



**Faculdade Barretos**

[www.faculdadebarretos.com.br](http://www.faculdadebarretos.com.br)



Desenvolvendo aplicativos Android  
facilmente com o MIT App Inventor



**MIT**  
APP INVENTOR

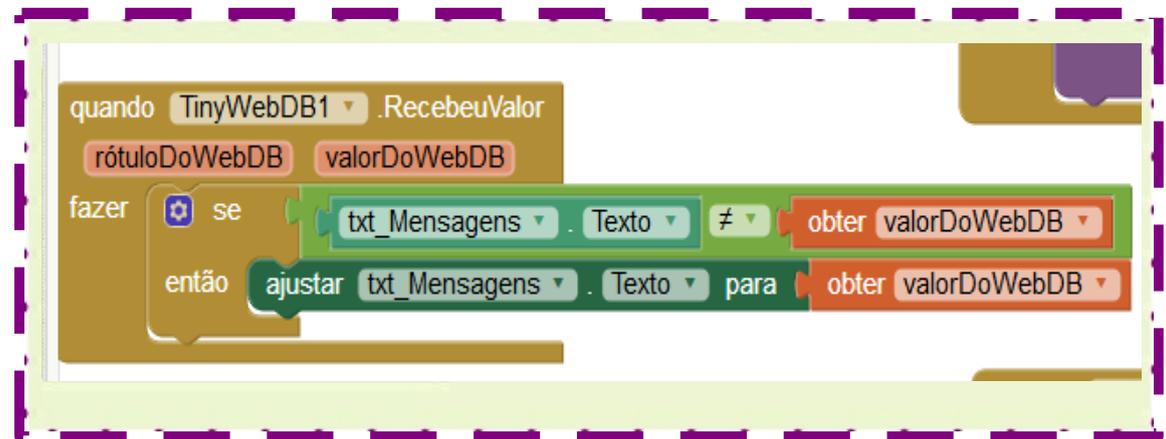
Prof. Me. Sérgio C. Portari Jr



O que é o App Inventor ?



Um ambiente de desenvolvimento na Web, originalmente criado no Google Labs, mas atualmente pertence ao MIT Labs. Tem como objetivo facilitar o desenvolvimento de Aplicativos para Android.



O Desenvolvimento de aplicativos Android requer um conhecimento da plataforma Java, do paradigma orientado a Objetos. Tornando difícil o desenvolvimento de um App por não desenvolvedores.



O App Inventor permite  
desenvolver aplicativos Android  
sem o conhecimento destes  
conceitos.





# Por que é tão fácil?

- **Nenhuma sintaxe.** A linguagem de blocos elimina a necessidade de lembrar e digitar código
- **Tudo está bem na frente de vocês.** Componentes e funções estão em gavetas. Basta encontrar, arrastar e soltar.
- **Componentes de alto nível.** O app inventor tem uma grande biblioteca.
- **Plug-in.** Você não pode fazer coisas que não fazem sentido.
- **Concreto.** Menos abstrato do que muitas linguagens.
- **Na Web,** caso tenha um celular Android, não vai precisar instalar nada no seu computador.

# O que podemos criar?



- Aplicativos educacionais
  - Com vídeos, imagens, texto-para-voz
- Aplicativos de localização
- Jogos
- Controladores de Robô
- Aplicativos para a Web
- Aplicativos pessoais



# O que não podemos criar?

- App Inventor não é um substituto para a linguagem de programação de aplicativos para o Android.
- Não é uma ferramenta para criar jogos e aplicativos **comerciais**.
- Não é possível criar jogos como Angry Birds ou aplicativos complexos como o Firefox Mobile, e as aplicações criadas estão **longe de ser bonitas**. Elas são, no entanto, funcionais.



OK. E como começo ?



# Primeiro projeto



1. Crie uma conta de e-mail do gmail (utilize o navegador Firefox ou Chrome).
2. Entre na sua conta do gmail.
3. Adicione uma nova aba no navegador.
4. Digite o seguinte endereço:
  - <http://appinventor.mit.edu/explore/>

# Primeiro projeto

Clique em



About ▾

News & Events ▾

Resources ▾

Create apps!

*Anyone Can Build Apps That Impact the World*

Google Custom



Donate!

MassTLC Distinguished Leadership Award 2017 - MIT App Inve...



Active Users: This Month: **988.9K** This Week: **308.0K** Today: **80.6K**

Registered Users: **6.8M** Countries: **195** Apps Built: **24.0M**



Introducing  
App Building Guides!

Check out our first App Building Guide developed by MIT App Inventor and Youth Radio.

Start today with the [Bootleg Snapchat App Guide](#).

Learn more about [Youth Radio](#).

Tweets by @MITAppInventor

MIT App Inventor Retweeted

**Susan Malone**  
@susan\_waym

Calling @TeachForAmerica &



Você pode acessar o App Inventor diretamente, navegando ao endereço <http://ai2.appinventor.mit.edu/>



# Primeiro projeto



Selecione  
seu e-mail

→ [https://appengine.google.com/\\_ah/loginform?state=AJKiYcEcpISAtsxoYINpOaqHPiVCHKJy\\_W-dDEVcfBRf62Aux3sS4iw8](https://appengine.google.com/_ah/loginform?state=AJKiYcEcpISAtsxoYINpOaqHPiVCHKJy_W-dDEVcfBRf62Aux3sS4iw8)

**Google** accounts

The application **MIT AppInventor Version 2** is requesting permission to access your Google Account.

Please select an account that you would like to use.

- carlafaria@ifma.edu.br
- etec@ifma.edu.br
- carlaslz@gmail.com
- redeetecbrasil@ifma.edu.br

Google is not affiliated with the contents of **MIT AppInventor Version 2** or its owners. If you sign in, Google will share your email address with **MIT** other personal information.

