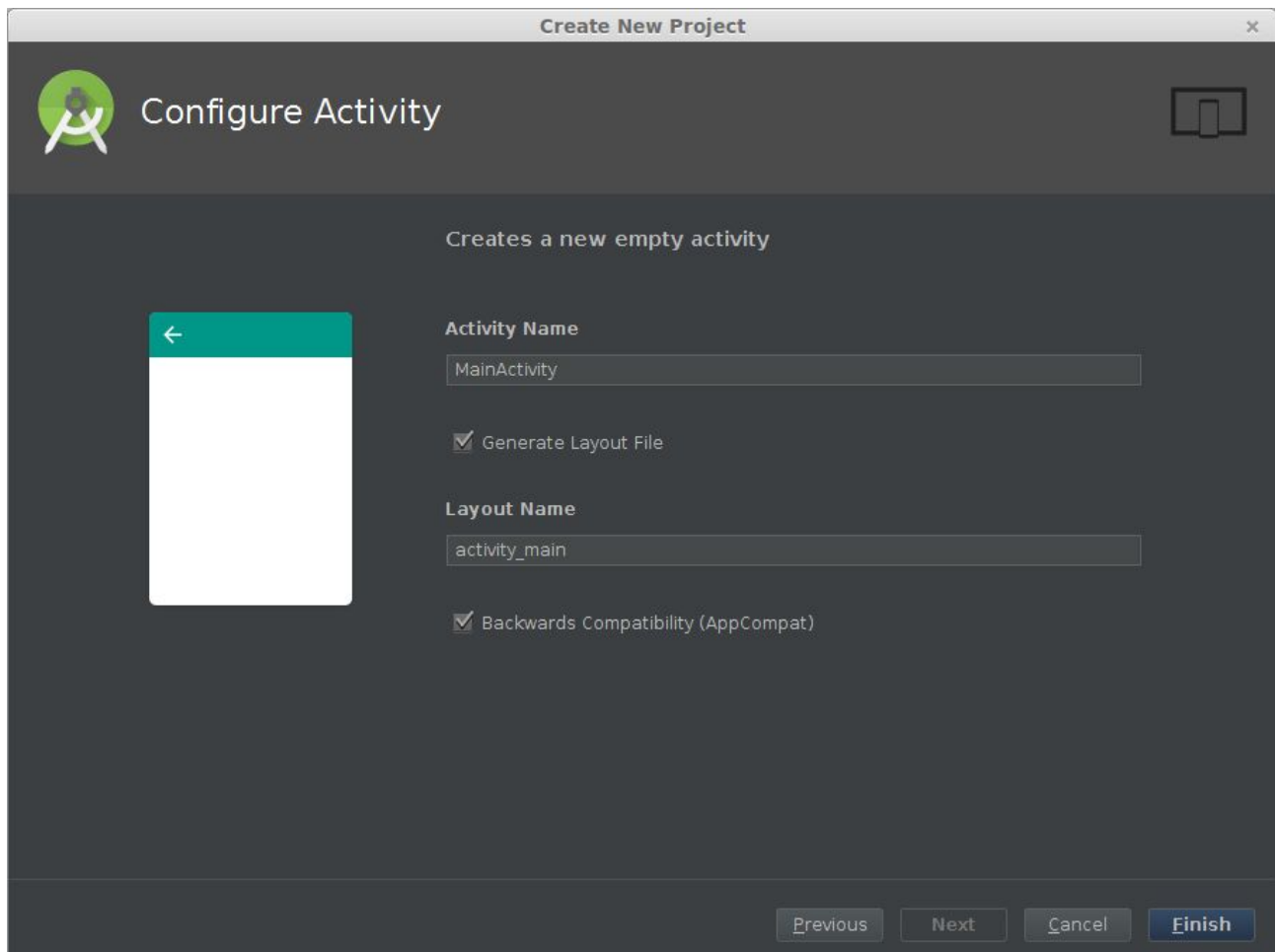
Agora, clique em *Next*, para avançarmos a próxima parte do processo de criação de projeto. Nesta tela, vamos identificar qual nível de API (*Application Programming Interface*) desejamos para criar nosso app, assim como a plataforma alvo. Vamos marcar a opção *Phone and Tablet* e usar o nível de API 19 *Android 4.4 Kit Kat*. Lembrando que a escolha do nível de API afeta o número de dispositivos capazes de executar o app e os recursos disponíveis para utilizar. Repare que o ambiente fornece uma estimativa de quantos dispositivos podem ser alcançados pela seu app. Para o nível 19, este número é de aproximadamente 90%, no momento da criação deste tutorial.



Na próxima tela, vamos definir a aparência da primeira tela de nosso app. Em um app Android, as ações que um usuário pode tomar, incluindo aí sua apresentação (interface), são chamadas de *Activity*. Mais a frente, estudaremos profundamente como programar e ligar diferentes Activities, para criar um fluxo de ações coeso. Por agora, vamos criar uma atividade vazia, o tipo mais básico de atividade e que dá mais liberdade para programarmos o comportamento desejado. Então, clique no quadro que diz *Empty Activity*. Esta será a primeira atividade da aplicação, por isso ela é chamada de atividade principal.

