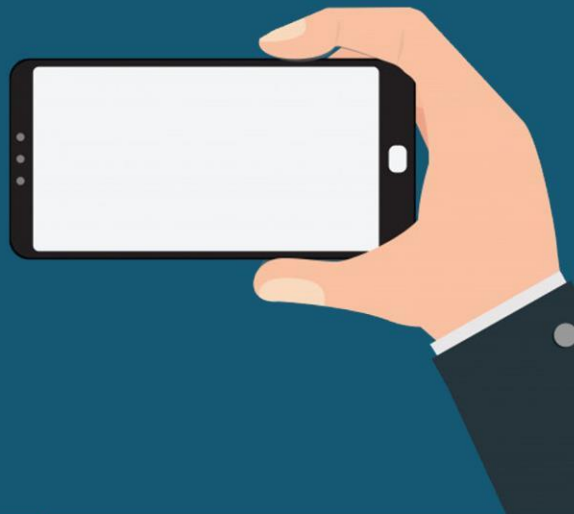


---

# AGENDA 7

---

## SENSORES



GEEaD - Grupo de Estudos de Educação a Distância Centro de Educação  
Tecnológica Paula Souza

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

**Expediente**

Autor:

*GUILHERME HENRIQUE GIROLI*

Atualização Técnica:

*Rogério Galdiano de Freitas*

Revisão Técnica:

*Eliana Cristina Nogueira*

Barion Revisão

Gramatical:

*Juçara Maria Montenegro Simonsen*

Santos Editoração e Diagramação:

*Flávio Biazim*

São Paulo – SP, 2021

## Tipos de Sensores

Os sensores são componentes de hardware e/ou software que permitem ao dispositivo uma interação com o ambiente. Capturando dados, analisando e devolvendo informações para o aparelho e para o usuário.

Encontramos sensores a nível de hardware, ou seja, sensores que são componentes eletrônicos físicos para capturar determinada informação, como por exemplo, a temperatura ou até mesmo a iluminação de um determinado ambiente.

Existem também sensores baseados em software ou sensores virtuais, ou seja, são sensores que para entregar um determinado dado, captura informações de um ou mais sensores de hardware, processa essas informações para entregar um resultado. Um exemplo é o sensor de aceleração linear.

O sistema operacional Android suporta a grande maioria dos sensores encontrados nos aparelhos, e é importante comentar que os aparelhos não possuem todos os tipos de sensor, isso depende muito do fabricante e da versão do dispositivo.

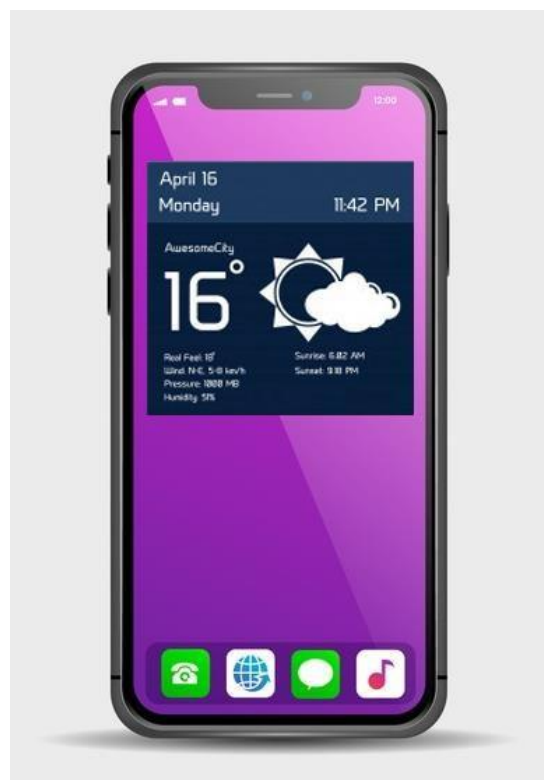


Figura 1 - Dispositivo Móvel com sensores.

A seguir a **Figura 2**, retirada do site oficial do Android Studio, mostra os tipos de sensor suportados pelo sistema operacional, com sua classificação de tipos e breve explicação sobre seu funcionamento.

