**Introdução ao Kodular**

O Kodular permite converter nossas ideias em aplicativos Android, utilizando uma plataforma on-line para criar aplicativos utilizando o método de arrastar e soltar.

Sua operação é muito simples, pois para criar aplicativos basta apenas arrastar "componentes" (os elementos visuais básicos de qualquer aplicativo, a interface e campos de digitação, por exemplo) começando do zero, ou seja, a partir de uma tela em branco.

Esses componentes podem ser caixas de texto, botões laterais ou flutuantes em um menu. Além disso, as aplicações nascidas da Kodular, possuem uma aparência agradável, pois os componentes usam o estilo das instruções da Material Design, uma linguagem para projetar interfaces desenvolvida pelo Google.

Não é necessário ter conhecimento em linguagem de programação, pois o Kodular já traz em blocos os códigos necessários para programar.

Os projetos desenvolvidos no Kodular ficam hospedados com segurança no Google Cloud Platform, não sendo necessário ficar fazendo backups nem download.

Outro ponto importante, é que o Kodular é gratuito, sem necessidade de pagar taxas ou planos para utilizá-lo.

O primeiro passo que devemos realizar, é acessar a plataforma de desenvolvimento, [Kodular.io](http://kodular.io).

Veja na imagem a seguir a tela do ambiente de desenvolvimento Kodular.

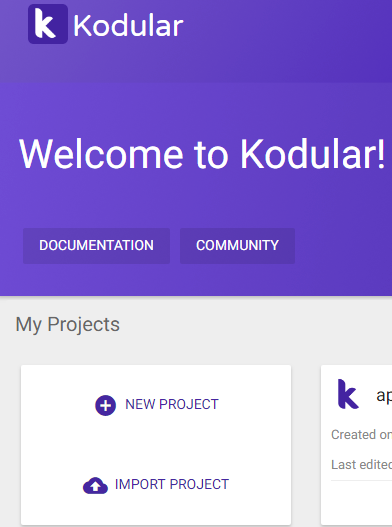


Para ter acesso ao ambiente Kodular, necessitamos realizar nosso logim na plataforma. Para isso, basta clicar em um dos botões **Create App** disponíveis na tela.

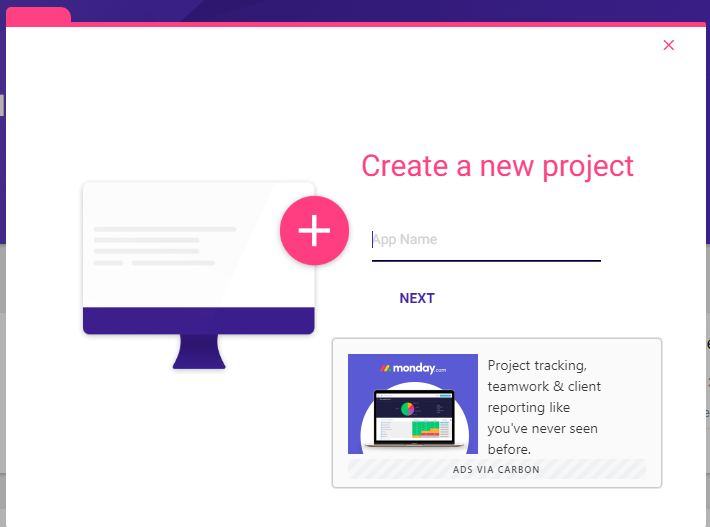
Após o seu clique, será solicitado que seja realizado um login. Utilize um endereço de e-mail do **Google**. Utilize também a mesma senha do seu e-mail para logar na plataforma Kodular, pois o Kodular utiliza a base de dados da Google.

Após a realização do login, será apresentado o espaço onde todos os seus projetos ficarão salvos.

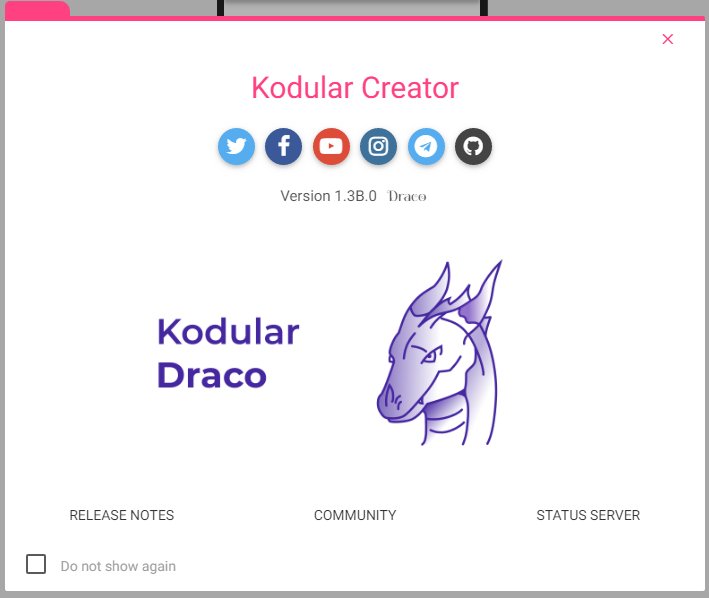
Para criarmos nosso primeiro projeto, clique na opção **New Project** conforme demonstra a imagem a seguir.



