
AGENDA 8

LOCALIZAÇÃO



GEEaD - Grupo de Estudos de Educação a DistânciaCentro de Educação
Tecnológica Paula Souza

GOVERNO DO ESTADO DE SÃO PAULO
EIXO TECNOLÓGICO DE INFORMAÇÃO E COMUNICAÇÃOCURSO TÉCNICO EM
DESENVOLVIMENTO DE SISTEMAS PROGRAMAÇÃO MOBILE I

Expediente

Autor:

GUILHERME HENRIQUE GIROLI

Atualização Técnica:

Rogério Galdiano de Freitas

Revisão Técnica:

Eliana Cristina Nogueira

BarionRevisão

Gramatical:

Juçara Maria Montenegro Simonsen

SantosEditoração e Diagramação:

Flávio Biazim

São Paulo – SP, 2021

GPS (Global Positioning System)

Os sistemas de localização já fazem parte da nossa rotina através dos aplicativos para smartphone. Muitas vezes não percebemos que utilizamos, e muitas pessoas desconhecem, mas utilizam os serviços proporcionados pelo GPS.

Um exemplo é quando enviamos nossa localização através de um aplicativo de mensagens para nosso amigo. Para que essa localização seja enviada, primeiramente nosso dispositivo efetuou uma busca e localizou as coordenadas geográficas que este aparelho se encontra.

E para localizar essas coordenadas, o dispositivo utiliza o sistema de GPS, que são informações obtidas através de satélites e processadas para estipular uma determinada localização.

O sistema de GPS é controlado pelo Governo Norte Americano, e é gratuito para utilização da população mundial. Essa utilização se tornou popular graças aos aparelhos smartphones, que introduziram este recurso, e hoje é um hardware básico em qualquer aparelho.

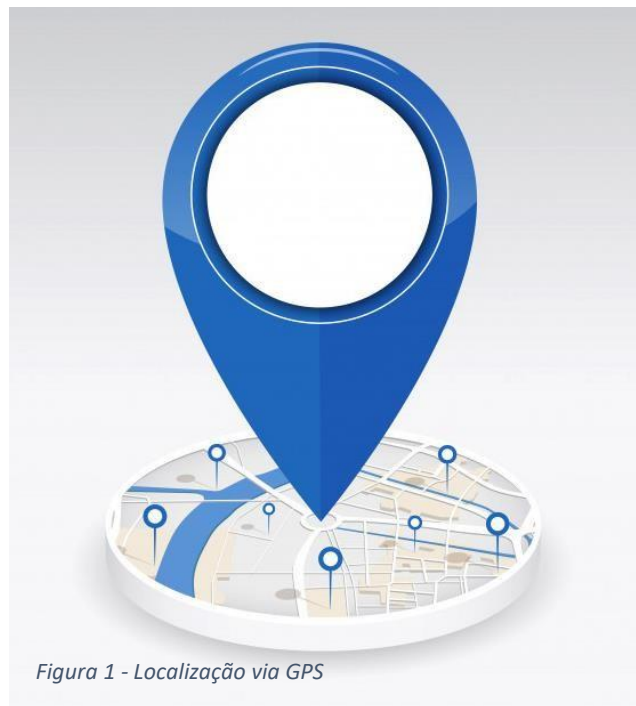


Figura 1 - Localização via GPS

Localização no Android

Segundo o site oficial de desenvolvimento do Android, disponível em: <https://developer.android.com/guide/topics/location/strategies>:

“Saber onde o usuário está permite que seu aplicativo seja mais inteligente e forneça melhores informações ao usuário. Ao desenvolver um aplicativo com reconhecimento de local para Android, você pode utilizar o provedor de localização de rede do GPS e do Android para adquirir o local do usuário. Embora o GPS seja mais preciso, ele só funciona ao ar livre, consome rapidamente a energia da bateria

e não retorna à localização tão rapidamente quanto os usuários desejam. O provedor de localização de rede do Android determina a localização do usuário usando sinais de torre de celular e Wi-Fi, fornecendo informações de localização de maneira interna e externa, responde mais rapidamente e usa menos energia da bateria. Para obter a localização do usuário em seu aplicativo, você pode usar o GPS e o Network Location Provider, ou apenas um”.