Para examinar o conteúdo completo sobre sensores acesse o site:

[https://developer.android.com/guide/topics/sensors/sensors\\_overview](https://developer.android.com/guide/topics/sensors/sensors_overview)

Sensor	Tipo	Descrição	Usos Comuns
TYPE_ACCELEROMETER	Hardware	Mede a força de aceleração em $m / s^2$ que é aplicada a um dispositivo em todos os três eixos físicos (x, y, z), incluindo a força da gravidade.	Deteção de movimento (agitar, inclinar, etc.)
TYPE_AMBIENT_TEMPERATURE	Hardware	Mede a temperatura ambiente em graus Celsius ( $^{\circ}C$ ). Ver nota abaixo.	Monitorando a temperatura do ar.
TYPE_GRAVITY	Software ou Hardware	Mede a força da gravidade em $m / s^2$ que é aplicada a um dispositivo em todos os três eixos físicos (x, y, z).	Deteção de movimento (agitar, inclinar, etc.)
TYPE_GYROSCOPE	Hardware	Mede a taxa de rotação de um dispositivo em $rad / s$ em torno de cada um dos três eixos físicos (x, y e z).	Deteção de rotação (girar, girar, etc.)
TYPE_LIGHT	Hardware	Mede o nível de luz ambiente (iluminação) em lx.	Controlando o brilho da tela.
TYPE_LINEAR_ACCELERATION	Software ou Hardware	Mede a força de aceleração em $m / s^2$ que é aplicada a um dispositivo em todos os três eixos físicos (x, y, z), excluindo a força da gravidade.	Monitorando a aceleração ao longo de um único eixo.
TYPE_MAGNETIC_FIELD	Hardware	Mede o campo geomagnético do ambiente para todos os três eixos físicos (x, y, z) em $\mu T$ .	Criando uma bússola.
TYPE_ORIENTATION	Programas	Mede os graus de rotação que um dispositivo faz em torno dos três eixos físicos (x, y, z). A partir do nível API 3, você pode obter a matriz de inclinação e a matriz de rotação de um dispositivo usando o sensor de gravidade e o sensor de campo geomagnético em conjunto com o <code>getRotationMatrix()</code> método.	Determinando a posição do dispositivo.
TYPE_PRESSURE	Hardware	Mede a pressão do ar ambiente em hPa ou mbar.	Monitorando as mudanças de pressão de ar.
TYPE_PROXIMITY	Hardware	Mede a proximidade de um objeto em cm em relação à tela de exibição de um dispositivo. Este sensor é normalmente usado para determinar se um fone está sendo conectado ao ouvido de uma pessoa.	Posição do telefone durante uma chamada.
TYPE_RELATIVE_HUMIDITY	Hardware	Mede a umidade relativa do ar em porcentagem (%).	Monitoramento do ponto de orvalho, absoluta e umidade relativa.
TYPE_ROTATION_VECTOR	Software ou Hardware	Mede a orientação de um dispositivo fornecendo os três elementos do vetor de rotação do dispositivo.	Deteção de movimento e detecção de rotação.

Figura 2 - Tipos de sensores suportados pela plataforma Android. Site:

[https://developer.android.com/guide/topics/sensors/sensors\\_overview](https://developer.android.com/guide/topics/sensors/sensors_overview) Acesso em: 5 de dezembro de 2018

## Estrutura do Sensor

O sensor do dispositivo é utilizado através de uma estrutura padrão, estabelecida pelo Android. O acesso aos seus dados brutos é feito através de classes Java para facilitar o processo de desenvolvimento de aplicativos. A seguir vamos verificar as principais classes para trabalho com sensores.

- **SensorManager:** É utilizado para criar uma instância dos serviços dos sensores presentes no aparelho, com métodos para facilitar o desenvolvimento do aplicativo. Através dela é possível iniciar e parar o uso do sensor, calibrar, verificar a precisão, entre outras funções.
- **Sensor:** Essa classe é utilizada para trabalhar com métodos de um sensor em específico, obtido através do **SensorManager**.
- **SensorEvent:** É uma classe utilizada para criar um objeto sobre determinado evento do sensor, com os dados, horário, precisão do sensor, qual tipo de sensor foi utilizado, entre outras informações.
- **SensorEventListener:** É utilizada para criar duas ações sobre um determinado sensor que esteja em uso. Essas ações são disparadas quando o valor do sensor que está sendo verificado sofre uma mudança chamado de **onSensorChanged()**, e outra ação que é disparada quando a precisão do sensor sofre uma alteração chamada de **onAccuracyChanged()**.