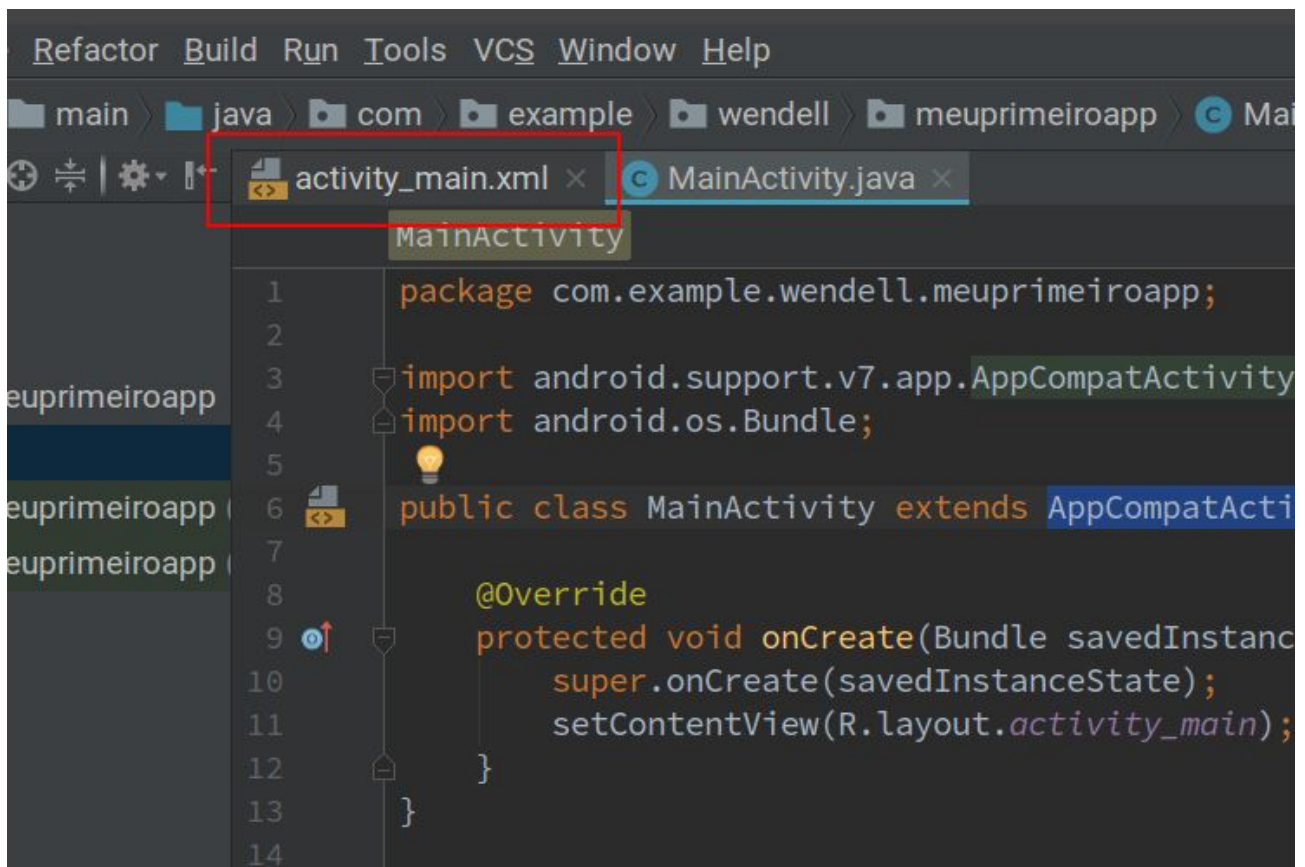


Na próxima tela, podemos definir o nome da nossa atividade e também o nome do arquivo de layout associado a ela. No arquivo de layout é que vamos definir a interface da atividade. Podemos usar os valores padrão que a IDE oferece. Clique em *Finish* e aguarde a IDE terminar de criar a estrutura do projeto.