Ao clicar em **New Project** será exibida a janela demonstrada na imagem a seguir solicitando o nome do projeto. Digite o nome **ver\_no\_mapa** e clique no botão **Next**.

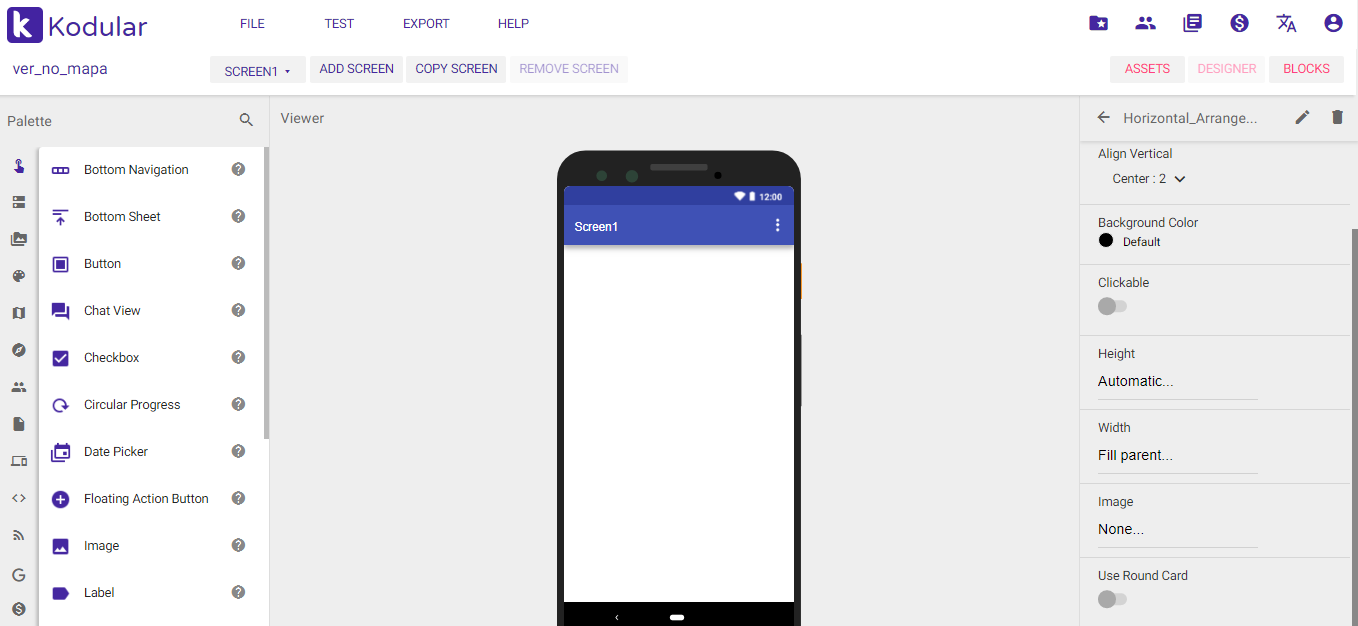


Após digitar o nome do projeto, será apresentado a tela exibida na imagem a seguir, que demonstra a versão do Kodular. Clique no ícone de fechar para começar o desenvolvimento.



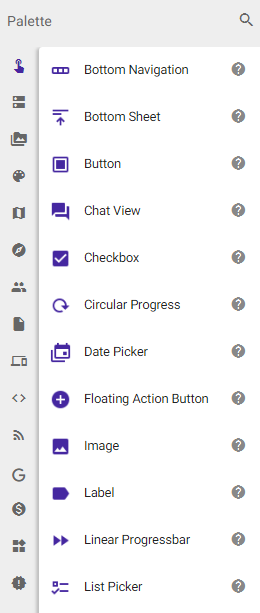
Clique na opção de fechar a janela, para prosseguir com o desenvolvimento.

A tela exibida a seguir, demonstra a área de desenvolvimento de aplicativos. Nela temos todas as opções de criação do leiaute do app.