## Boas Práticas ao Utilizar Sensores

Ao utilizar um ou mais sensores, o desenvolvedor precisa levar em consideração alguns fatores, como por exemplo o uso desnecessário da bateria do aparelho. O Android por padrão, não desabilita um sensor quando a tela do dispositivo é desligada, desta forma o desenvolvedor mobile necessita trabalhar com métodos para que o sensor seja desligado quando o mesmo não estiver em uso.

Alguns sensores exigem um alto uso da bateria do aparelho durante seu funcionamento, e deixar esses sensores ativos mesmo não sendo utilizados pode gerar alguns problemas e insatisfação do usuário com o aplicativo.

## Trabalhando com Sensores

Primeiramente o desenvolvedor necessita pesquisar a documentação oficial do Android para conhecer a forma correta de trabalho para cada sensor. Por exemplo, os sensores de giroscópio utilizam coordenadas para passar os dados referentes a localização do dispositivo. A **Figura 3** mostra os eixos x, y e z de um aparelho. Em contrapartida, um sensor de luz, que vamos utilizar para desenvolver o projeto da Ana, utiliza um único dado, para informar a quantidade de iluminação recebida por um dispositivo.

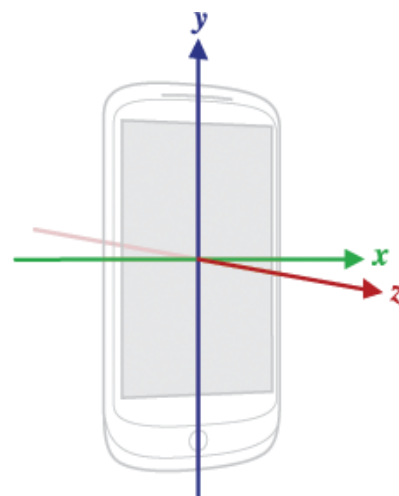


Figura 3 - Eixos x, y e z transmitidos por um sensor de giroscópio.

A documentação oficial do Android aborda todas as formas de utilização presente no sistema operacional, e está disponível pelo site: <https://developer.android.com/guide/topics/sensors/>.

No projeto recebido por Ana, vamos utilizar um sensor de luz, que captura o valor de iluminação recebida por um determinado dispositivo. Esse sensor também é utilizado para o processo de iluminação da tela, você já deve ter notado que em ambientes escuros, o celular retira automaticamente o brilho da tela para o conforto do usuário e economizar energia. E em ambientes claros o sistema operacional aumenta o brilho da tela para facilitar a visualização do usuário.

Vamos iniciar o projeto!

## Projeto – “Sensores”

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

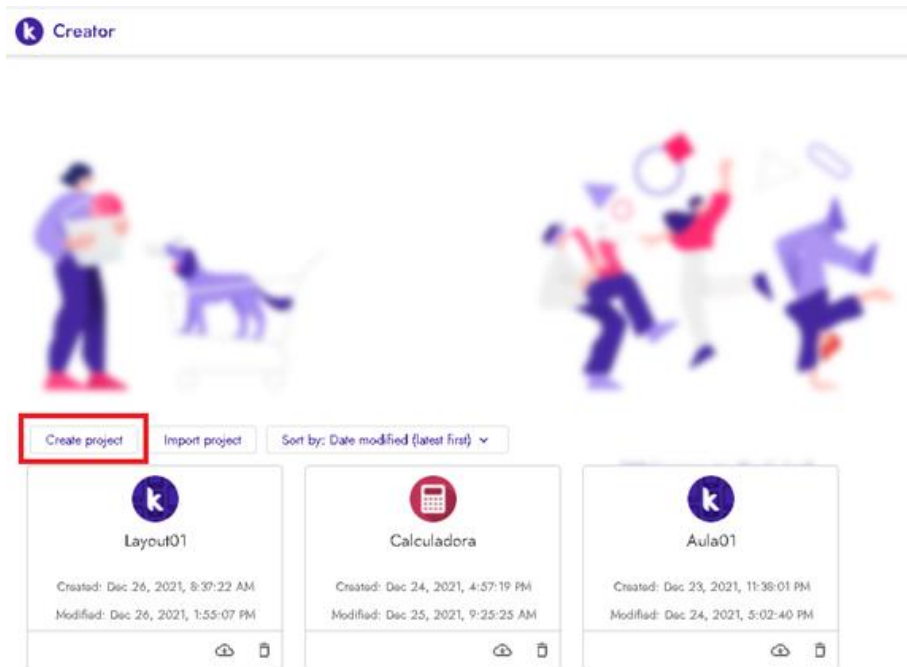
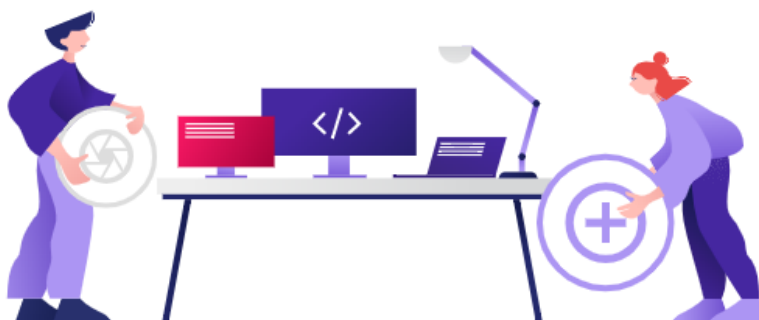