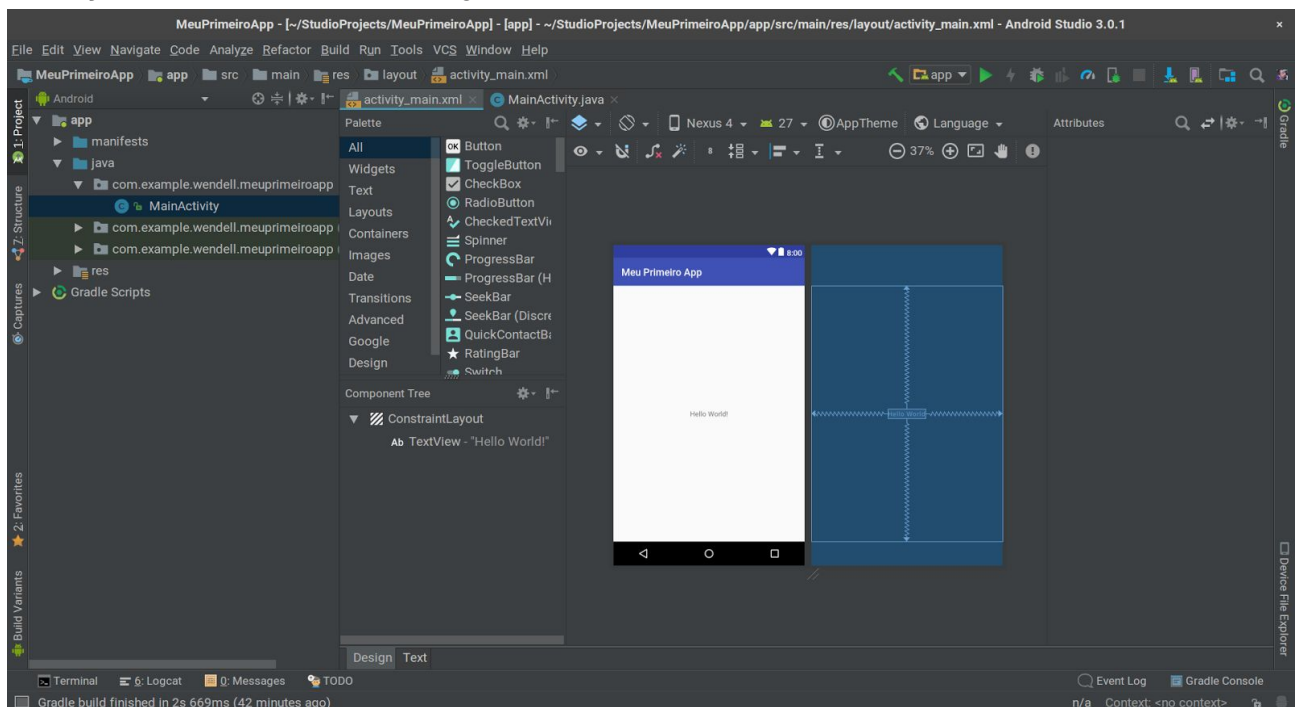


Neste ponto, estamos prontos para começar a programar nosso projeto! Repare nos elementos da interface do ambiente de desenvolvimento. Tente reconhecer os elementos que foram discutidos na última aula. Repare no código da atividade principal. Podemos ver a declaração do pacote e os imports das bibliotecas que são necessárias para a atividade. Podemos ver que a classe *MainActivity* foi declarada juntamente com o comando *extends*, que significa uma herança a partir da classe *AppCompatActivity*. Esta classe é a classe de atividade básica do SDK do Android. Ela também traz algumas funções para facilitar a compatibilidade entre diferentes versões do sistema, respeitando a versão mínima declarada no início do projeto.

Vamos observar agora o editor da interface da atividade. Clique na guia que tem o nome do arquivo de layout, *activity_main.xml*, do lado esquerdo da guia com o nome da atividade, conforme indicado na figura a seguir.