Allow

No thanks

[Sign in to another account](#)

Clique em  
"Allow"

©2011 Google - [Google Home](#) - [Terms of Service](#) - [Privacy Policy](#) - [Help](#)

# Primeiro projeto



**Welcome to App Inventor!**

Welcome to MIT App Inventor 2

**New Release (done)**  
**May 26, 2014: DatePicker Incompatibility**  
[More Information](#)

---

Got an Android phone or tablet? Find out how to [Set up and connect an Android device.](#)

Don't have an Android device? Find out how to [Set up and run the Android emulator.](#)

(Emulator and USB connections are currently for Mac and Windows only. Support for Linux is *coming soon!*)

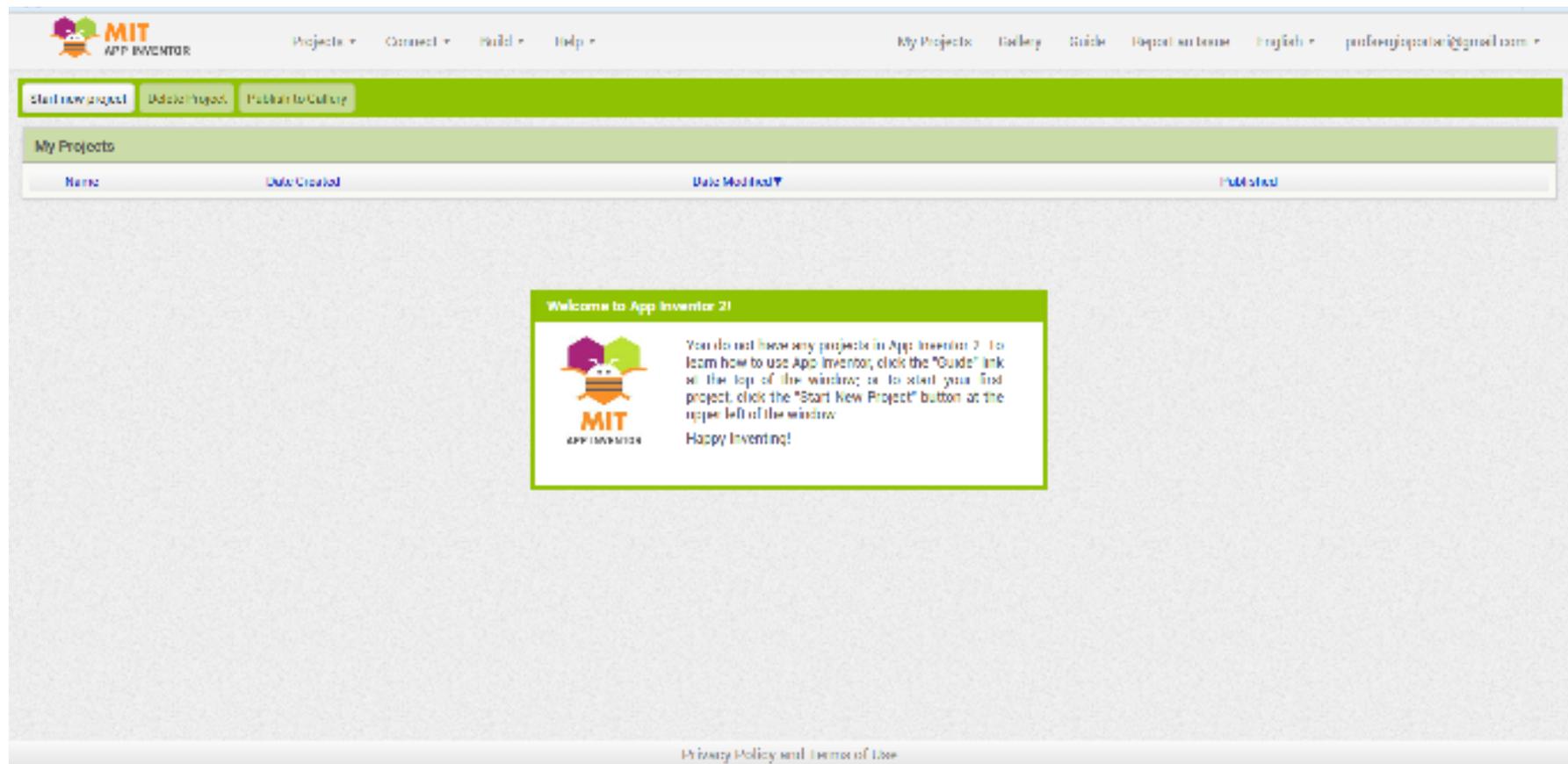
(Support for Internet Explorer is *coming soon!*)

Continue

Clique em  
"Continue"