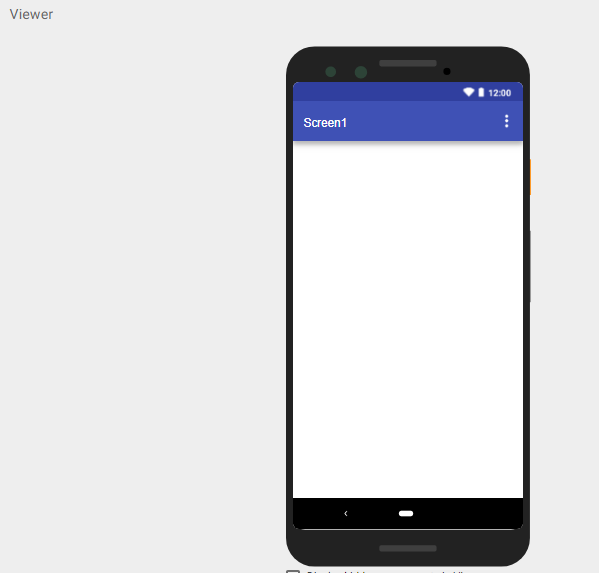


## Desenvolvimento do Leiaute

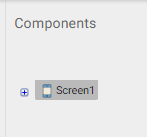
Ao lado esquerdo da tela, encontramos a guia Palette com todos os componentes disponíveis no ambiente Kodular, para inserirmos em nosso app. A imagem a seguir exibe a guia Palette.



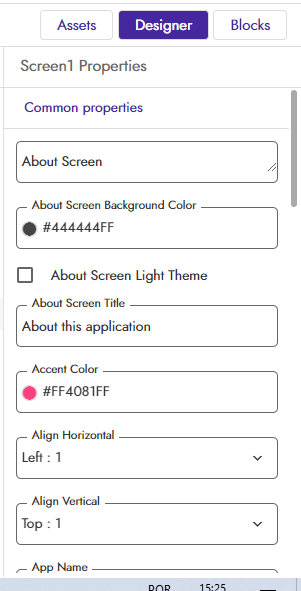
A área VIEWER é o espaço reservado para inserirmos os objetos que irão compor nosso app.



Ao lado direito da tela, podemos encontrar a área com todos os componentes inseridos em nossa Screen. Como ainda não inserimos nenhum objeto, podemos visualizar apenas a **Screen1**. Ao clicar sobre o componente, será ativado a janela com as propriedades do objeto.



No canto direito da tela ficam todas as configurações disponíveis para os componentes, dentro da aba Designer.

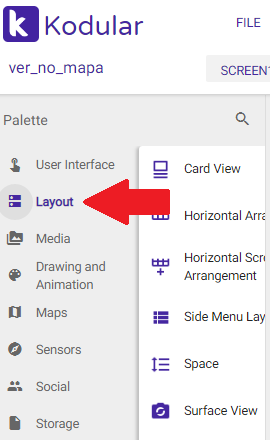


## O aplicativo a ser desenvolvido

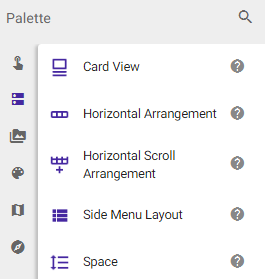
Como exemplo de desenvolvimento de app em Android através da plataforma Kodular, iremos desenvolver um simples aplicativo que terá a função exibir em um mapa o local da latitude e longitude que foi digitado pelo usuário.

Cada objeto que inserimos na screen, fica posicionado sempre na parte superior e ao lado esquerdo da screen. Cada linha por default aceitará apenas um único objeto, mas temos uma solução para adicionar mais que um objeto na linha. O componente responsável por administrar essa função é o **Horizontal Arrangement**.

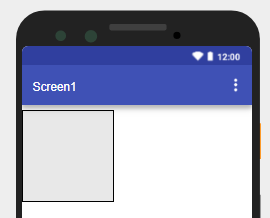
Vamos iniciar nosso desenvolvimento localizando e inserindo um componente **Horizontal Arrangement** em nossa screen1. O primeiro passo é localizar na guia Palette, a seção de **Layout**. Para isso, clique no segundo ícone de cima para baixo ao lado esquerdo da tela, conforme demonstra a imagem a seguir.



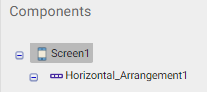
Ao clicar, os objetos da guia Palette estarão disponíveis para utilização, conforme ilustra a imagem a seguir.



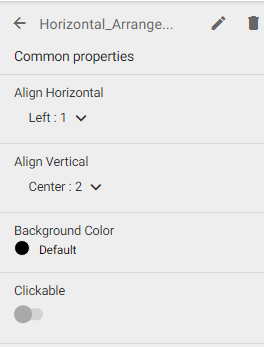
Para inserir o **Horizontal Arrangement**, clique sobre ele, arraste até a área da screen1 e solte. Veja como deverá ficar na imagem a seguir.