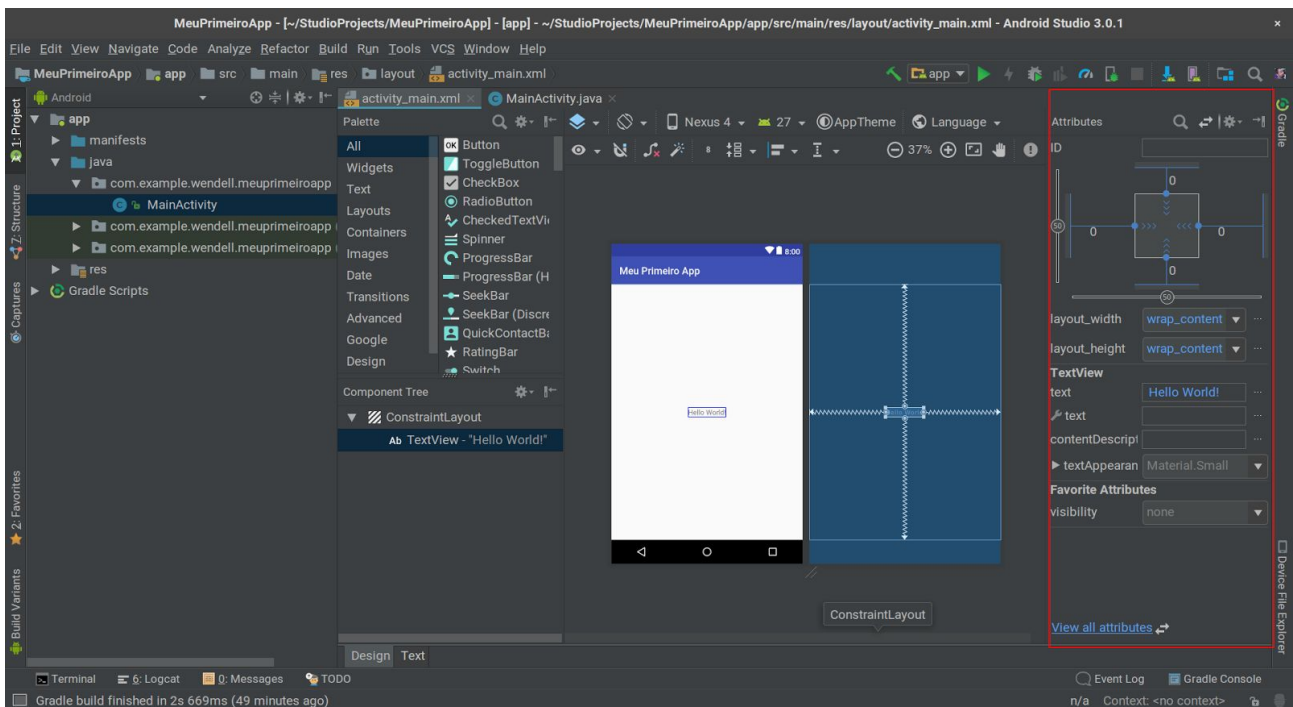


Esta ação nos levará à tela de design de interface:

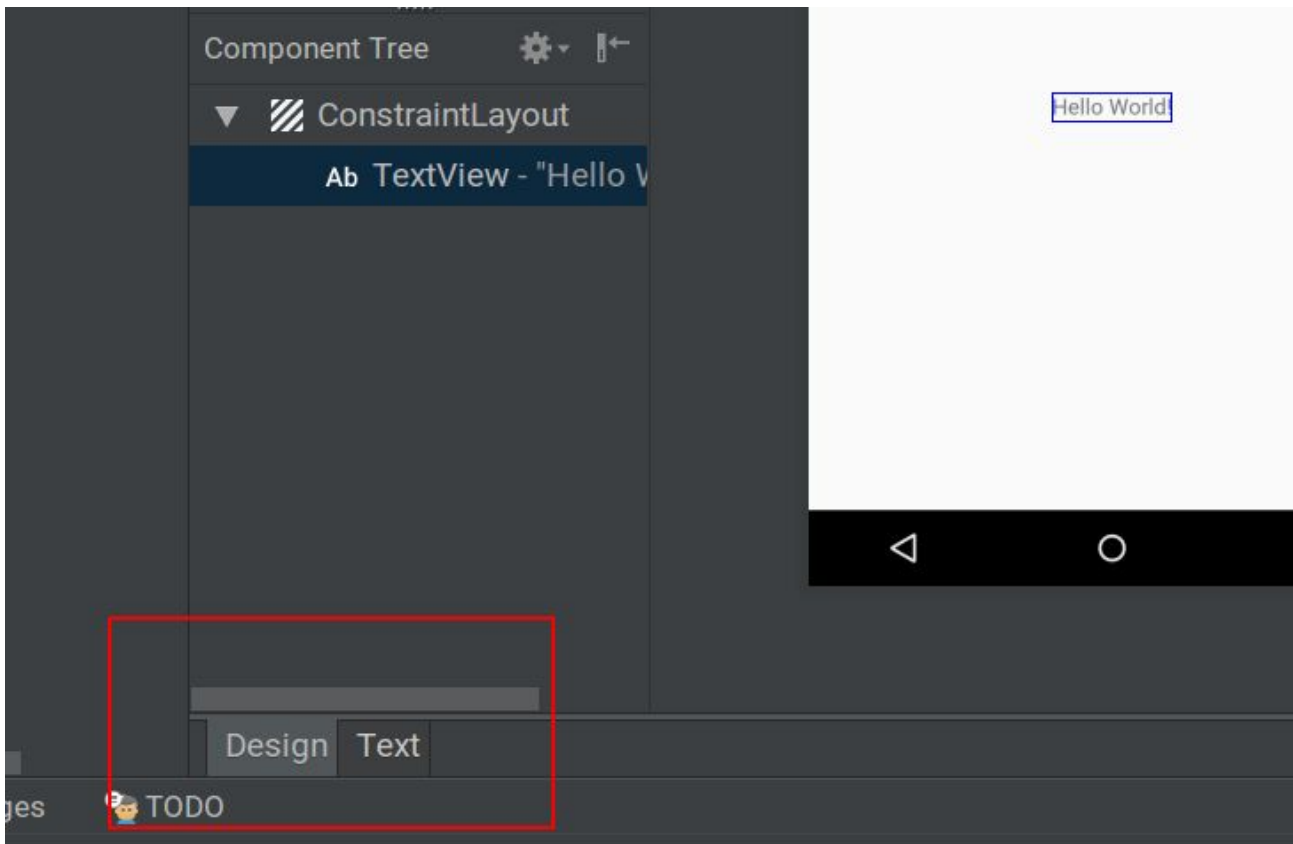


Repare que no centro do editor, há duas visualizações, uma mostrando um preview de como a interface se parecerá e outra mostrando o posicionamento dos elementos dentro da interface. À esquerda do editor, temos uma caixa de ferramentas com os vários elementos de interface que podemos usar para construir nosso app.

