

05

## Faça o que eu fiz na aula

A primeira parte para começarmos nossa implementação do Capacitor, é criar o projeto Ionic 4.

Dentro do seu terminal, crie um projeto utilizando o comando `ionic start cozinhando-casa`. Selecione a opção `blank` (que é a primeira) na hora da escolha do template.

```
Marias-MBP:Desktop marialuizaforanagieri$ ionic start cozinhando-casa
```

**Let's pick the perfect starter template! 🎉**

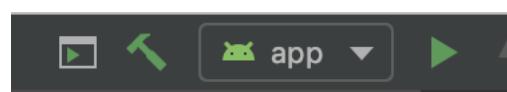
Starter templates are ready-to-go Ionic apps that come packed with everything you need to build your app. To bypass this prompt next time, supply `template`, the second argument to `ionic start`.

```
? Starter template: (Use arrow keys)
❯ blank      | A blank starter project
  sidemenu  | A starting project with a side menu with navigation in the content
area
  tabs       | A starting project with a simple tabbed interface
```

Na sequência, use o comando `npm i --save @capacitor/cli @capacitor/core` para instalar as dependências do Capacitor no projeto. E então, quando o processo de instalação é concluído, utilize o comando `npx cap init` para inicializar o Capacitor no projeto.

A primeira coisa que a interface pede é o nome do projeto e a segunda é o ID do pacote do projeto. Na sequência, utilize o comando `ionic build` para gerar a pasta `www`. E então será possível adicionar as plataformas com os comandos `npx cap add android` e `npx cap add ios`.

Para rodar o projeto nos dispositivos, é necessário abrir o projeto na IDE de cada plataforma. No caso do Android, use o comando `npx cap open android`. Isso abre o Android Studio. No qual você pode clicar no botão de “run” — representado por uma seta verde que fica na barra superior à direita —, e então selecione um dispositivo para rodar o projeto.



No iOS, utilize o comando `npx cap open ios`. Isso abre o xcode. Selecione um emulador na barra superior direita da IDE e, na sequência, clique no botão para rodar, que fica à esquerda do botão utilizado para escolher o emulador.