Quando o projeto requerer uma localização mais precisa do aparelho, é fundamental utilizar o GPS, em conjunto com o provedor do Android ou sozinho. Uma vez que o sistema de localização que não utiliza GPS, pode prover uma localização com uma margem de erro maior, ou seja, pode atribuir uma localização com um erro de quilômetros da realidade do dispositivo.

Quando utilizamos um aplicativo que contém as funções de localização, não imaginamos alguns desafios que o sistema necessita vencer para entregar uma posição mais real possível daquela determinada virtualmente.

A **Figura 2** mostra alguns aspectos que podem influenciar na determinação de uma determinada localização geográfica para um aparelho smartphone.

Desafios na determinação da localização do usuário

A obtenção da localização do usuário a partir de um dispositivo móvel pode ser complicada. Há vários motivos pelos quais uma leitura de local (independentemente da origem) pode conter erros e ser imprecisa. Algumas fontes de erro no local do usuário incluem:

- **Várias origens de localização**

GPS, Cell-ID e Wi-Fi podem fornecer uma pista para a localização dos usuários. Determinar qual usar e confiar é uma questão de compensações em precisão, velocidade e eficiência da bateria.

- **Movimento do usuário**

Como a localização do usuário é alterada, você deve contabilizar o movimento reestimando a localização do usuário de vez em quando.

- **Precisão de variação**

As estimativas de localização provenientes de cada fonte de localização não são consistentes em sua precisão. Uma localização obtida há 10 segundos a partir de uma fonte pode ser mais precisa do que a localização mais recente de outra ou da mesma fonte.

Figura 2 - Desafios na determinação de localização do usuário. Disponível em: <https://developer.android.com/guide/topics/location/strategies>.

Obter a localização através da classe LocationManager

O sistema Android possui classes para facilitar o desenvolvimento de aplicativos. Alguns recursos mais complexos contam com a assessoria delas para que o desenvolvedor ganhe tempo na produção dos projetos.

A **LocationManager** foi desenvolvida para que o aplicativo comunique com o sistema de localização do dispositivo, seja através do GPS ou pelo provedor de localizações.

A classe pode ser utilizada para solicitar em determinados intervalos de tempo a localização do aparelho, ou também pode ser programada para que quando o dispositivo entre em uma determinada área geográfica chame uma aplicação ou serviço.

Permissões de uso para a localização do aparelho

Já vimos anteriormente que alguns recursos do aparelho são bloqueados para o aplicativo. Desta forma quando surge a necessidade de utilização, o aplicativo tem que solicitar ao usuário o seu consentimento para que ele tenha acesso.

O sistema de localização do aparelho vem bloqueado para os aplicativos. Vamos informar no arquivo **Manifest.xml** a nossa intenção de utilizar esse hardware do aparelho.



Vamos acompanhar o início do desenvolvimento do nosso projeto “**Localizacao**” que utiliza a localização do aparelho Android.

Projeto Localização

- Abra a plataforma de desenvolvimento do Kodular: <https://www.kodular.io/creator>.
- Clique no botão **Create Project**.