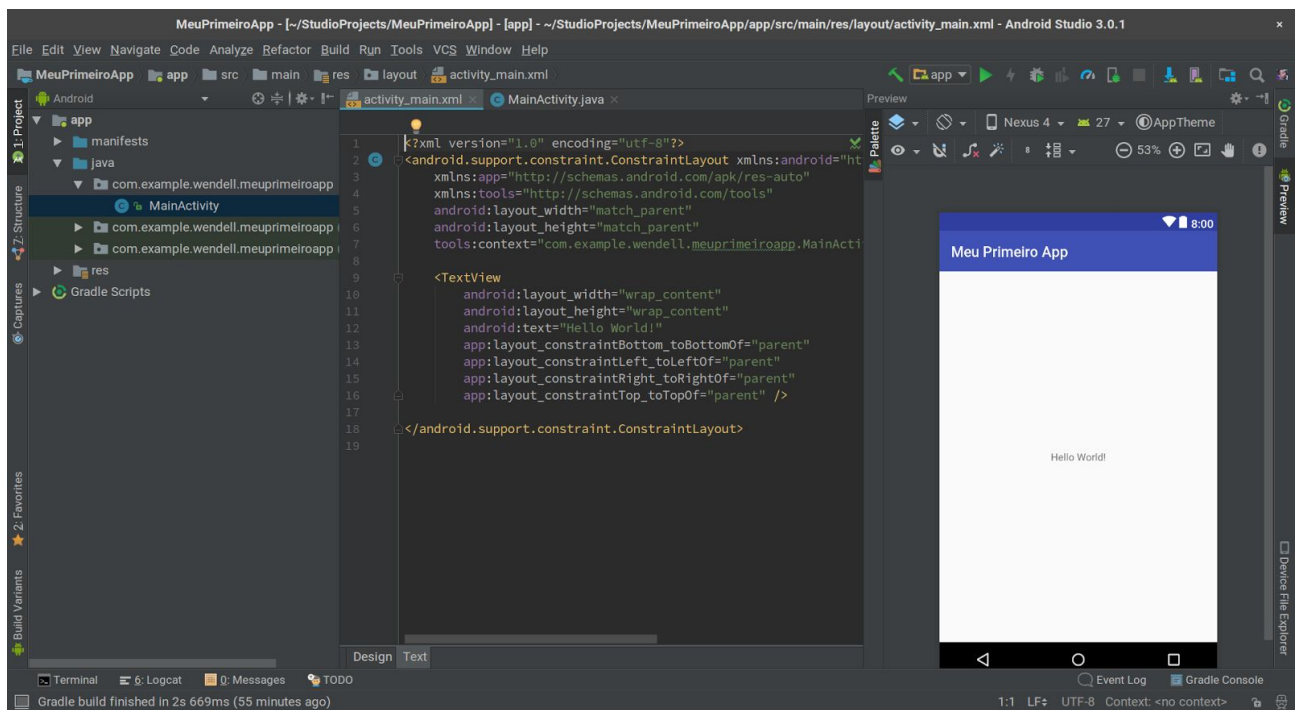
Repare agora na etiqueta de texto bem no centro da aplicação. Ao clicá-la, repare que no lado direito aparecem as propriedades do elemento, onde podemos editá-las.




Agora, clique na palavra *Text*, que fica bem ao lado da palavra *Design*, à esquerda e abaixo da janela de edição de interface.