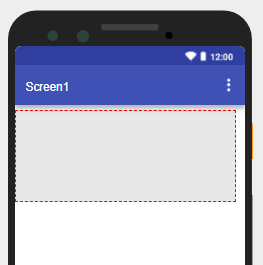
Necessitamos realizar algumas alterações em suas propriedades. Mas primeiramente vamos identificar o local de acesso as propriedades dos objetos. Observe na área de objetos inseridos ao lado esquerdo de sua tela, os nomes screen1 e Horizontal Arrangement, conforme demonstra a imagem a seguir.



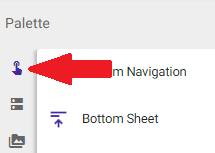
Basta clicar sobre o nome **Horizontal Arrangement** para exibir a janela de propriedades. Veja a janela na imagem a seguir.



A primeira propriedade que alteraremos é a que trata do alinhamento vertical dos objetos que estarão dentro da Horizontal Arrangement. Altere a propriedade **Align Vertical** para a opção **center**.  
Para a propriedade **Width** que trata da largura de exibição do objeto na tela, indicaremos a opção **Fill Parent**, pois queremos que o objeto ocupe toda a área da tela.  
Ao término destas configurações, veja na imagem a seguir como deverá estar sua screen.



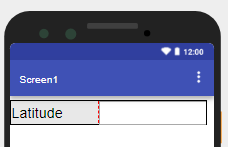
Dentro do espaço da Horizontal Arrangement que preparamos acima, iremos preparar o espaço para a digitação da Latitude. Para isso devemos inserir uma Label e uma Textbox que estão na guia Palette e na seção de **User Interface**, conforme demonstra a imagem a seguir.



Insira uma Label e ao seu lado direito uma Textbox, ambas dentro da Horizontal Arrangement.

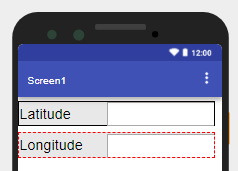
Acesse a área das propriedades e realize as seguintes configurações nas propriedades da Label:  
**Font Size - 20  
Width - 100 pixels  
Text - Latitude**

Após as configurações da label, veja na imagem a seguir como deverá estar a sua screen1.



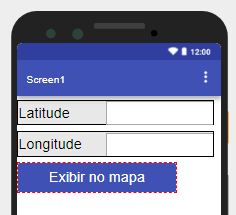
Precisaremos inserir um espaço para a digitação da Longitude. Vamos repetir o procedimento e inserir mais uma horizontal arrangement abaixo da existente em sua screen1. Faça as mesmas configurações vista na horizontal arrangement anterior e altera as propriedades **Align Vertical** e **whidt**. Em seu interior, insira mais uma label e outra textbox. Nessa nova label, realize as mesmas alterações nas propriedades da label anterior com apenas uma mudança: para a propriedade **Text** digite a palavra **Longitude**.

Após as configurações da label, veja na imagem a seguir como deverá estar a sua screen1.

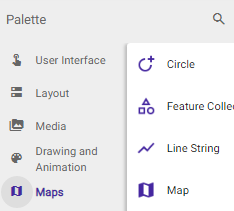


Acesse a guia de componentes Palette, e insira um objeto **Button** em sua screen1. Após, realize as seguintes configurações em suas propriedades:  
**Background Color - Indigo  
Font Size - 20  
Width - 80 porcent  
Text - Exibir no mapa  
Text Alignment - Center**

Veja na figura a seguir como a sua screen está até o momento.



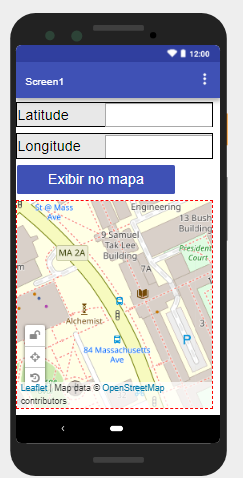
Para inserir um mapa para a visualização na screen1, primeiramente acesse a guia **Maps** Palette. A imagem a seguir demonstra a seleção da guia **Maps**.



Na guia **Maps** localize o objeto **Map**, selecione e arraste-o para a sua screen1 abaixo do botão que tem a função de exibir a latitude e longitude no mapa.

Acesse as propriedades do mapa e realize as seguintes alterações:  
**Height - Fill Parent  
Width - Fill Parent**

A imagem a seguir, exibe a tela que temos até o momento.



Neste momento já poderemos começar a realizar a programação dos comandos através dos blocos.