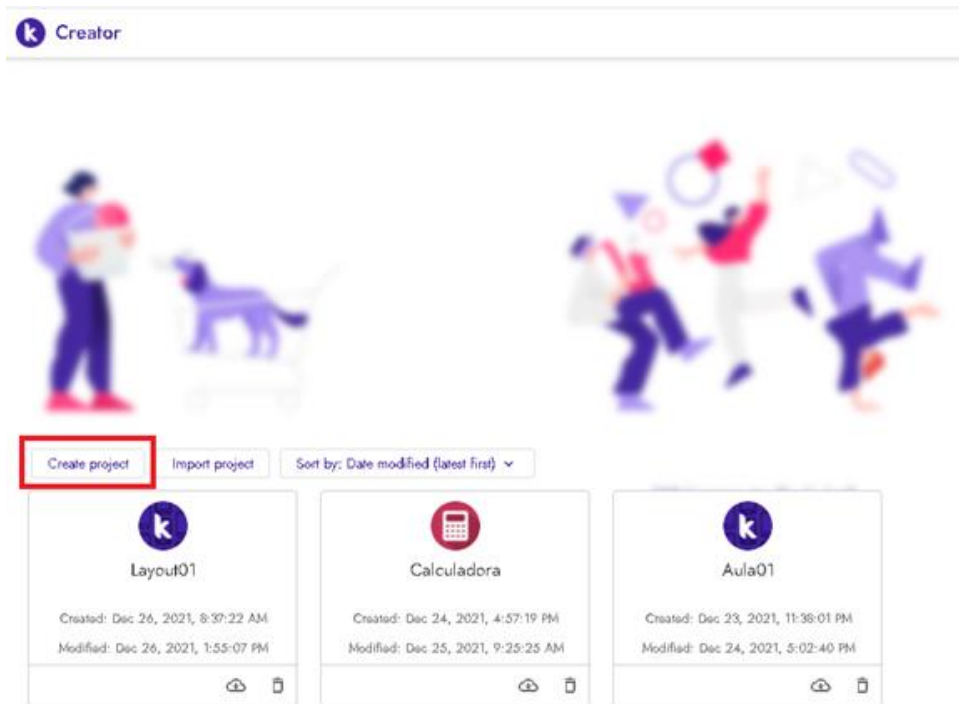


Figura 3 - Criando um novo projeto

- Digite o nome **LOCALIZACAO** e clique no botão **Next**.



Figura 4 - Criando o projeto Compartilhar

- Clique no botão **Finish** para finalizar a criação do novo projeto.
- Altere as propriedades do objeto **SCREEN**.

Propriedade	Valor	Função
Title	Sensores	Definir o título da aplicação em desenvolvimento.
Align Horizontal	Center	Alinhar todos os componentes na horizontal ao centro.
Align Vertical	Center	Alinhar todos os componentes na vertical ao centro.

- Escolha um ícone para sua aplicação, sendo que a padronização do ícone deverá ser feita pelo botão **Configurações**.
- Insira um componente **Button**, categoria **User Interface**.
- Altere as propriedades do componente **Button**.

Propriedade	Valor	Função
Font Bold	✓ Marcado	Definir a opção negrito
Font Size	18	Definir o tamanho da fonte
Width	200px	Definir a largura do componente
Text	Mostrar Localização	Definir o conteúdo a ser exibido do componente button.
Text Alignment	Center	Definir o alinhamento ao centro.
Name	btn_Localiza	Definir o nome do componente.

- Clique no componente **Label** e insira na área de **VIEWER**.
- Altere as propriedades do objeto **Label**.

Propriedade	Valor	Função
Font Size	16	Definir o tamanho da fonte.
Font Bold	✓ Marcado	Definir a opção de negrito na fonte
Text	Localização:	Definir o texto que será apresentado pelo Label .
Name	lbl_Titulo	Definir o nome do componente.

- Clique no componente **Label** e insira na área de **VIEWER**.
- Altere as propriedades do objeto **Label**.

Propriedade	Valor	Função
Font Size	16	Definir o tamanho da fonte.
Font Bold	✓ Marcado	Definir a opção de negrito na fonte
Text Color	#F44336FF	Define a cor da fonte da letra
Text		Definir o texto que será apresentado pelo Label .
Name	lbl_Localiza	Definir o nome do componente.

- Inserir o componente **Space** da categoria **Layout**, categoria **General** na área de **VIEWER**.
- Altere as propriedades do objeto **Space**.