Figura 4 - Criando um novo projeto

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

Create new project



Give your new project a name

Cancel

Next

Figura 5 - Criando o projeto Compartilhar

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

Propriedade	Valor	Função
<b>Title</b>	Sensores	Definir o título da aplicação em desenvolvimento.
<b>Align Horizontal</b>	Center	Alinhar todos os componentes 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	Ligar Sensor de Luz	Definir o conteúdo a ser exibido do componente button.
Text Alignment	Center	Definir o alinhamento ao centro.
Name	btn_Sensor	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	Luminosidade:	Definir o texto que será apresentado pelo <b>Label</b> .
Name	lbl_luz	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
Font Bold	✓ Marcado	Definir a opção negrito
Font Size	18	Definir o tamanho da fonte
Width	200px	Definir a largura do componente
Text	Impressão Digital	Definir o conteúdo a ser exibido do componente button.
Text Alignment	Center	Definir o alinhamento ao centro.
Name	btn_Digital	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		Definir o texto que será apresentado pelo <b>Label</b> .
Name	lbl_Digital	Definir o nome do componente.

- Clique na categoria **Sensors**

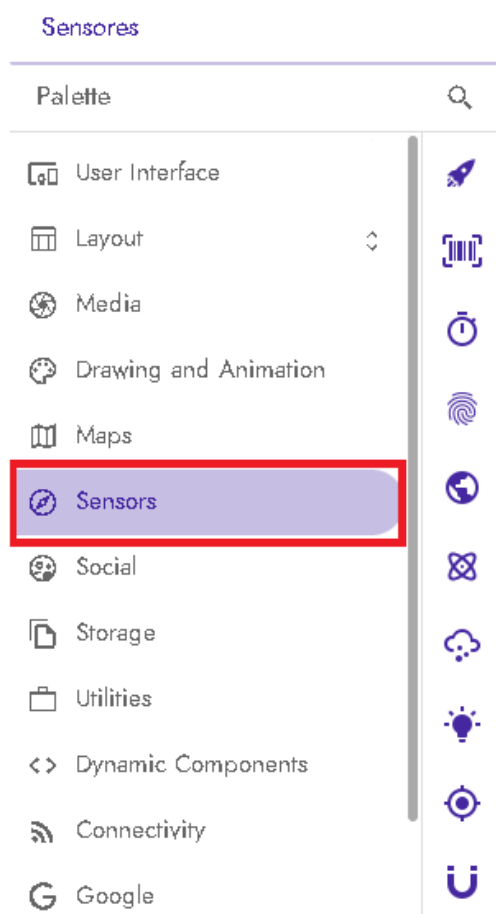


Figura 6 - Categoria Sensors

- Clique no componente **Accelerometer Sensor** e insira na área de **VIEWER**.