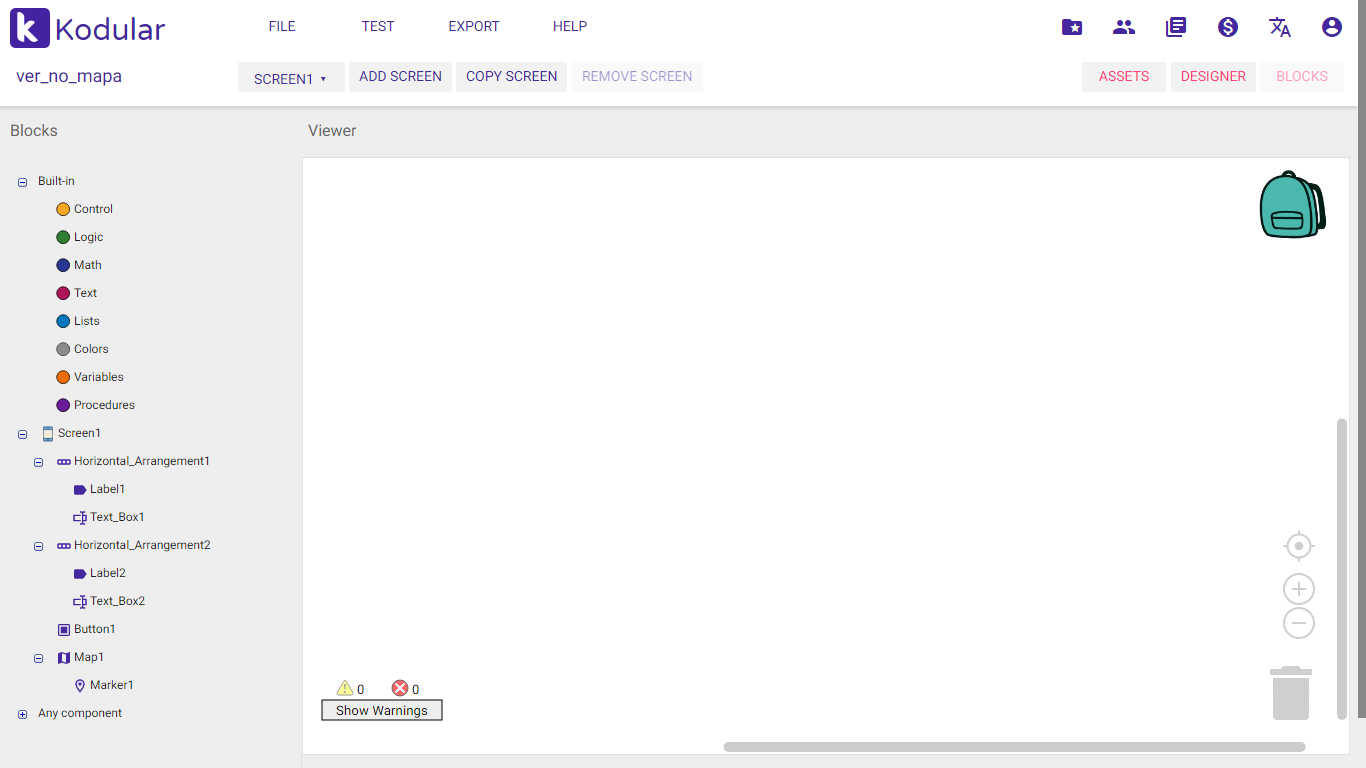
## Inserindo os comandos

Após a criação do leiaute de nosso app, chegamos no momento de inserir os códigos da programação. Porém, na plataforma Kodular, os comandos já estão disponibilizados em formatos de blocos. Primeiramente, vamos acessar a área dos blocos, clicando no botão **Blocos** no canso superior direito da sua tela.

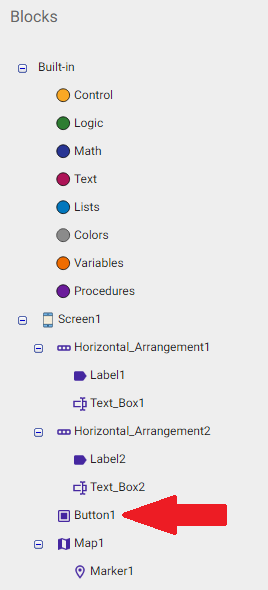
A imagem a seguir exibi o botão **Blocos**.



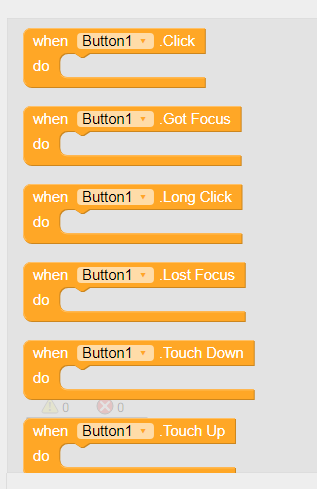
Ao acessar a área de blocos, será exibido a tela para programação, conforme imagem a seguir.