Propriedade	Valor	Função
Height	40px	Definir a altura em px para representar o espaçamento.
Name	spc_Linha01	Definir o nome do componente

- Insira um componente **Button**, categoria **User Interface**.
- Altere as propriedades do componente **Button**.

Propriedade	Valor	Função
Enabled	Desmarcado	Define que o botão não estará ativo.
Font Bold	✓ Marcado	Definir a opção negrito
Font Size	18	Definir o tamanho da fonte
Width	200px	Definir a largura do componente
Text	Exibir Mapa	Definir o conteúdo a ser exibido do componente button.
Text Alignment	Center	Definir o alinhamento ao centro.
Name	btn_Mapa	Definir o nome do componente.

- Insira um componente **Button**, categoria **User Interface**.
- Altere as propriedades do componente **Button**.

Propriedade	Valor	Função
Enabled	Desmarcado	Define que o botão não estará ativo
Font Bold	✓ Marcado	Definir a opção negrito
Font Size	18	Definir o tamanho da fonte
Width	200px	Definir a largura do componente
Text	Sair	Definir o conteúdo a ser exibido do componente button.
Text Alignment	Center	Definir o alinhamento ao centro.
Name	btn_Sair	Definir o nome do componente.

- Clique na categoria **Sensors**

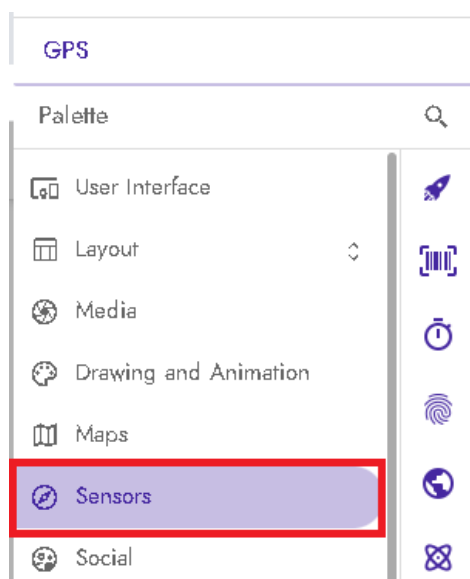


Figura 5 - Categoria Sensors

- Insira um componente **Location Sensor**, categoria **Sensors**.

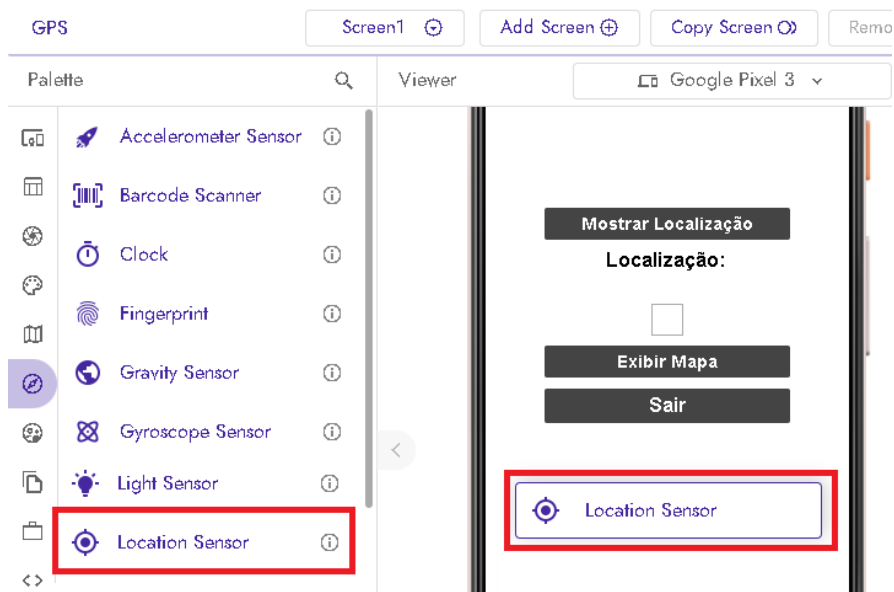


Figura 6 - Inserindo o componente Location Sensor.