# Primeiro projeto

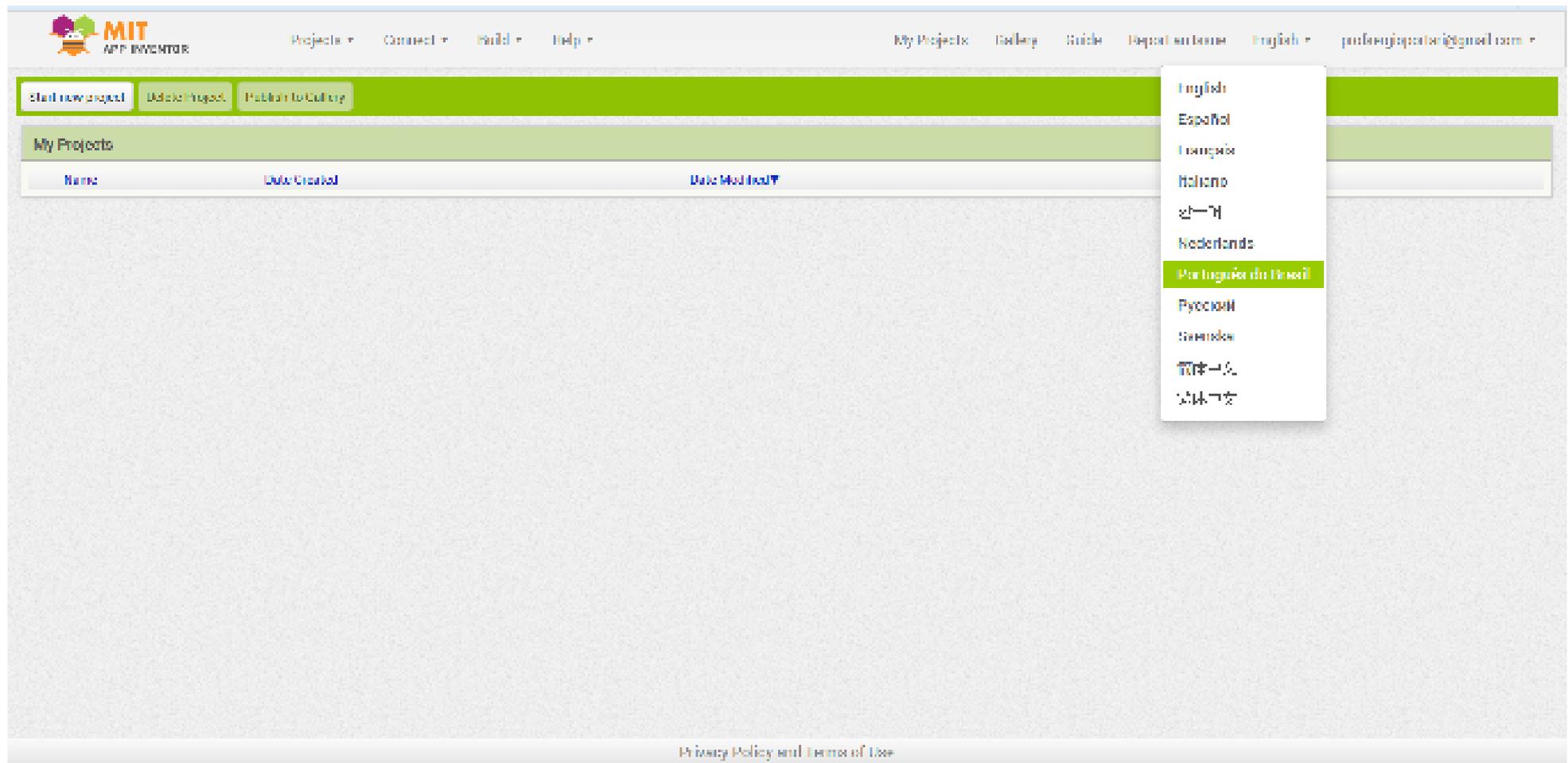
Se for a primeira vez que usa o App Inventor, você verá a página de Projetos



The screenshot displays the MIT App Inventor web interface. At the top, there is a navigation bar with the MIT App Inventor logo and links for Projects, Connect, Build, and Help. On the right side of the header, there are links for My Projects, Gallery, Guide, Report an Issue, English, and a user profile link. Below the header, there is a green bar with three buttons: Start new project, Manage Projects, and Publish to Gallery. The main content area features a 'My Projects' section with a table that has columns for Name, Date Created, Date Modified, and Published. The table is currently empty. In the center of the page, there is a green-bordered box titled 'Welcome to App Inventor 2!' containing the MIT App Inventor logo and a message: 'You do not have any projects in App Inventor 2. To learn how to use App Inventor, click the "Guide" link at the top of the window; or to start your first project, click the "Start New Project" button at the top left of the window. Happy Inventing!'. At the bottom of the page, there is a link for Privacy Policy and Terms of Use.

# Primeiro projeto

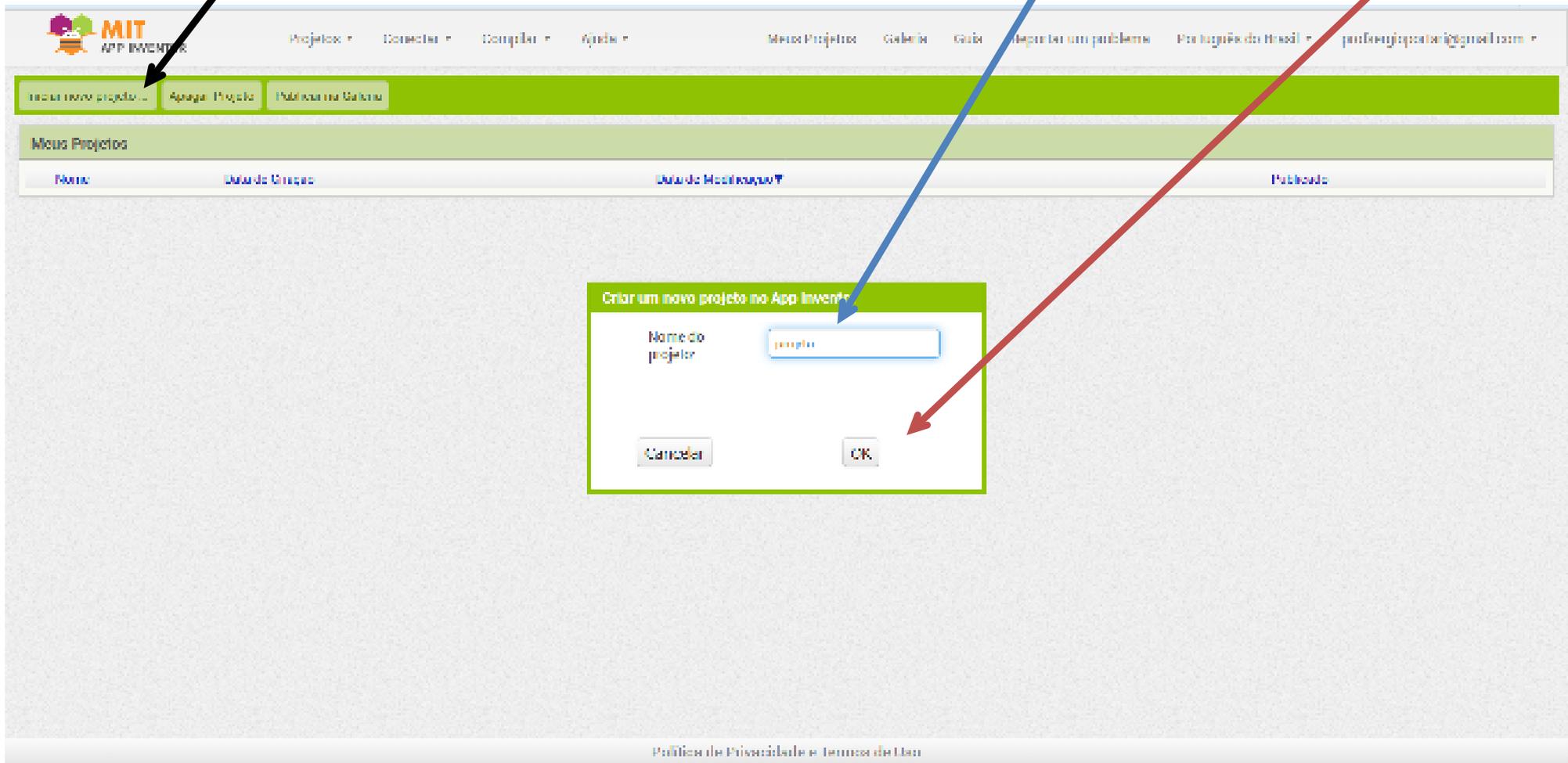
- Você pode traduzir para Português Brasil



