

Olá Codi!

Construindo seu primeiro aplicativo: HelloCodi

Agora que você configurou seu computador e dispositivo e aprendeu como o Designer e o Editor de Blocos funcionam, você está pronto para criar o aplicativo HelloCodi. Neste ponto, você deve ter o Designer ou Blocks Editor aberto em seu navegador e um dispositivo ou emulador Android/iOS conectado para teste durante a compilação. (Consulte as [Instruções de configuração do App Inventor 2](#) se você não tiver essas coisas em execução.) Escolha Iniciar um novo projeto no menu Projetos e nomeie-o como HelloCodi.

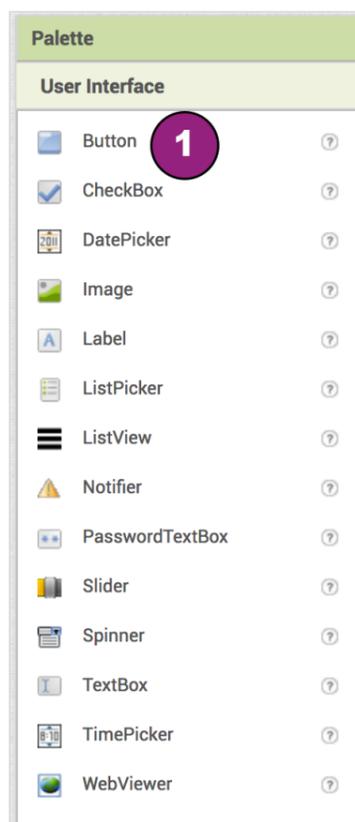
HelloCodi: Toque na abelha, ouça o zumbido da abelha!

HelloCodi é um aplicativo simples que você pode construir em muito pouco tempo. Você cria um botão que tem uma foto de Codi, a abelha, e depois programa o botão para que, quando for clicado, o telefone emita um zumbido.

Para construir o HelloCodi, você precisará de um arquivo de imagem de Codi, a abelha. Baixe esses arquivos para o seu computador clicando nos links a seguir. Para fazer o download: após clicar em um link, clique com o botão direito do mouse na imagem e selecione "Salvar como". Salve o arquivo na área de trabalho ou na pasta de downloads, ou em qualquer lugar onde você possa encontrá-lo facilmente mais tarde.

- Codi imagem: [codi.jpg](#) (clique com o botão direito do mouse e salvar)
- Som de abelha: [Bee-Sound.mp3](#) (clique com o botão direito do mouse e salvar)

Selecione componentes para projetar seu aplicativo

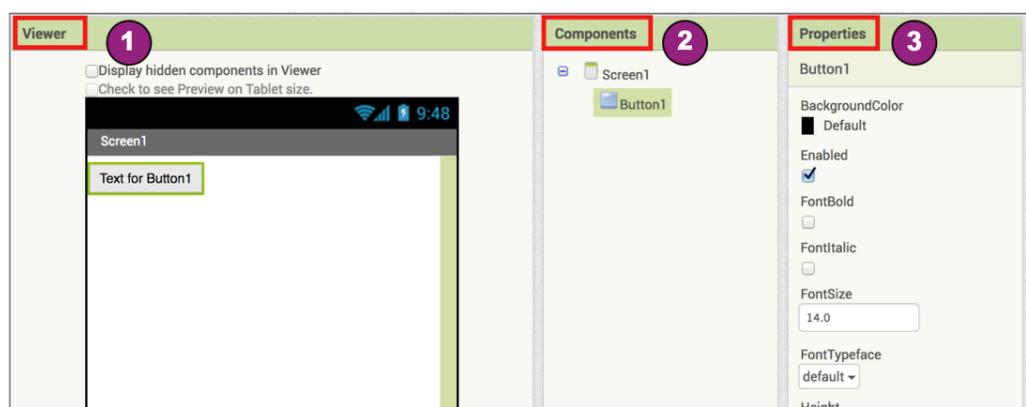


Os componentes do App Inventor estão localizados no lado esquerdo da *janela do designer*, sob o título **Paleta**. Componentes são os elementos básicos que você usa para criar aplicativos em um telefone Android ou iPhone. São como os ingredientes de uma receita. Alguns componentes são muito simples, como um componente **Label**, que apenas mostra o texto na tela, ou um componente **Button** (nº 1 à esquerda) que você toca para iniciar uma ação.