- Altere as propriedades do componente **Location Sensor**.

Propriedade	Valor	Função
Name	Localiza_Sensor	Definir o nome do componente.

- Clique na categoria **Connectivity**

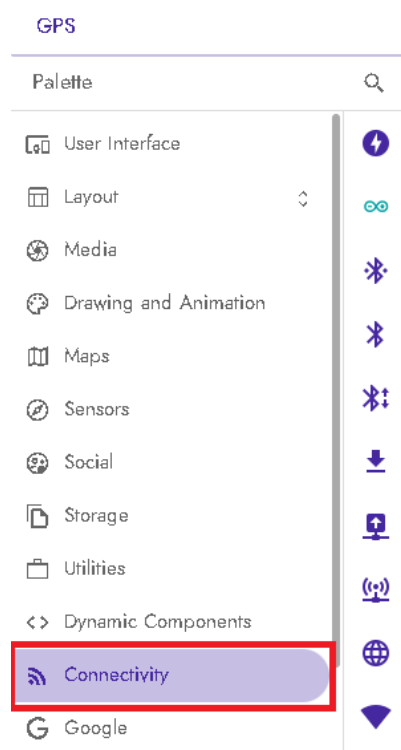


Figura 7 - Aba Connectivity

- Clique no componente **Activity Starter** e arraste para a área **Viewer**.

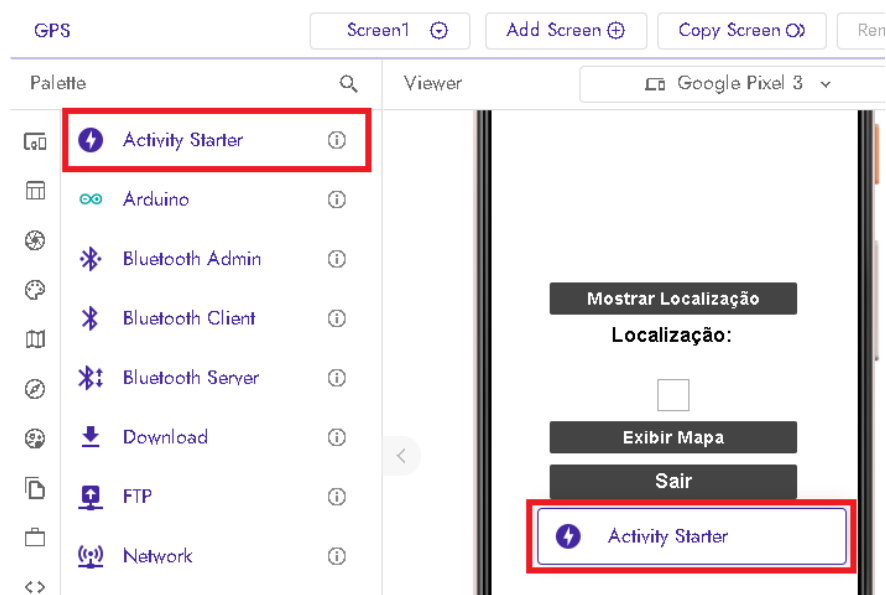
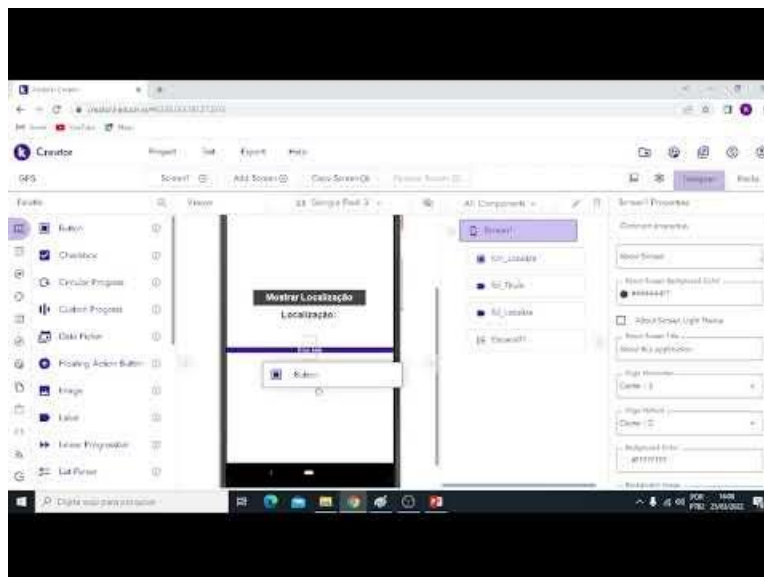