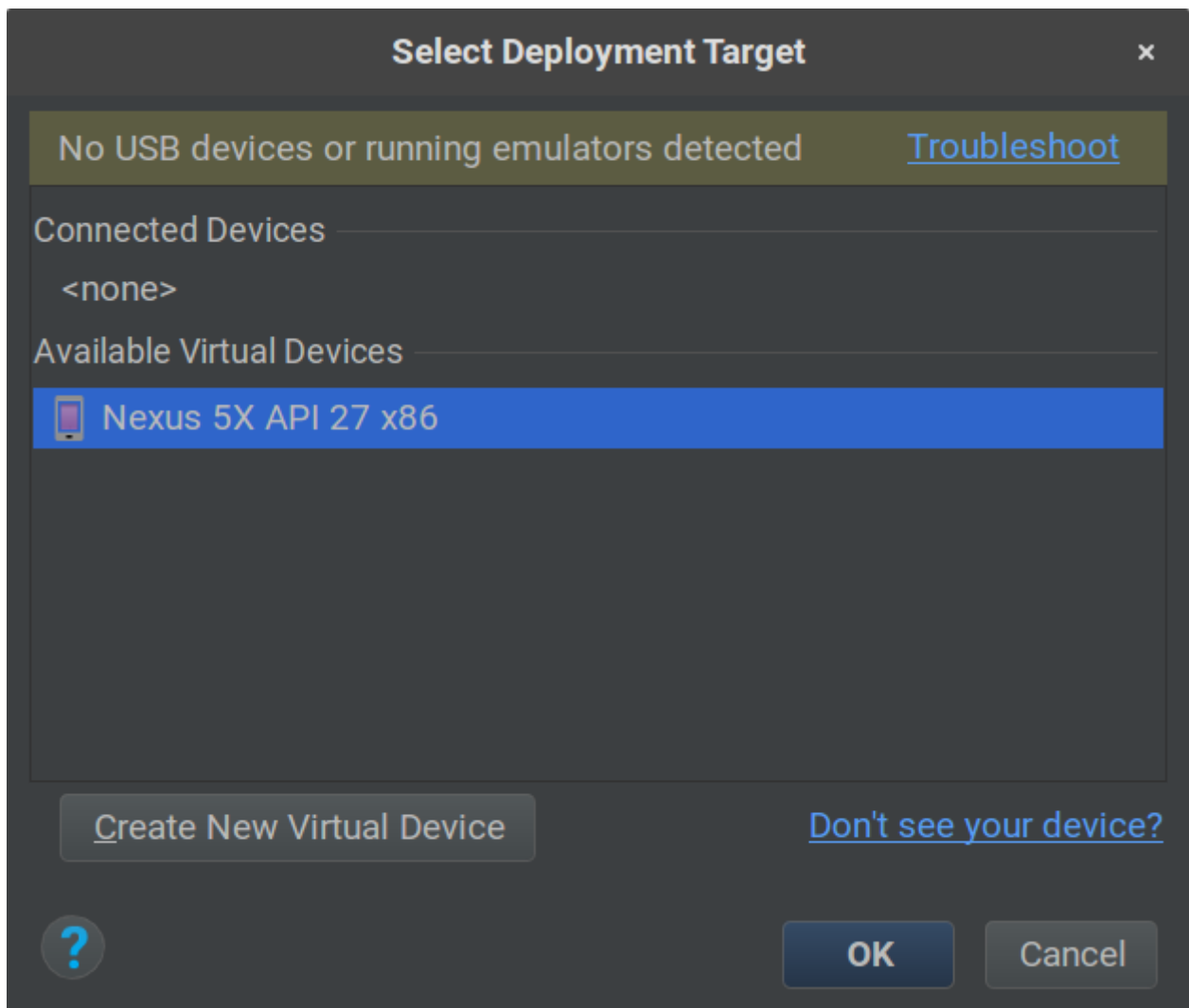


Agora é exibido o código fonte na linguagem XML, onde os elementos de interface e suas propriedades são definidos. Podemos construir nossas interfaces tanto pelo editor de texto quanto pelo editor visual. Experimente ambas para decidir qual das duas é mais confortável para você.