Outros componentes são mais elaborados: uma **tela** de desenho que pode conter imagens estáticas ou animações, um sensor de **acelerômetro** que funciona como um controle de Wii e detecta quando você move ou balança o telefone, componentes que enviam mensagens de texto, componentes que reproduzem música e vídeo, componentes que obtêm informações de sites da Web e assim por diante.

Para usar um componente em seu aplicativo, você precisa clicar e arrastá-lo para o visualizador no meio do **Designer**. Quando você adiciona um componente ao **Visualizador** (#1 abaixo), ele também aparecerá na lista de componentes no lado direito do Visualizador.

Os componentes (#2 abaixo) têm propriedades ajustáveis. Essas propriedades alteram a maneira como o componente aparece ou se comporta no aplicativo. Para visualizar e alterar as **Propriedades** de um componente (#3 abaixo), você deve primeiro selecionar o componente desejado em sua lista de componentes.

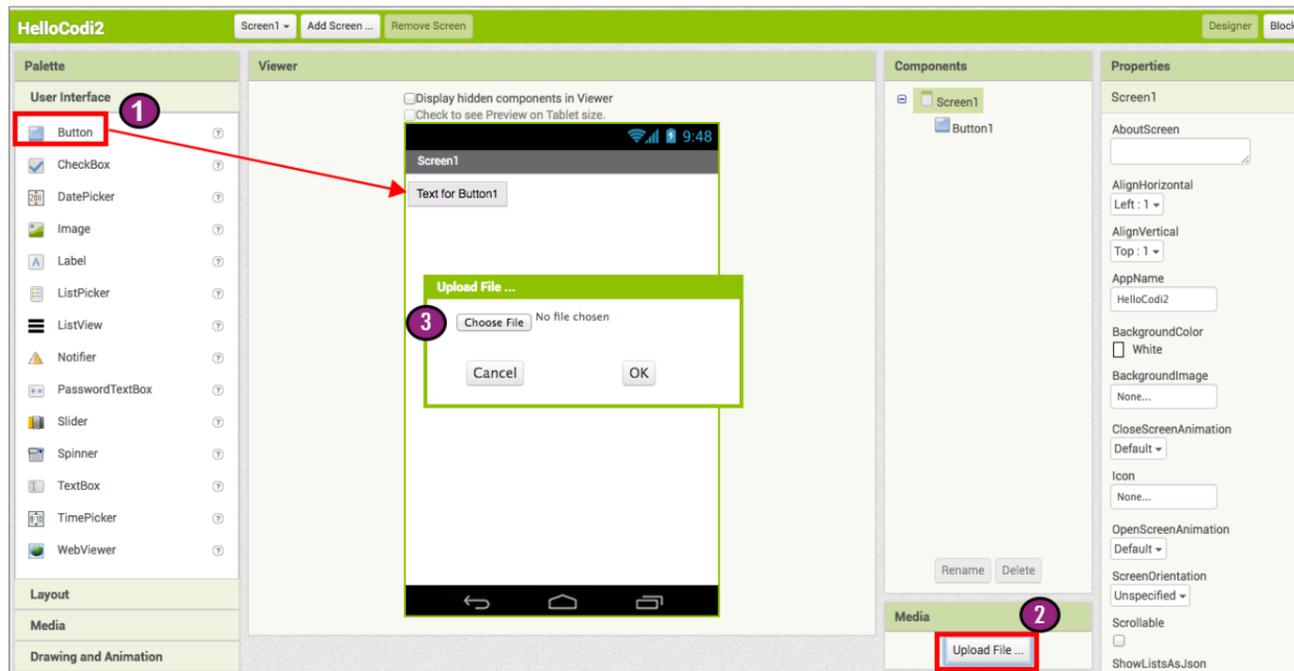


Etapas para selecionar componentes e definir propriedades

HelloCodi terá um componente **Button** que exibe a imagem da abelha que você baixou anteriormente. Para realizar isso:

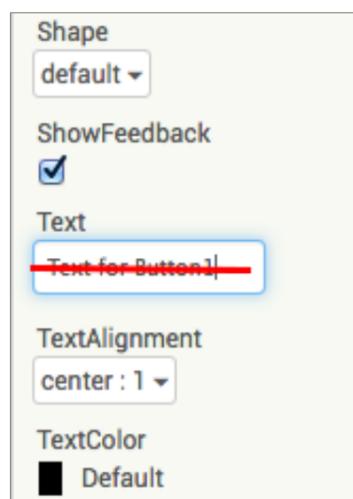
Etapa 1a . Na paleta **User Interface** , arraste e solte o componente **Button** na Tela1 (#1).

Passo 1b .Para dar ao botão a imagem da abelha, no painel **Propriedades** , em Imagem, clique no texto "Nenhum..." e clique em "Carregar Arquivo..." (#2). Uma janela aparecerá para permitir que você escolha o arquivo de imagem. Clique em "Procurar" e navegue até o local do arquivo *codi.jpg* que você baixou anteriormente (#3). Clique no arquivo *codi.jpg* , clique em "Abrir" e clique em "OK".



Etapa 2 . Altere a propriedade **Text** do Button :

Exclua "Text for Button1", deixando a propriedade text do Button em branco para que não haja escrita sobre a imagem da abelha.

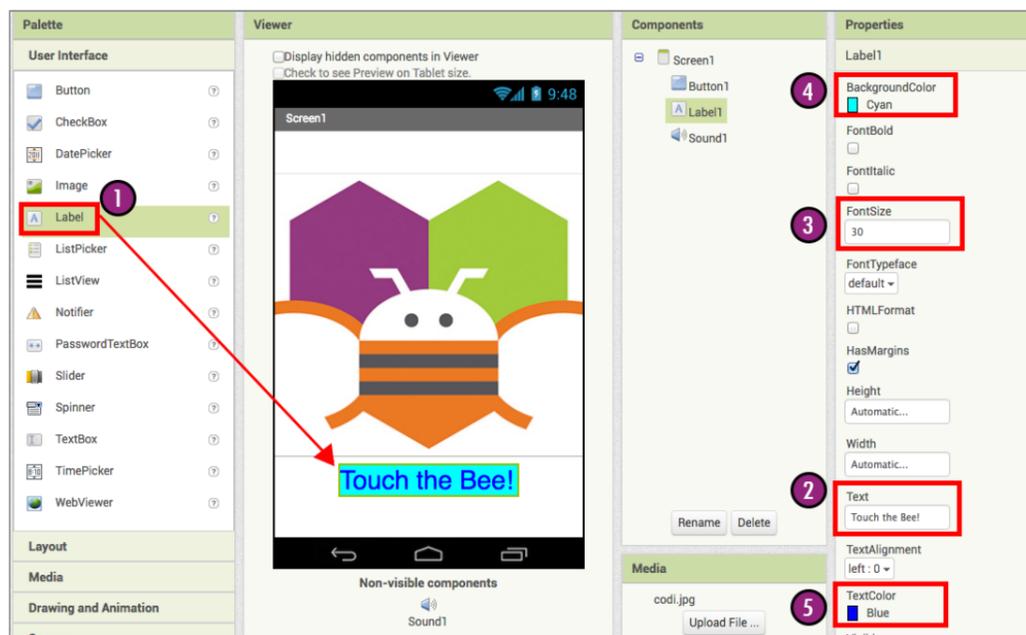


Etapa 3 . Na paleta **User Interface** , arraste e solte o componente **Label** no **Viewer** (#1) , colocando-o abaixo da imagem da abelha. Ele aparecerá em sua lista de componentes como **Label1** .