The screenshot shows the MIT App Inventor web interface. At the top left is the MIT App Inventor logo. The navigation bar includes links for Projects, Connect, Build, Help, My Projects, Gallery, Guide, Report a Issue, English, and a user profile link. Below the navigation bar are three buttons: Start new project, Delete Project, and Publish to Gallery. The main content area is titled 'My Projects' and contains a table with columns for Name, Date Created, and Date Modified. A language dropdown menu is open on the right side of the page, listing various languages. 'Português do Brasil' is highlighted in green, indicating it is the selected language. Other visible languages include English, Español, Français, Italiano, 日本語, Nederlands, Pycckий, Svenska, 简体中文, and 繁體中文. At the bottom of the page, there is a link for 'Privacy Policy and Terms of Use'.

# Primeiro projeto

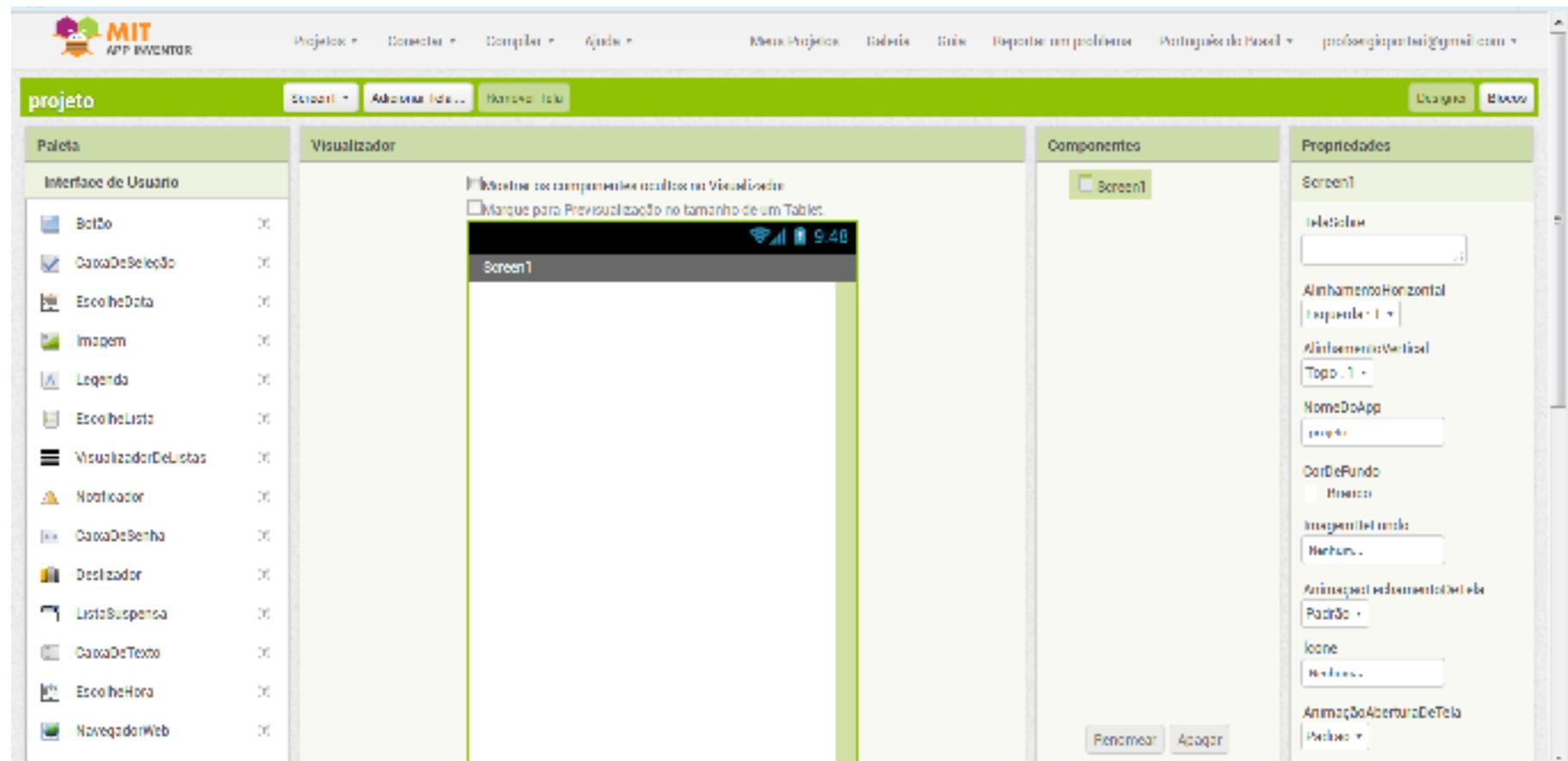
Clique em Iniciar Novo Projeto, dê um nome e aperte OK



