

## Material Design

### Transcrição

Vamos desenvolver o nosso aplicativo **Garçonapp**. O objetivo é implementar o código HTML, CSS e Java Script, que criará as funcionalidades do nosso app. Se você quiser focar apenas na aula de Cordova, poderá saltar este capítulo.

Vamos implementar o app. Começaremos com uma discussão sobre design. Usaremos o **Material Design**, criado pelo Google. O aplicativo Garçonapp será usado apenas pela equipe do restaurante, em um dispositivo Android comprado pelo proprietário. Então, usaremos um design do Google, que se integra bem ao Android, parece com um app normal.

O *Material Design* foi criado pelo Google como uma metodologia de design que pode ser aplicado em diversos cenários, desde aplicativos nativos e Web. Você irá acessar o [design do Google \(https://design.google.com/\)](https://design.google.com/). No site, encontraremos alguns exemplos de como funcionam os componentes, por exemplo, como os botões (*Buttons*), modelos que já vimos usados em apps Android, o uso de cores fortes e um sombreado. Eles se inspiraram na ideia de folhas de papel sobrepostas. Nesta parte, apenas encontraremos a descrição dos modelos. Também encontraremos uma explicação de como funciona uma lista no *Material Design*, os espaçamentos usados.

É possível implementar estes componentes no app? Sim. Existem *frameworks* que seguem a linha do *Material Design*, já implementando em HTML, CSS e JS. Podemos utilizá-los dentro do Cordova. O mais famoso deles é o [Polymer Project \(https://elements.polymer-project.org/\)](https://elements.polymer-project.org/). Trata-se de um projeto ainda maior que o Material Design, porque tem todos os componentes. Porém, eles foram além e implementaram os componentes do Material Design em HTML e CSS. Se clicarmos em **Paper Elements**, veremos todos os componentes que são do Material Design. Por exemplo, se quisermos trabalhar com botões, usaremos uma *tag* nova chamada `paper-button`. Se clicarmos em *Demo*, veremos como eles são visualizados.

Encontraremos lá diversos elementos. Nós não usaremos o *Polymer* no projeto. Mas conheceremos outras soluções, como o *Material Design Lite*, um outro *framework* do Google, feito com HTML puro. Temos alguns exemplos de criações que podemos gerar trabalhando com algumas classes do *Material Design*. Temos várias opções de [componentes \(http://www.getmdl.io/components/index.html\)](http://www.getmdl.io/components/index.html): *Menus*, *Sliders*, *Toggles*, *Tables*, *Text Fields* e *Tooltips*.

O *framework* que iremos utilizar é o [Materialize \(http://materializecss.com/\)](http://materializecss.com/), que não é o oficial do Google, mas é super fácil de usar. Para utilizá-lo, basta fazer o download no site e poderemos trabalhar com os componentes, como botões. No caso, eles serão bastantes semelhante ao do Polymer, mas criados apenas com HTML. Se queremos criar um Card, temos um layout e vemos que trabalhamos com algumas classe ( `card-content` , `card-title` , `card-action` ...). Teremos as opções de outros componentes também. Os formulários disponíveis, facilmente já encontramos em algum app, que quando clicamos nos campos a serem os `labels` se movem. Encontramos todos esse componentes já feitos pelo projeto *Materialize*, basta usarmos o HTML correto, com as classes corretas. Por isso, iremos implementá-lo, porque ele é um mecanismo pronto, bastante simples e não teremos dificuldades. Para o nosso aplicativo será suficiente.

No fim, queremos criar um app simples, que lista todas a opções de bolos e bebidas do cardápio. A garçonete irá clicar sobre o botão de cada item e o app irá marcá-los. Quando o pedido for completado, ela irá finalizar o pedido pressionando um botão da parte inferior. Com isso, o pedido irá chegar diretamente na cozinha. Tudo será feito utilizando o design do *Material Design*, com componentes do *Materialize*.