Figura 8 - Inserindo o componente Activity Starter

- Altere as propriedades do componente **Activity Starter**.

Propriedade	Valor	Função
Name	Act_Starter	Definir o nome do componente.

Em caso de dúvida na construção da interface deste aplicativo, assista ao vídeo **Agenda 08 – Criando a interface GPS**, disponível em: <https://youtu.be/JoKmxyaFj10>.



Após a construção do layout do nosso aplicativo, iremos agora alterar para a opção de programação em blocos (**Blocks**).



Figura 9 - Alterando a programação em blocos

Vamos programar. Construa o código de acordo com a **Figura 10**, para que o aplicativo possa realizar todas as funções necessários.

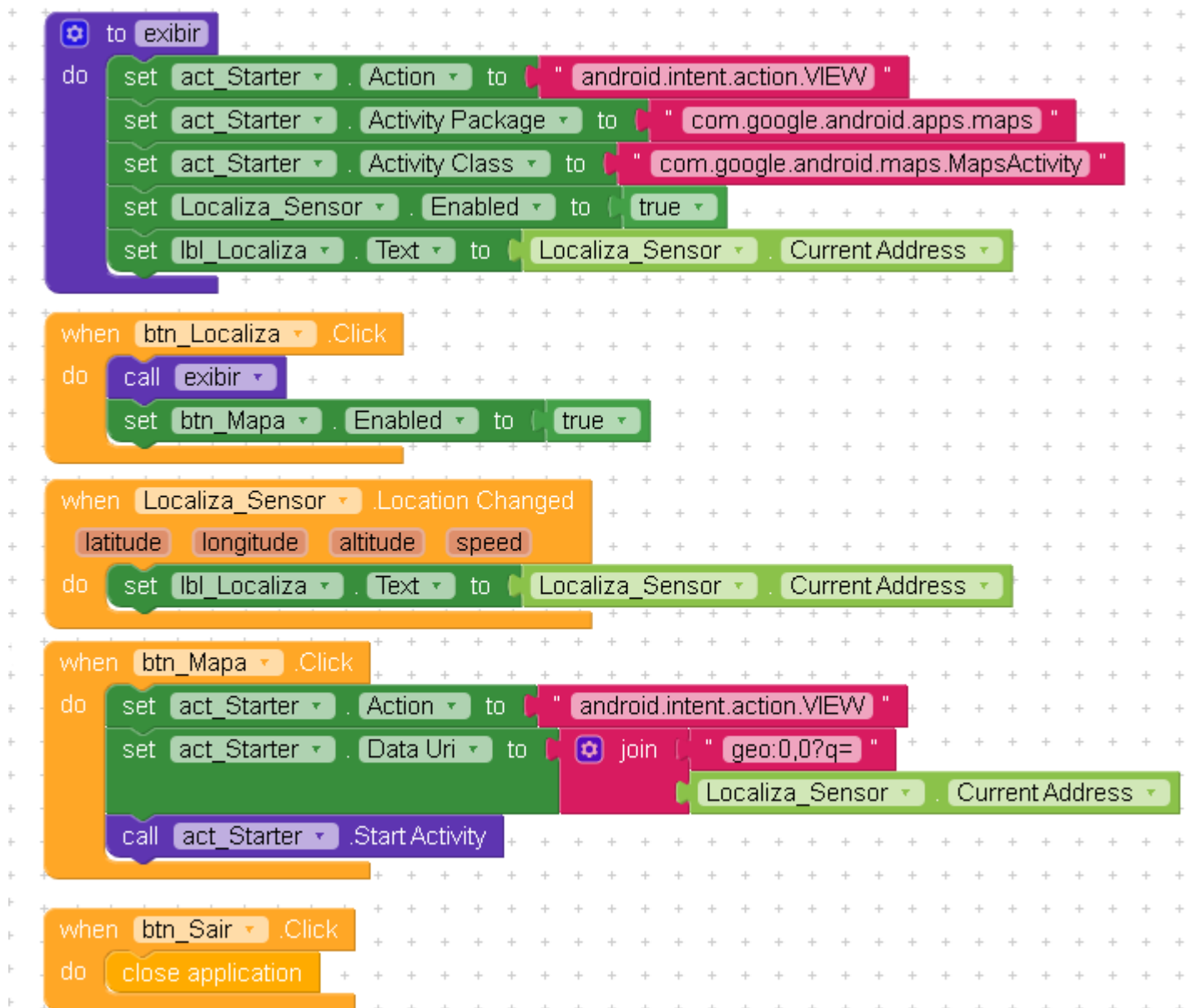


Figura 10 - Bloco de programação.

to exibir

```

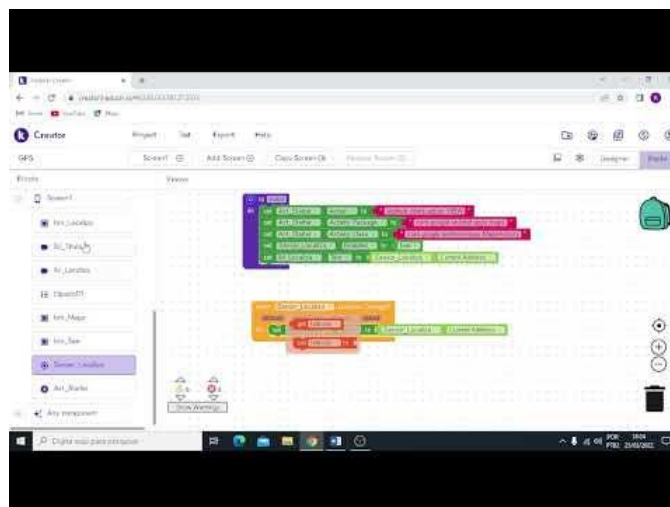
set act_Starter.Action to "android.intent.action.VIEW"