The screenshot displays the MIT App Inventor web interface. At the top, the MIT App Inventor logo is on the left, and a navigation menu includes 'Projetos', 'Conectar', 'Compilar', 'Ajuda', 'Meus Projetos', 'Galeria', 'Guia', 'Reportar um problema', 'Português do Brasil', and 'professores@mitappinvento.org.br'. Below the navigation bar is a green header with three buttons: 'Iniciar novo projeto...', 'Apagar Projeto', and 'Publicar na Loja'. Underneath is a section titled 'Meus Projetos' with a table header containing 'Nome', 'Data de Criação', 'Data de Modificação', and 'Publicado'. A modal dialog titled 'Criar um novo projeto no App Inventor' is open in the center. It features a text input field for 'Nome do projeto' containing the text 'projeto', and two buttons at the bottom: 'Cancelar' and 'OK'. A black arrow points from the text 'Clique em Iniciar Novo Projeto' to the 'Iniciar novo projeto...' button. A blue arrow points from the text 'dê um nome' to the 'Nome do projeto' input field. A red arrow points from the text 'e aperte OK' to the 'OK' button. At the bottom of the page, there is a link for 'Política de Privacidade e Termos de Uso'.

# Primeiro projeto

- A primeira janela que se abre é o *Component Designer*



# Component designer

Dividida em 4 colunas

The screenshot displays the MIT App Inventor Component Designer interface, which is organized into four main columns:

- Paleta (Palette):** Located on the left, it contains a list of user interface components such as Botão, CaixaDeSeleção, EscolheData, Imagem, Legenda, EscolheLista, VisualizadorDeListas, Notificador, CaixaDeSenha, Deslizador, ListaSuspensa, CaixaDeTexto, EscolheHora, and NavegadorWeb.
- Visualizador (Visualizer):** The central area where the application is previewed. It shows a mobile device screen with a status bar at the top displaying signal strength, Wi-Fi, and battery levels (9.4B). Below the status bar, a component labeled "Screen1" is visible.
- Componentes (Components):** A panel on the right side of the visualizer, currently showing a single component named "Screen1".
- Propriedades (Properties):** The rightmost panel, which displays the configuration options for the selected "Screen1" component. These include fields for "Idetificador", "AlinhamentoHorizontal" (set to "Esquerda"), "AlinhamentoVertical" (set to "Topo"), "NomeDoApp", "CorDeFundo", "IntensidadeDoFundo", "AnimaçãoDeAberturaDeTela", "Ícone", and "AnimaçãoDeAberturaDeTela".

At the top of the interface, there is a menu bar with options like "Projetos", "Conectar", "Compilar", and "Ajuda". Below the menu bar, a green header bar shows the current project name "projeto" and tabs for "Screen1", "Adicione Tela...", and "Remova Tela".

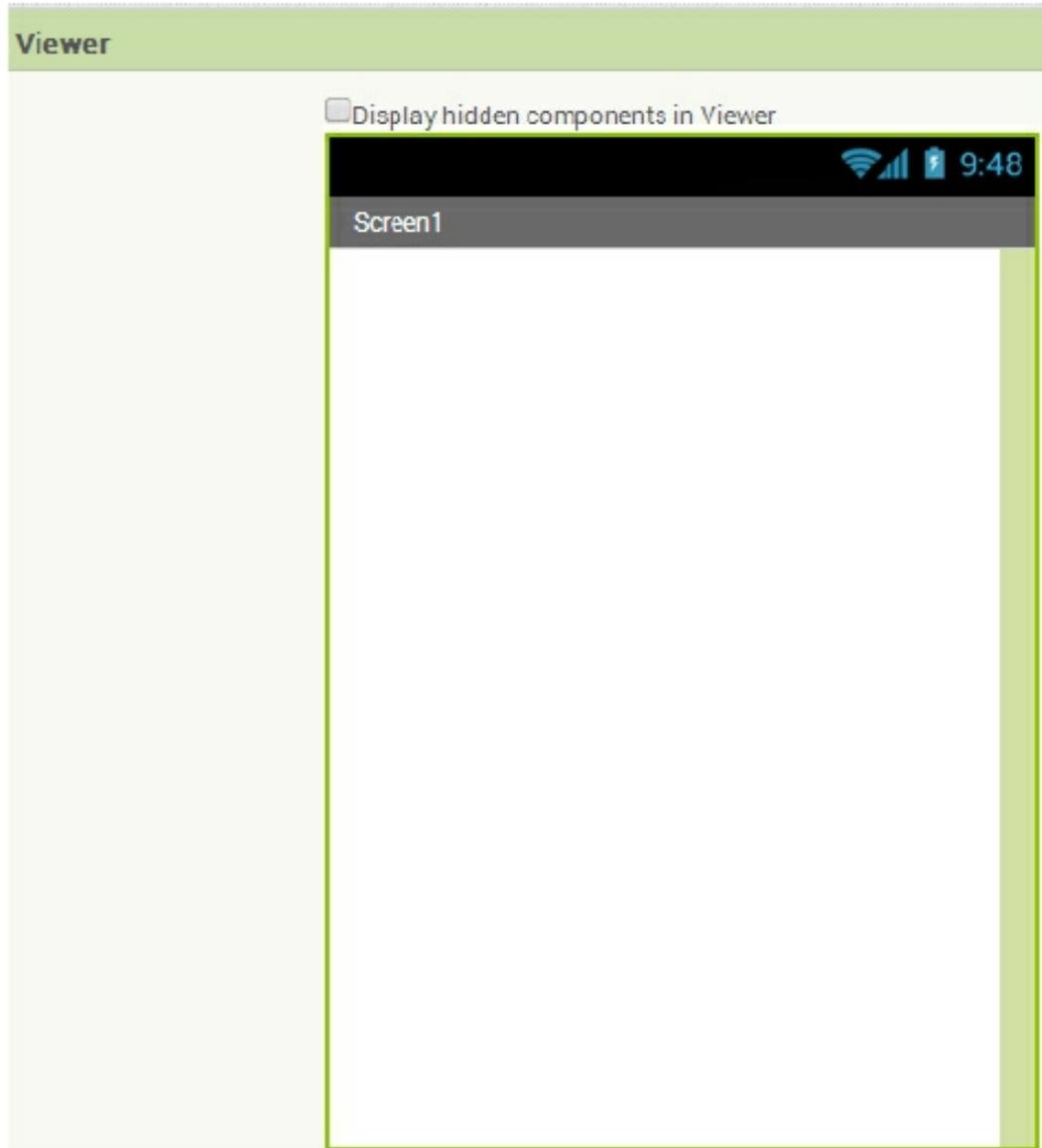
# Component designer - Viewer

The image shows the MIT App Inventor Component Designer interface. The top navigation bar includes the MIT App Inventor logo and menu items: "Projetos", "Conectar", "Compilar", "Ajuda", "Meus Projetos", "Galeria", "Sua", "Reportar um problema", "Português do Brasil", and "professor@portuguis@gmail.com". Below the navigation bar, the project name "projeto" is displayed, along with buttons for "Screen1", "Adicionar Tela...", and "Remover Tela", and "Design" and "Executar" buttons.