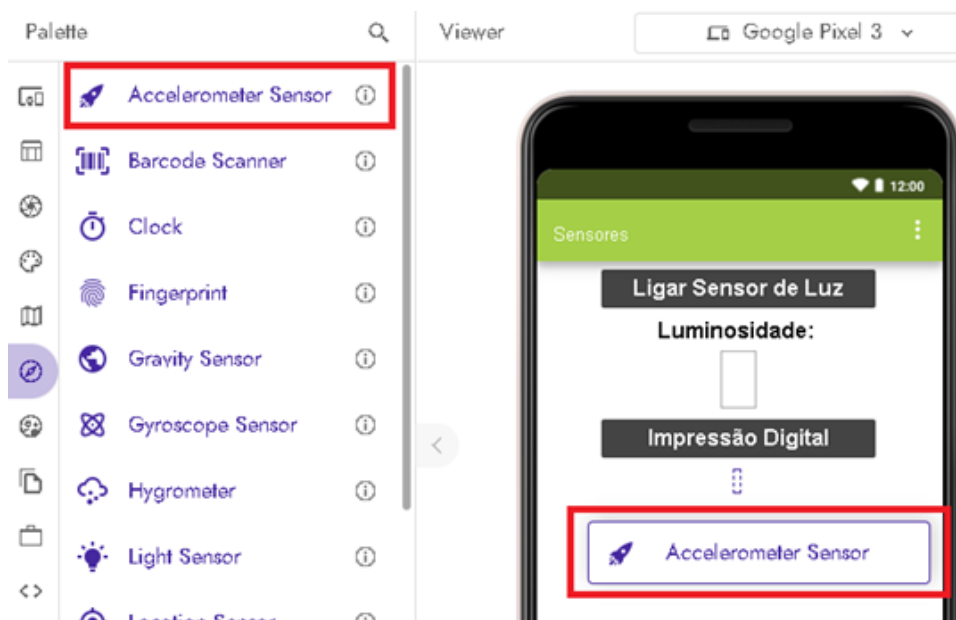


Figura 7- Inserir componente Accelerometer Sensor

- Altere as propriedades do objeto **Accelerometer Sensor**.

Propriedade	Valor	Função
<b>Enabled</b>	✓ Marcado	Definir o componente como ativado.
<b>Name</b>	Acelero_Sensor	Definir o nome do componente.

- Clique no componente **Light Sensor** e insira na área de **VIEWER**.

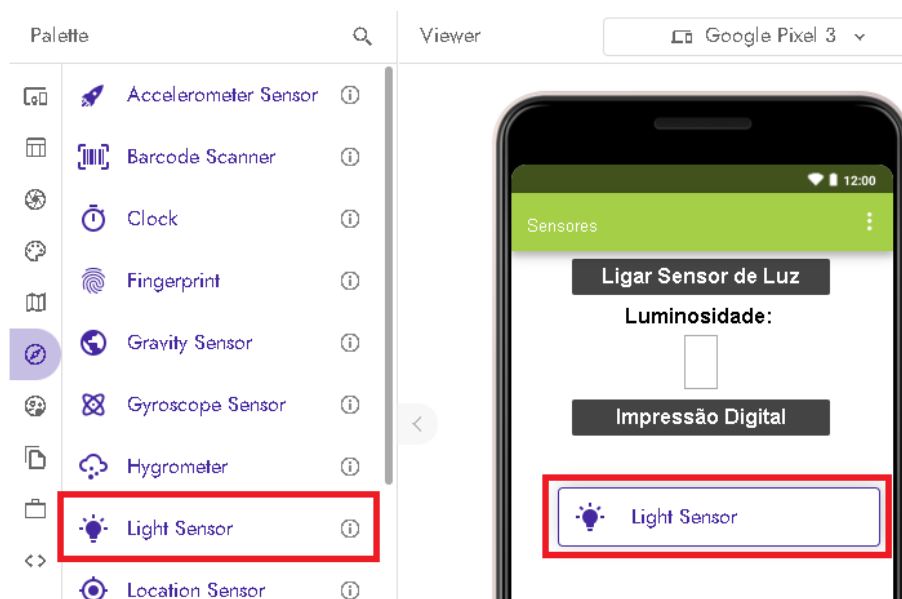


Figura 8 - Inserir componente Light Sensor.

- Altere as propriedades do objeto **Light Sensor**.

Propriedade	Valor	Função
<b>Enabled</b>	✓ Marcado	Definir o componente como ativado.
<b>Name</b>	Luz_Sensor	Definir o nome do componente.

- Clique no componente **Fingerprint** e insira na área de **VIEWER**.