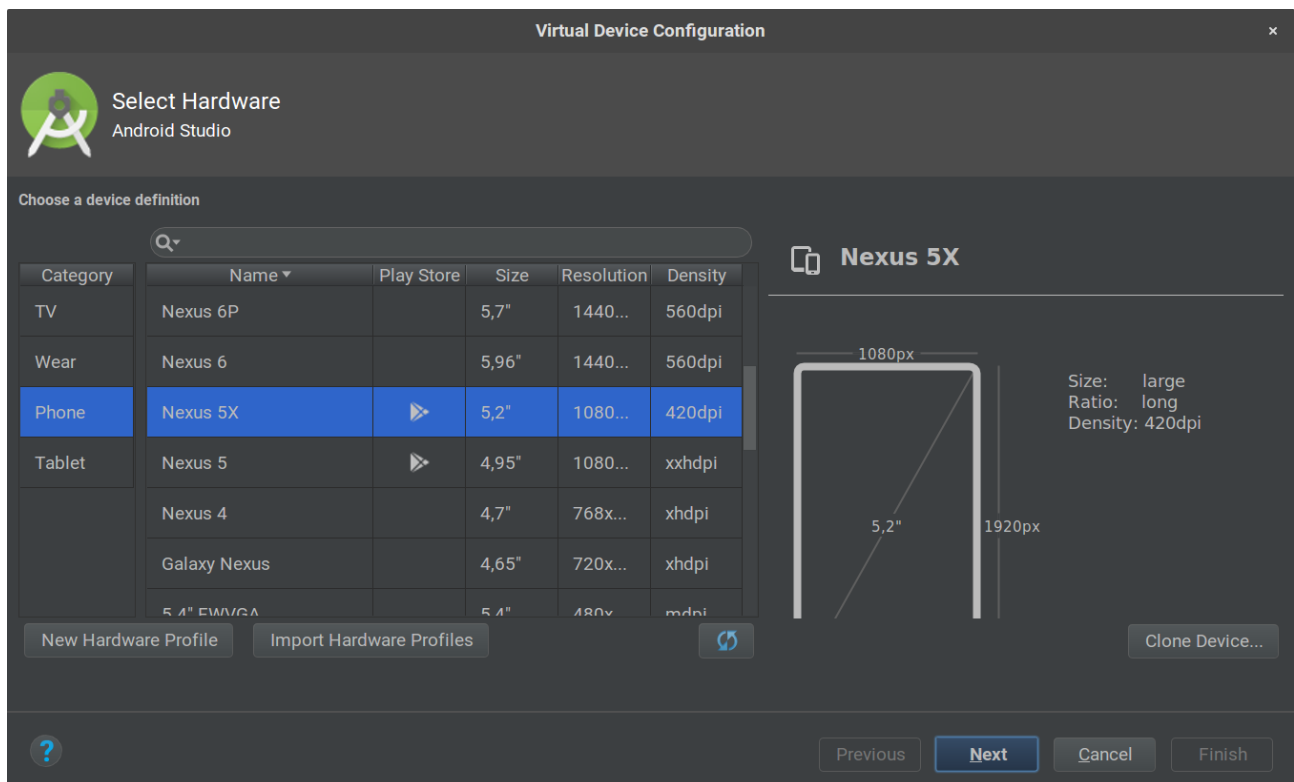


Agora vamos compilar nosso app e executá-lo no emulador. Clique no menu *Run > Run App(s)* ou

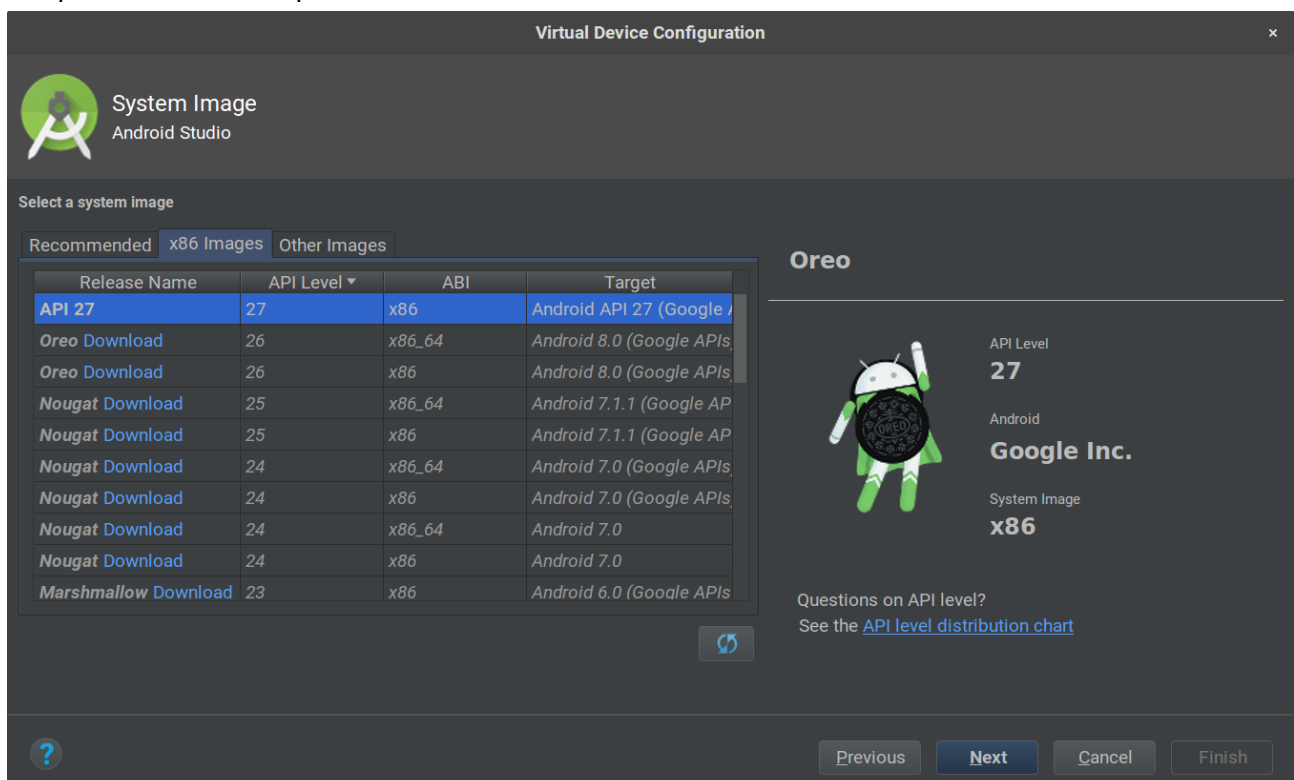
use o atalho de teclado *Shift+F10*. Ou ainda, clique no botão , na barra de ferramentas para lançar o app. Após o processo de compilação, aparecerá um diálogo onde selecionaremos o destino do deploy. Neste diálogo, aparecerão os emuladores disponíveis. Se um dispositivo estiver conectado em uma porta USB e estiver disponível para conexão, ele também será mostrado nesta tela.