The interface is divided into four main panes:

- Paleta:** A list of UI components under the heading "Interface de Usuário". Components include Botão, CaixaDeSeleção, EscolheData, Imagem, Legenda, EscolheLista, VisualizadorDeListas, Notificador, CaixaDeSenha, Deslizador, ListaSuspensa, CaixaDeTexto, EscolheHora, and NavegadorWeb.
- Visualizador:** A central pane showing a preview of the application. It contains a header "Mostrar os componentes usados no Visualizador" and a checkbox "Marque para Previewização no tamanho de um Tablet." Below this is a mobile device simulator showing a screen titled "Screen1". This entire pane is highlighted with a red rectangular border.
- Componentes:** A pane showing the components currently used in the application, with a single component "Screen1" listed.
- Propriedades:** A pane showing the properties of the selected component, "Screen1". Properties include "TelaSele", "AlinhamentoHorizontal" (set to "Esquerda"), "AlinhamentoVertical" (set to "Topo"), "NomeDoApp" (set to "projeto"), "CorDeFundo" (set to "Branco"), "ImagemDeFundo" (set to "Nenhuma"), "AnimaçãoDeFundoDaTela" (set to "Padrão"), "Ícone" (set to "Nenhuma"), and "AnimaçãoAberturaDeTela" (set to "Padrão").

# Component designer - Viewer



O Viewer” (Visualizador) permite o usuário organizar cada um de seus objetos, montando o aplicativo como ele deve ser. Uma janela de exibição simula a tela de um smartphone com o sistema operacional Android.

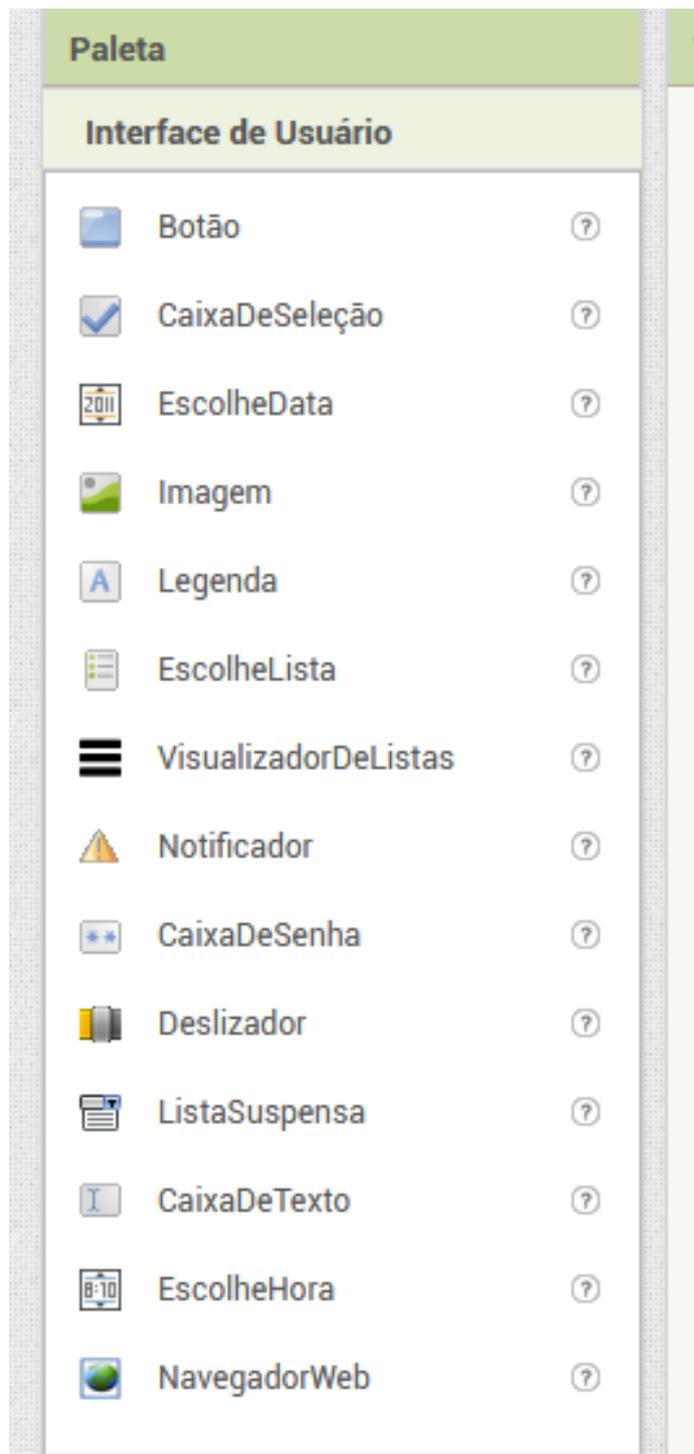
# Component designer - Palette

The image shows the MIT App Inventor Component Designer interface. The top navigation bar includes the MIT App Inventor logo and menu items: Projetos, Conectar, Compilar, Ajuda, Meus Projetos, Paleta, Guia, Reportar um problema, Português do Brasil, and professor@mit.edu@gmail.com. Below the navigation bar, the project name 'projeto' is displayed, along with tabs for 'Screen1', 'Adicionar tela...', and 'Remover tela', and buttons for 'Carregar' and 'Salvar'.