set act_Starter.Activity Package to "com.google.android.apps.maps"
set act_Starter.Activity Class to "com.google.android.maps.MapActivity"
  
```

when btn_Mapa.click

```

set act_Starter.Action to "android.intent.action.VIEW"
set act_starter,Data Uri to join "geo:0,0?q="
  Sensor_Localiza.CurrentAddress
  
```

Ao concluir a programação, assista ao vídeo **Agenda 08 – Criando a programação em blocos GPS**, disponível em: https://youtu.be/t38VvJ3l_do



Para finalizar o projeto, exporte o **arquivo APK** para o dispositivo móvel e realizar a instalação através do aplicativo **Kodular Companion**.

- Clique no menu **Export**, na opção Android App (.apk).

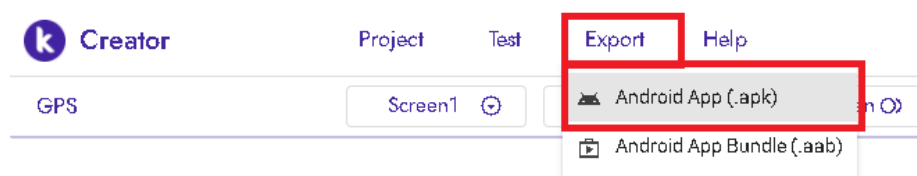


Figura 11 - Menu Export, opção Android App (.apk)

- Utilize o aplicativo **Kodular Companion** para escanear o qrcode e siga todos os passos para a instalação do aplicativo, de acordo com material anterior.