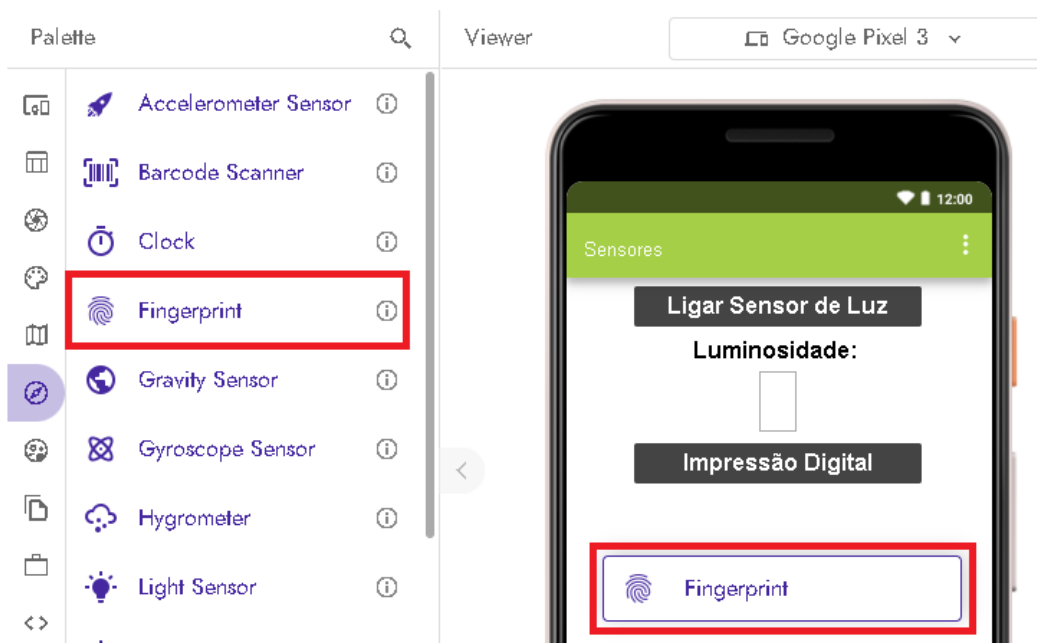
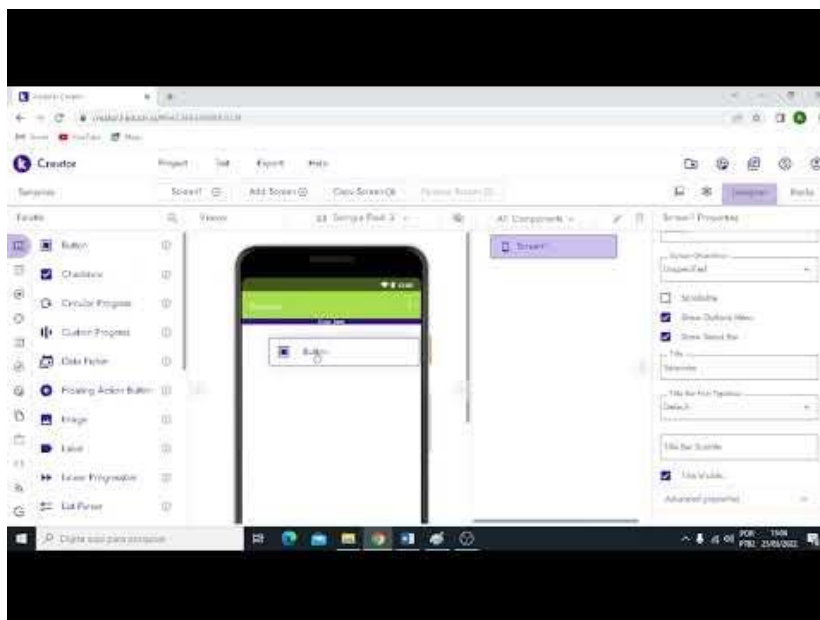


Figura 9- Inserindo o componente Fingerprint

- Altere as propriedades do objeto **Fingerprint Sensor**.

Propriedade	Valor	Função
<b>Enabled</b>	✓ Marcado	Definir o componente como ativado.
<b>Name</b>	Finger_Sensor	Definir o nome do componente.

Assista ao vídeo **Agenda 07 – Criando a interface Sensores**, disponível em: <https://youtu.be/7mjViEALjiA>



Após a construção do layout do aplicativo, vamos alterar para a opção de programação em blocos (Blocks).