The main workspace is divided into four panels:

- Paleta:** A list of UI components for the 'Interface de Usuário' category, each with a '0x' count. The list includes: Botão, CaixaDeSeleção, EscolheData, Imagem, Legenda, EscolheLista, VisualizadorDeListas, Notificador, CaixaDeSenha, Deslizador, ListaSuspensa, CaixaDeTexto, EscolheHora, and NavegadorWeb. This panel is highlighted with a red border.
- Visualizador:** A preview window showing a mobile device screen with a status bar at the top displaying 'Screen1' and the time '9:48'. It includes checkboxes for 'Mostre os componentes ocultos no Visualizador' and 'Marque para Previsualização no tamanho de um Tablet'.
- Componentes:** A list of components currently on the screen, showing 'Screen1'.
- Propriedades:** A panel for configuring the selected component, 'Screen1'. It includes properties such as 'Relação', 'AlinhamentoHorizontal' (set to 'Esquerda'), 'AlinhamentoVertical' (set to 'Topo'), 'NomeDoApp' (set to 'projeto'), 'CorDeFundo' (set to 'Branco'), 'Inspecionelink' (set to 'Nenhuma'), 'AnimaçãodeAberturaDeTela' (set to 'Padrão'), and 'icone' (set to 'Nenhuma').

At the bottom of the Componentes and Propriedades panels, there are buttons for 'Remover' and 'Adicionar'.



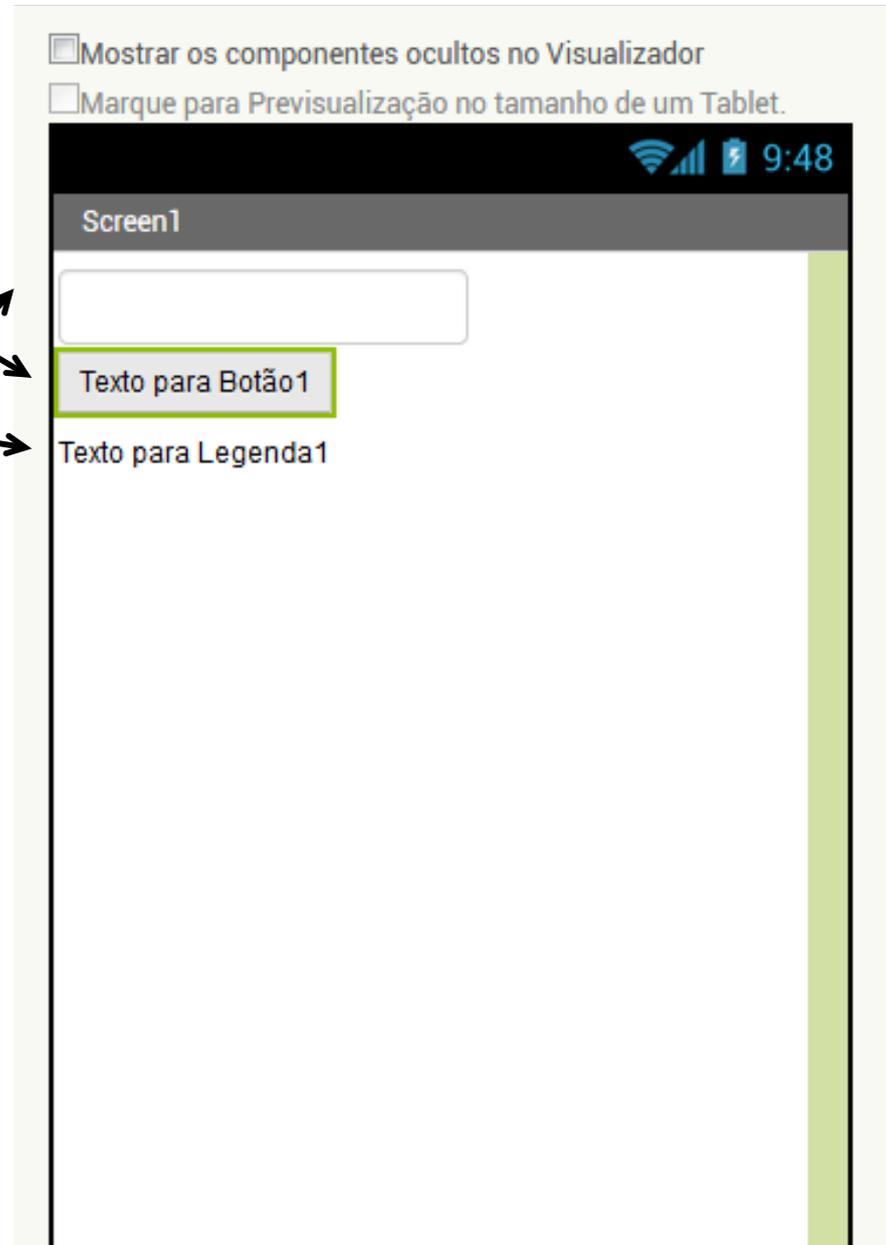
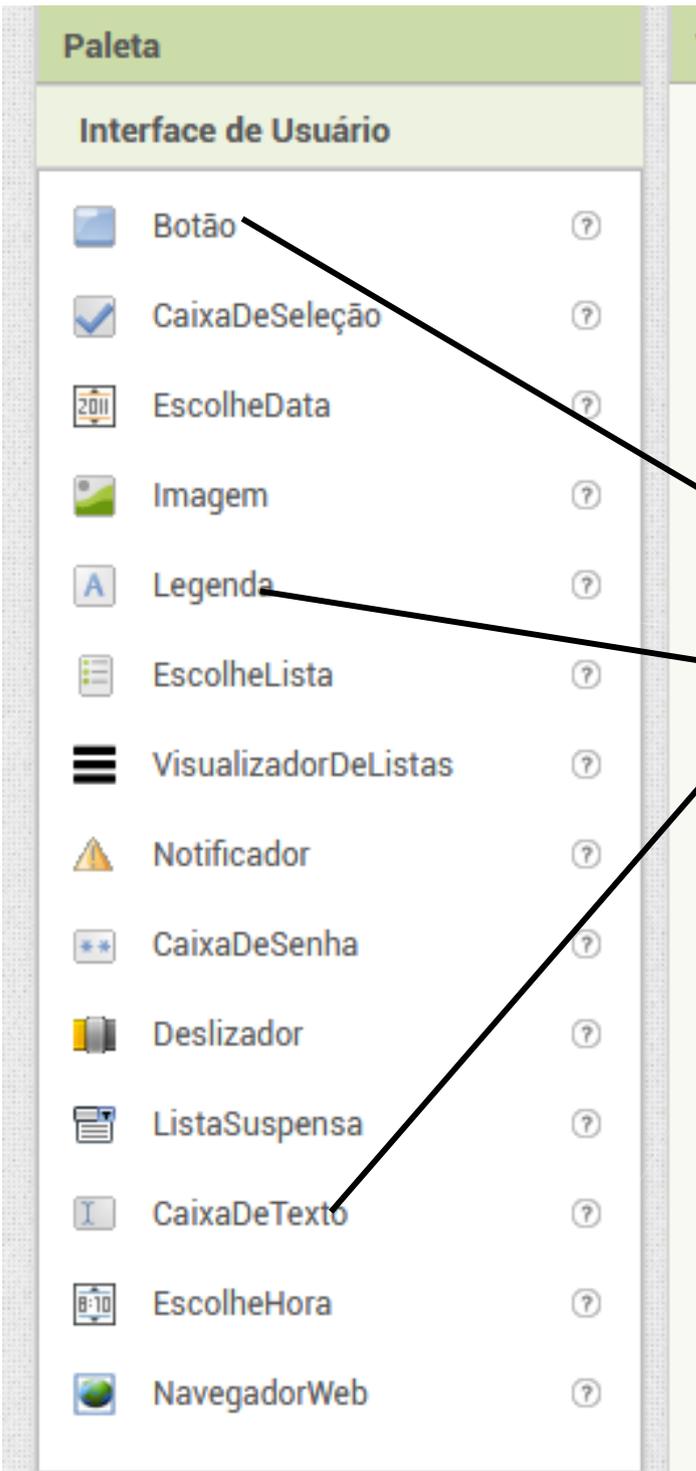
# Component designer - Palette