Android App for "GPS"

Scan the QR code on your phone to install the app or download the APK file to your computer.

Note: This link is valid only for 10 minutes. It is recommended to export your app as an Android App Bundle for distribution via Google Play.

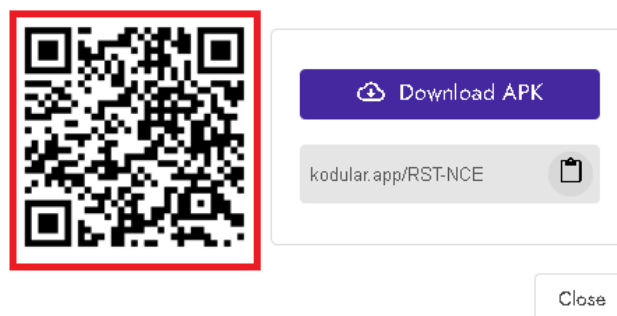


Figura 12 - Disponibilizando o Qrcode.

Ao iniciar o aplicativo, o usuário deverá clicar no botão **Mostrar Localização**, desta forma será exibida a localização atual. Em seguida, o usuário poderá clicar no botão **Exibir Mapa** para visualizar a localização no mapa real.

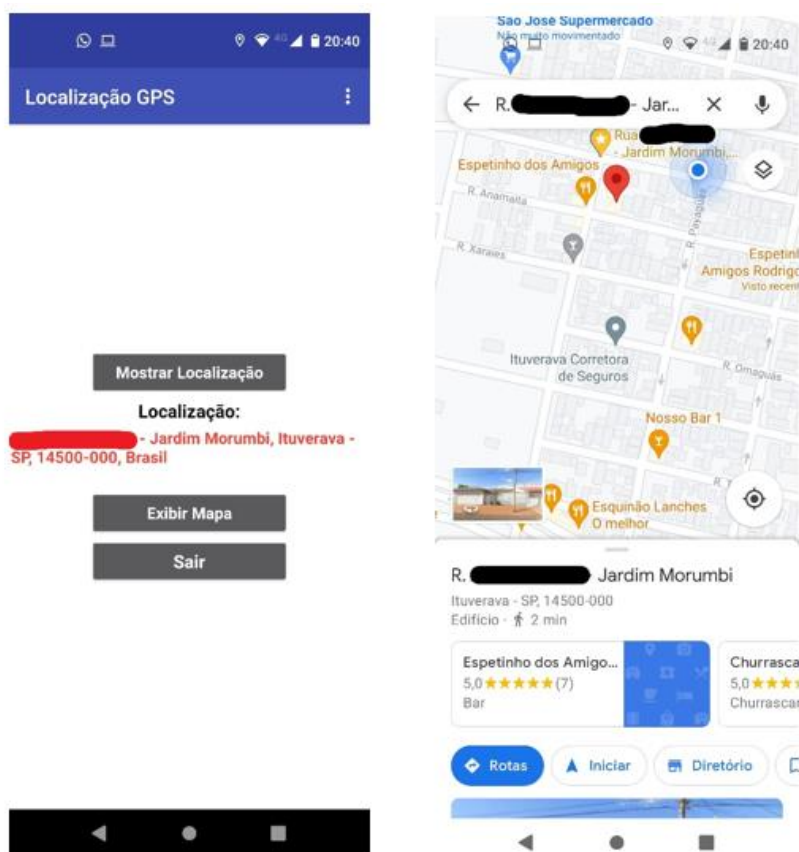


Figura 13 - Exibindo o endereço e a localização no mapa.