Caso não exista nenhum dispositivo disponível no campo Available Virtual Devices, teremos que criá-lo antes de poder rodar a aplicação. Caso já exista um dispositivo virtual configurado, selecione-o e clique em Ok e pule para a próxima etapa. Caso contrário, clique no botão *Create New Virtual Device*. Na próxima tela, temos várias opções de dispositivos que podem ser emulados pela plataforma.

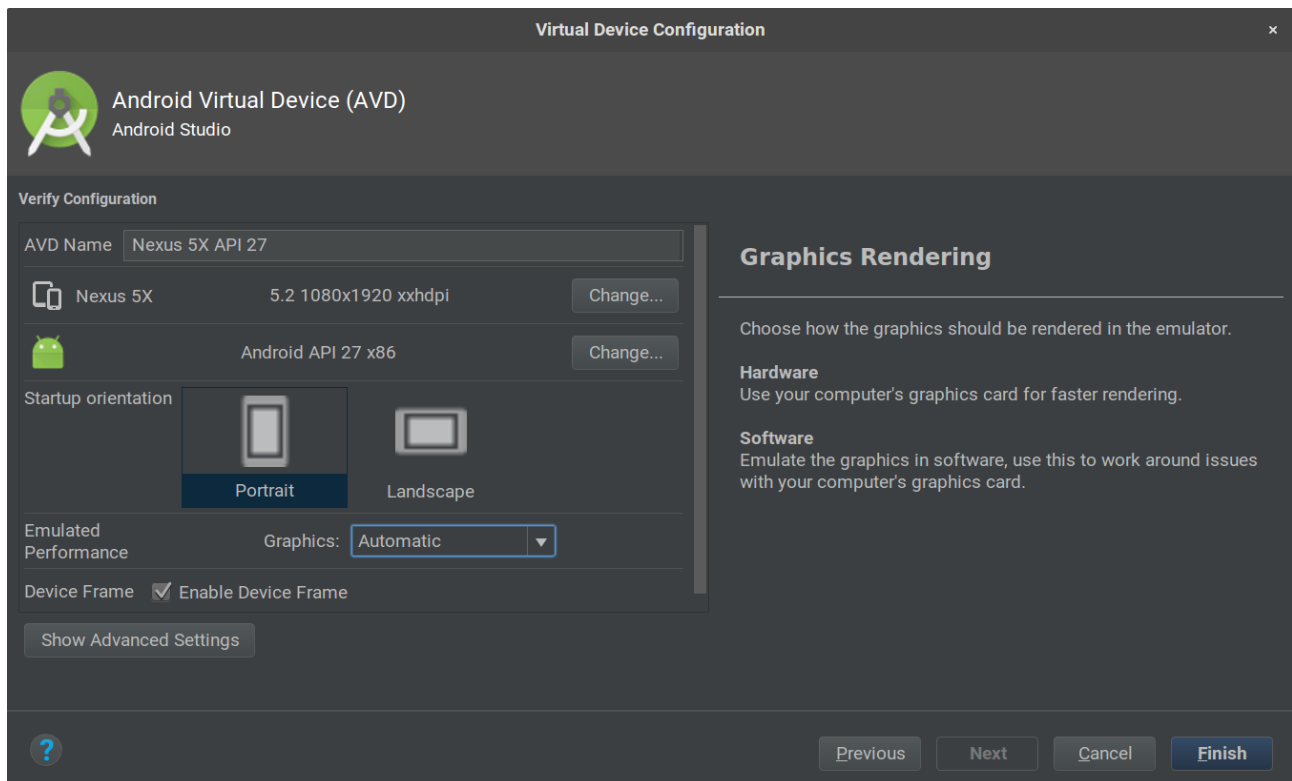


Na categoria, escolha *Phone*, e para o dispositivo, escolha Nexus 5X. Clique em *Next*. Agora, precisamos escolher uma imagem de sistema para emular. Se ela não estiver disponível na máquina, será solicitado seu download. Escolha uma imagem da categoria x86, que será compatível com o computador.



Após escolher a imagem, clique em *Next*. Na próxima tela são mostradas algumas propriedades que podem ser editadas, mas usaremos as opções padrão por agora. Clique em *Finish* para

encerrar a configuração e aguarde o dispositivo ser criado. Após a criação do emulador, execute novamente seu app, clicando no botão ou no menu *Run > Run 'app'*.



Com um emulador configurado e pronto para rodar, surgirá uma janela com a representação do dispositivo virtual. O app será lançado automaticamente. O dispositivo virtual simula um telefone com Android e tem quase todas as funções de um dispositivo real.

Pronto! Com isso, construímos e fizemos a emulação de nosso primeiro aplicativo.

Exercícios

Explore a ferramenta de projeto de interface. Tente acrescentar novos elementos, alterar algumas propriedades e observar como estas alterações se refletem no visualizador. Após alterar, execute novamente a aplicação e observe as mudanças.

Caso você tenha um telefone com Android e um cabo USB disponível, tente rodar a aplicação nele.

Android Emulator - Nexus_5X_API_27_x86:5554