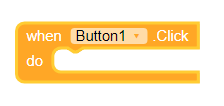
Ao lado esquerdo da tela, encontramos a área da **Built-in**, é nela que encontraremos todos os comandos necessários para a nossa programação.  
Como iremos programar o botão para e exibição das coordenadas no mapa, precisamos primeiramente localizar o objeto **Button1** que está na guia **Built-in**, conforme imagem demonstrada a seguir.



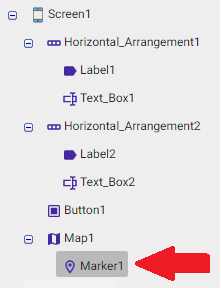
Após localizar o objeto Button1, clique sobre ele para habilitar a exibição de todos os eventos, funções e propriedades do objeto, conforme imagem exibida a seguir.



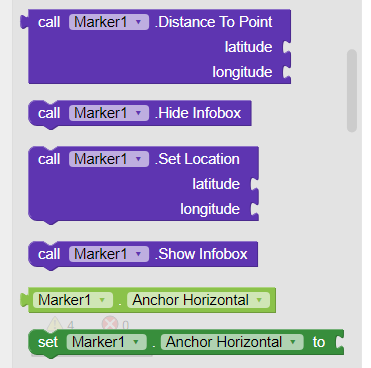
Após a exibição dos blocos, devemos localizar o evento **When Button1.Click**, clicar e arrastá-lo para a área de **Viewer** ao lado esquerdo da tela, para podermos começar a programá-lo. A imagem a seguir, nos mostra o botão inserido na área de programação Viewer.



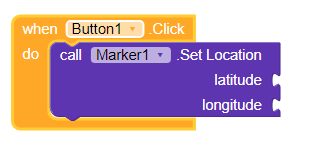
Quando o usuário clicar no button1, deveremos criar o marcador indicando o local da latitude e longitude informada pelo usuário do app. Para tanto, localize na guia **Built-in** o objeto **Maker1**, conforme demonstra a imagem a seguir.



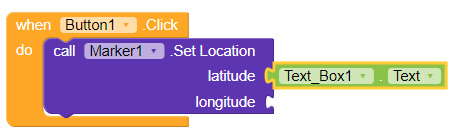
Após clicar sobre ele, identifique o comando **Call Marker1.Set Location**, conforme demonstra a imagem a seguir.



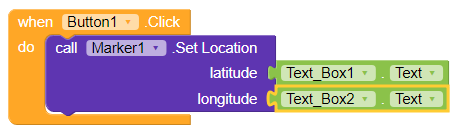
Selecione o comando **Call Marker1.Set Location** e insira-o dentro do espaço do **button1**, conforme exibe a figura a seguir.



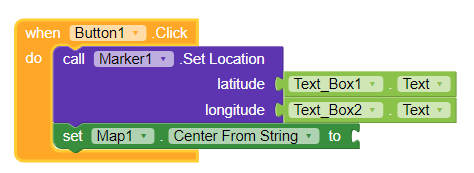
Note que no bloco que acabamos de inserir, existem duas opções de entradas: Uma para a Latitude e uma para a Longitude. Essas informações estão disponíveis nas textboxs que o usuário irá preencher. Para utilizá-las, selecione da guia **Built-in** o objeto **Textbox1** e após, localize o comando **Text\_Box1.Text** e posicione-o na entrada referente a latitude, conforme demonstra a imagem a seguir.



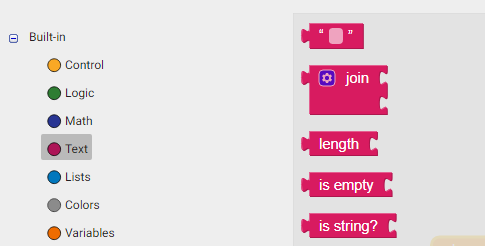
Da mesma maneira que a anterior, só que agora com a **Textbox2**, selecioneo comando **Text\_Box2.Text** referente a longitude e encaixe o comando no último espaço disponível. A imagem a seguir demonstra o bloco de comando concluído.



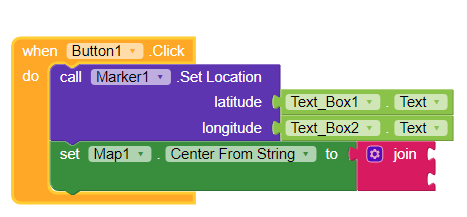
Com o procedimento acima, acabamos e configurar a exibição do marcador de exibição do local em seu mapa, faltando ainda posicionar o mapa para que seja exibido no mesmo local em que o marcador se encontra. Para isso, na guia **Built-in** localize o objeto **Map1** e localize o comando **Set Map1.CenterFrom String To** e posicione-o logo abaixo do último bloco inserido na tela Viewer.A imagem a seguir demonstra o comando **Set Map1.CenterFrom String To** posicionado.