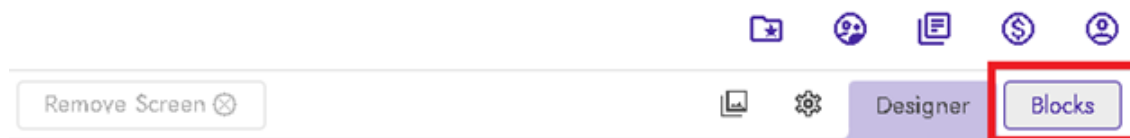


Figura 10 - Alterando a programação em blocos

Vamos construir o código de acordo com a **Figura 11**, para que o aplicativo possa realizar todas as funções necessárias.

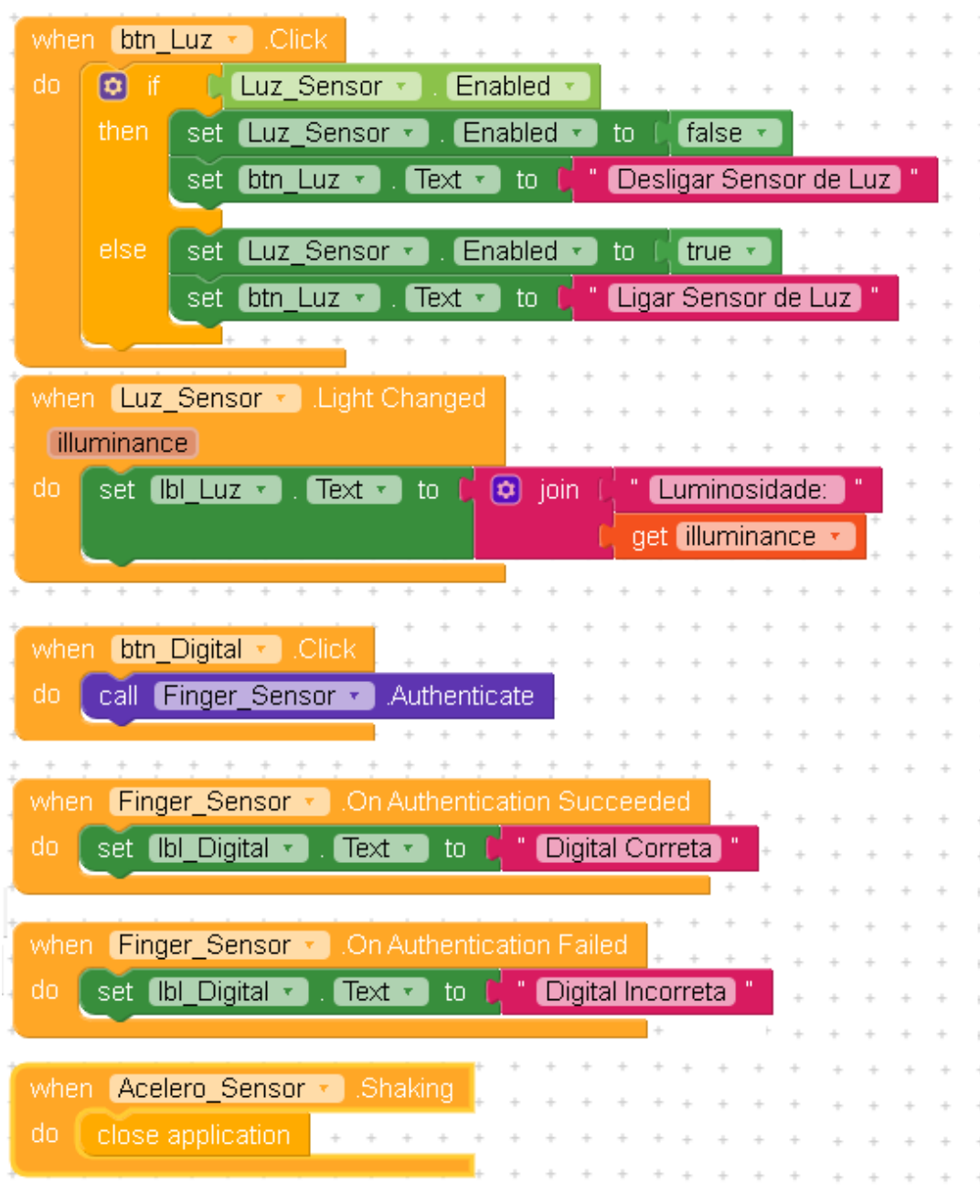
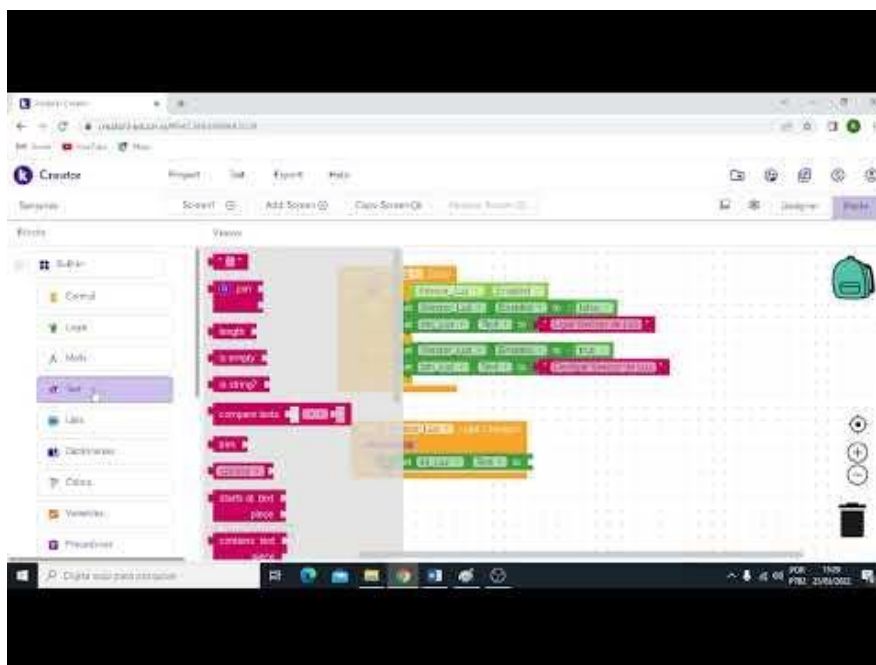