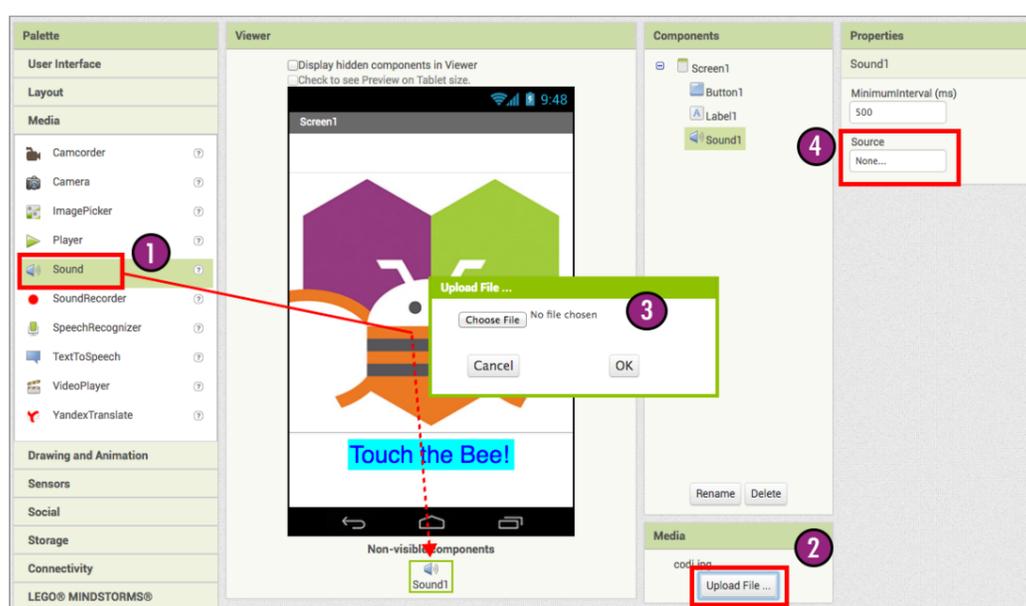
No painel **Propriedades** , altere o

- (2) **Propriedade de texto** de Label1 para ler "Touch the Bee". Você verá o texto mudar no Designer e no seu dispositivo.
- (3) **FontSize** para 30.
- (4) **BackgroundColor** de Label1 clicando na caixa. Você pode alterá-lo para qualquer cor que desejar.
- (5) **TextColor** para qualquer cor que desejar. (Observação: se BackgroundColor e TextColor forem iguais, você não poderá ler seu texto!)

Aqui, a cor de fundo é definida como aqua e a cor do texto é definida como azul.



Etapa 4 . Em Paleta, clique na gaveta de **mídia** e arraste um componente de **som** e coloque-o no visualizador (#1). Onde quer que você o solte, ele aparecerá na área na parte inferior do Visualizador marcado **Componentes não visíveis** . No painel Mídia, clique em Carregar arquivo... (#2) Navegue até o local do arquivo *Bee-Sound.mp3* que você baixou anteriormente e faça o upload para este projeto (#3). Sob o painel Propriedades, veja que a propriedade Source atualmente diz None... Clique na palavra None... para alterar Source do componente Sound1 para *Bee-Sound.mp3* (#4).