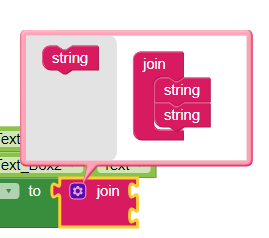
Este comando, centralizará o mapa em uma coordenada da latitude e longitude. Porém como essas informações estão em objetos diferentes, necessitamos realizar um comando que realize a junção das informações. O comando que realiza esta função o **Join**, está disponível na guia **Built-in** na seção **Text**. A imagem a seguir demonstra o bloco **Join**.



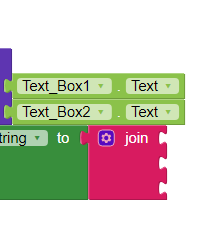
Encaixe-o na frente do bloco de comando **Set Map1.CenterFrom String To**, conforme demonstra a imagem a seguir.



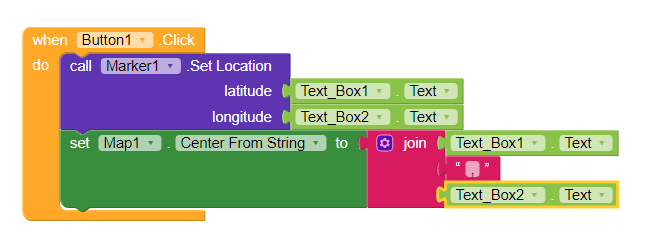
O bloco **Join**, possui apenas duas opções de encaixe e como precisamos incluir a latitude e a longitude separadas por uma vírgula (,) necessitamos acrescentar mais uma opção no Join. Para isso no próprio comando Join já inserido, localize um ícone azul de configuração e clique sobre ele. Será apresentado um complemento, conforme demonstrado na imagem a seguir.



Clique na palavra **String** a esquerda, e arraste-a para o lado direito. O resultado, será o exibido na imagem a seguir.



No primeiro expaço devemos inserir o bloco da latitude que está no bloco de comando **Textbox1.Text**.  
No segundo bloco, acesse a guia **Built-in** e na seção **Text** localize o bloco de texto vazio e encaixe no Join. Dentro de seu interior digite apenas uma vírgula (,) sem espaços.  
No último espaço insera o bloco da longitudee que está disponível no bloco de comando **Textbox2.Text**. A imagem a seguir nos mostra todos os blocos de comandos realizados e finalizados.



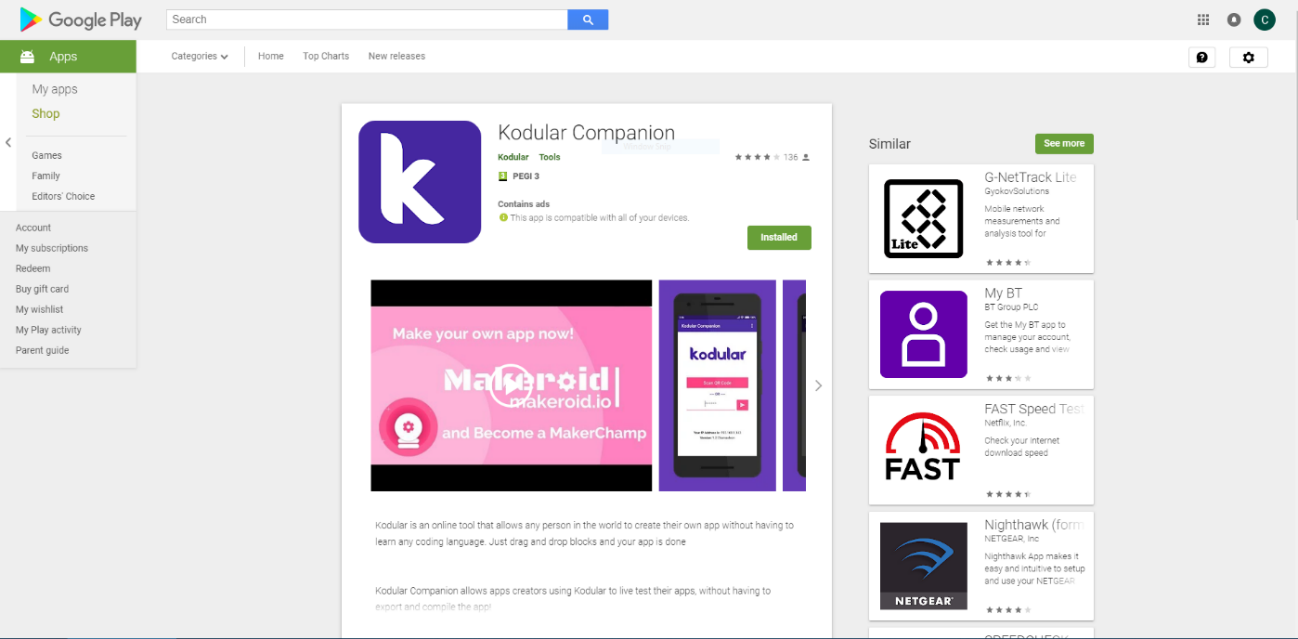
No próximo capítulo iremos instalar o aplicativo em um dispositivo Android para realizarmos os testes.