Figura 11 - Código de Blocos de Programação

Reveja todos os passos através do vídeo **Agenda 07 – Criando a programação em blocos Sensores**, disponível em: <https://youtu.be/uuyHxX8j9BM>



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

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

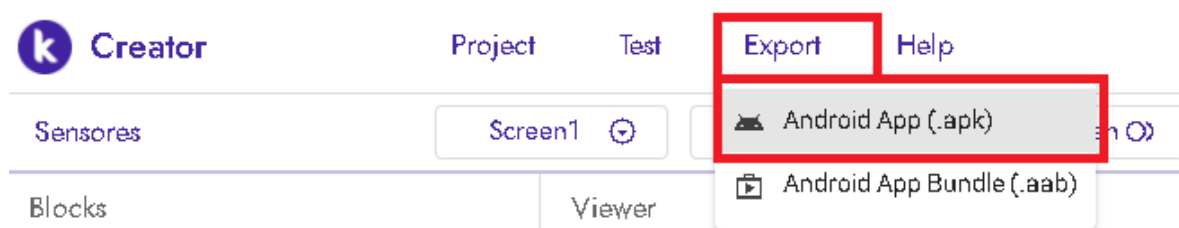


Figura 12 - Menu Export, opção Android App (.apk)

- 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 "Sensores"

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.

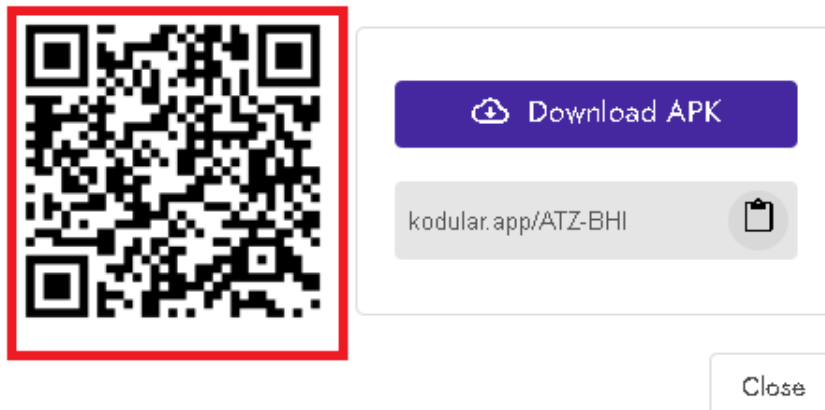


Figura 13 - Disponibilizando o Qrcode.

Ao iniciar o aplicativo, o usuário poder girar o dispositivo móvel, que a função do componente **Acelerômetro** irá fechar o aplicativo.