Programando com o Editor de Blocos

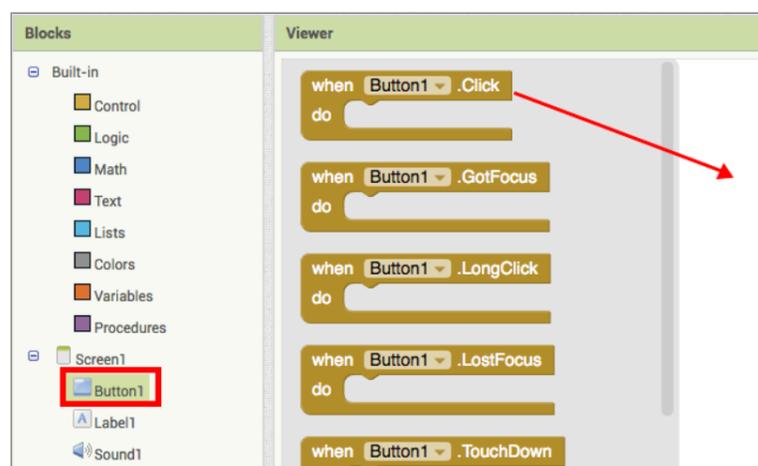
Até agora, você está organizando a tela e os componentes do seu aplicativo no *Designer* , que está em uma janela do navegador da web. Para começar a programar o comportamento do aplicativo, você precisa ir ao *Editor de Blocos* . Clique no botão Blocos no canto superior direito da tela para acessar o Editor de blocos.



Depois de ter o Editor de Blocos à sua frente, continue na próxima etapa para começar a programar seu aplicativo com blocos.

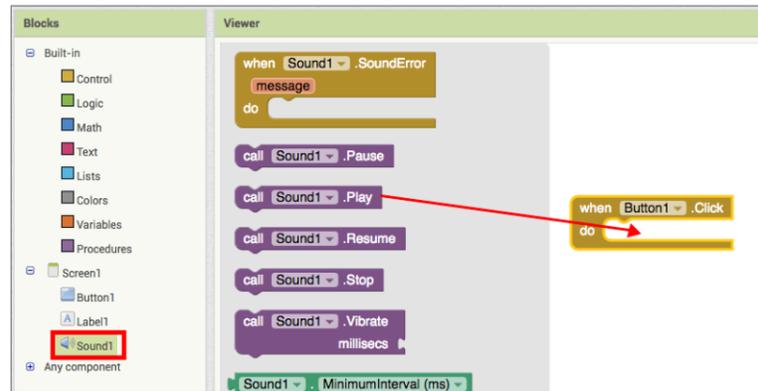
Tocando o som

Etapa 1 . No lado esquerdo do Editor de Blocos, clique na gaveta **Button1 para abri-la**. Arraste e solte o bloco `when Button1.Click` na área de trabalho (a área aberta à direita).



Esses blocos amarelos mostarda são chamados de blocos de **manipuladores de eventos** . Os blocos do manipulador de eventos especificam como o dispositivo móvel deve responder a determinados eventos: um botão foi pressionado, o telefone está sendo sacudido, o usuário está arrastando o dedo sobre uma tela etc. `quando Button1.Click` é um manipulador de eventos.

Passo 2a . Clique na gaveta **Sound1** e arraste o bloco `Sound1.Play` e conecte-o à seção "do" do bloco `when Button1.Click` . Os blocos se conectam como peças de quebra-cabeça e você pode ouvir um som de clique quando eles se conectam.



Os blocos roxos são chamados de blocos de **comando** , que são colocados no corpo dos manipuladores de eventos. Quando um manipulador de eventos é executado, ele executa a sequência de comandos em seu corpo. Um comando é um bloco que especifica uma ação a ser executada (por exemplo, tocar som) quando o evento (por exemplo, pressionar Button1) é acionado.

Seus blocos devem ficar assim neste momento:



Agora você pode ver que o **bloco de comando** está no **manipulador de eventos** . Este conjunto de blocos significa; "quando Button1 for clicado, Sound1 será reproduzido." O manipulador de eventos é como uma categoria de ação (por exemplo, um botão está sendo clicado) e o comando especifica o tipo de ação e os detalhes da ação (por exemplo, tocando um som).

Você pode ler mais sobre os blocos e como eles funcionam aqui: [Blocos integrados do App Inventor](#) .

Experimente no seu dispositivo ou com o emulador! Quando você pressiona o botão, você deve ouvir o zumbido da abelha. Parabéns, seu primeiro aplicativo está em execução!