## Testando o seu aplicativo

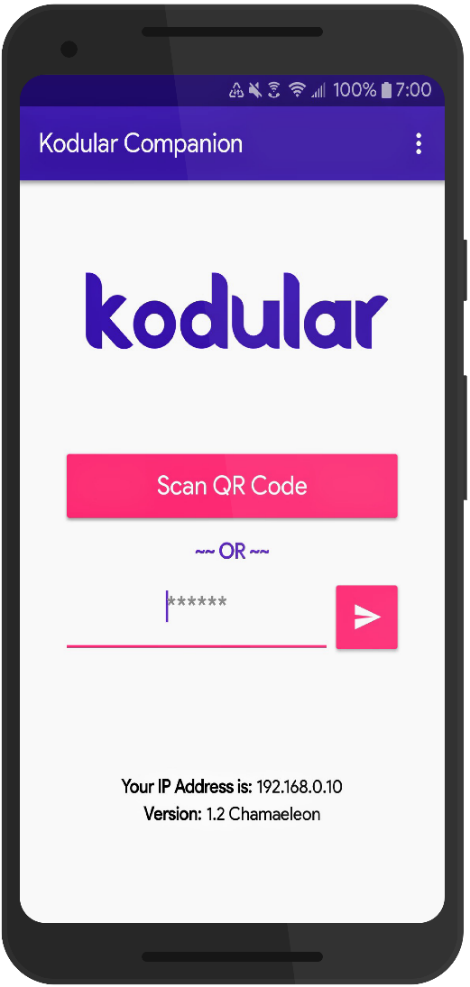
Neste curso, veremos como testar o app que desenvolvemos, sem a necessidade de instalá-lo em seu dispositivo Android.

O Kodular possui um dispositivo que nos ajudará nessa tarefa de utilizar o nosso aparelho Android para realizar a emulação do app que desenvolvemos. O primeiro passo que temos que realizar é instalar em nosso dispositivo Android o app **Kodular Companion**. Realize o download e a instalação do Kodular Companion na Google Play Store, através do [link](file:///C:\Users\adria\OneDrive\Documentos\Escola\Cursos%20Oswaldo\curso%20Kodular\%20https:\play.google.com\store\apps\details?id=io.makeroid.companion).

Você verá a tela do app conforme imagem a seguir.



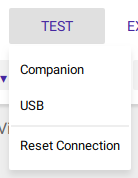
Realize a sua instalação e abra o app em seu dispositivo Android. A seguinte tela será apresentada:



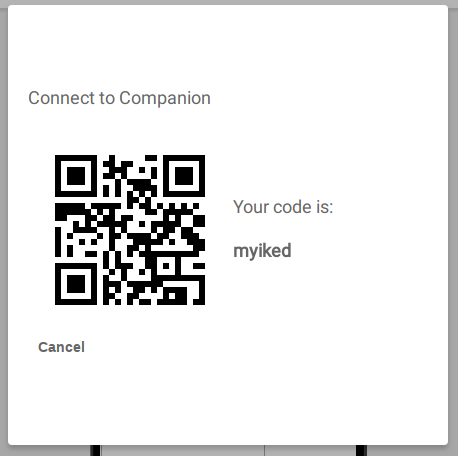
Agora no ambiente de desenvolvimento do Kodular, na Barra de aplicativos, escolha a opção **Test**. A imagem a seguir demonstra a Barra de aplicativos.



Clique na opção "Companion" para obter um QR Code que vamos usar para conectar nosso telefone ao ambiente de desenvolvimento do Kodular. Veja na imagem a seguir.



Aguarde alguns instantes até que seja gerado o código QR, conforme demonstra a imagem a seguir.



Após o surgimento do QR Code em seu computador, retorne ao app **Kodular Companion** no seu dispositivo Android, e clique na opção **Scan QR Code** para a leitura das informações ou simplesmente digite o código que você vê no ambiente do Kodular e clique em enviar.

Você acabou de vinculou seu telefone Android ao Kodular Creator e dentro de alguns instantes, seu app estará disponível para testes em seu dispositivo Android.

Lembre-se de que esse processo de conexão não é um processo único e você precisa realizar esta tarefa toda vez que fechar o Kodular ou Kodular Companionr.

Esta é apenas uma maneira que temos de testar um app e você precisará estar conectado a uma rede Wi-fi em seu dispositivo Android, para que o processo aconteça.