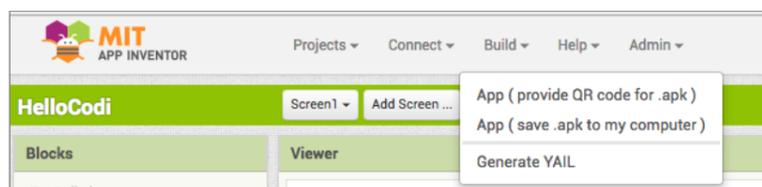
Observação: há um problema conhecido com o componente Som em alguns dispositivos. Se você vir um "Erro de SO" e o som não for reproduzido - ou estiver muito atrasado na reprodução, volte ao Designer e tente usar um componente Player (encontrado em Mídia) em vez do componente Som.

Empacotando seu aplicativo

Enquanto seu telefone/tablet/emulador foi conectado ao App Inventor, seu aplicativo foi executado em tempo real no seu dispositivo. Se você desconectar o dispositivo/emulador, o aplicativo desaparecerá. Você sempre pode fazê-lo retornar reconectando o dispositivo, mas para ter um aplicativo em execução sem estar conectado ao App Inventor, você deve "**empacotar**" o aplicativo para produzir um pacote de aplicativos (arquivo apk).

Por enquanto, esta última etapa, empacotar aplicativos, funciona apenas para telefones Android. Estamos trabalhando ativamente para trazer esse recurso para dispositivos iOS em breve!

Para "empacotar" o aplicativo para instalar no seu dispositivo ou enviar para outra pessoa, clique na guia **Construir** na parte superior da tela. Em Build, há duas opções disponíveis para você escolher:



1. **App (provide QR code):** You can generate a Barcode (a QR Code), which you can use to install the app on a mobile device that has a camera.

this barcode is only good for two hours. If you want to share your app with others via barcode over a longer period, you'll need to download the .apk file to your computer and use a third-party software to convert the file into a barcode. More information can be found [here](#).

2. **App (save to my computer):** You can download the app to your computer as an apk file, which you can distribute and share as you like by manually installing it on other devices (sometimes called "[side loading](#)").

Review

Here are the key ideas covered so far:

- You build apps by selecting components (ingredients) and then telling them what to do and when to do it.

- You use the **Designer** to select components and set each component's properties. Some components are visible and some aren't.
- You can add media (sounds and images) to apps by uploading them from your computer.
- You use the **Blocks Editor** to assemble blocks that define the components' behavior
- `when ... do ...` blocks define *event handlers*, that tell components what to do *when* something happens.
- `call ...` blocks tell components to do things.

Scan the Sample App to your Phone

Scan the following barcode onto your phone to install and run the sample app.



Or [download the apk](#)

Download Source Code

If you'd like to work with this sample in App Inventor, download the [source code](#) to your computer, then open App Inventor, click **Projects**, choose **Import project (.aia) from my computer...**, and select the source code you just downloaded.

Next steps

Now that you know the basics of how App Inventor works we recommend you:

- Complete additional [Tutorials](#).
- Review the [User Guide for App Inventor 2](#).
- Join the [Community Discussion Forum](#).
- Read the guide to [Built-in Blocks](#).
- Or, if you've been using the emulator and want to start using your mobile, you can [set up your Android/iOS device](#) to build apps.

Algo não está funcionando certo? Visite a [página](#) de solução de problemas ou consulte o [Fórum da comunidade do App Inventor](#) para obter ajuda.

Inventor de aplicativos do MIT

© 2012-2021 Instituto de Tecnologia de Massachusetts



Esta obra está licenciada sob uma Licença Creative Commons Atribuição-Compartilhalgal 4.0 Internacional .

[Termos de serviço e política de privacidade](#)

Apoio: Comunidade

Outras dúvidas: E-mail

Twitter: @MITAppInventor

GitHub: mit-cml

Acessibilidade: accessibilidade.mit.edu