Na “Palette” (Paleta), é onde ficam todos os componentes utilizáveis num aplicativo. Esta paleta é dividida em seções para facilitar a localização dos componentes, que vão dos básicos (botões, imagens e textos)

Para utilizar um desses componentes basta clicar sobre ele e arrastar para cima da coluna, chamada “Viewer” (Visualizador).

Nem todos os componentes da paleta são visíveis, por exemplo os utilizados para manipular e acessar recursos do celular, como web, sensores ...

**Atividade:** Arraste os seguintes componentes para o visualizador: textBox, Button e Label

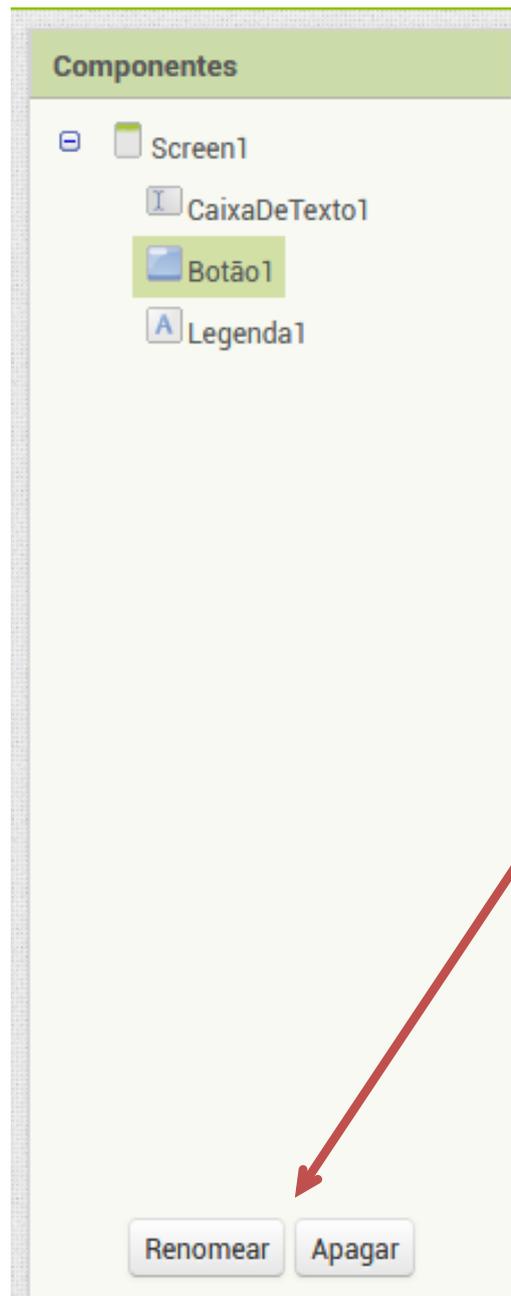


# Component designer - Components

The screenshot displays the MIT App Inventor Component Designer interface. The top navigation bar includes links for 'Projetos', 'Conectar', 'Compilar', 'Ajuda', 'Meus Projetos', 'Galeria', 'Sobre', 'Reportar um problema', 'Português do Brasil', and 'profregioportug@gmail.com'. The main workspace is divided into several panels:

- Paleta:** A list of user interface components such as Botão, CaixaDeSeleção, EscolheData, Imagem, Legenda, EscolheLista, VisualizadorDeListas, Notificador, CaixaDeSenha, Deslizador, ListaSuspensa, CaixaDeTexto, EscolheHora, and NavegadorWeb.
- Visualizador:** A preview window showing a mobile device screen with a form containing a text input field, a button labeled 'Texto para Botão1', and a label 'Texto para Legenda1'.
- Componentes:** A tree view of the components on the screen, including 'Screen1', 'CaixaDeTexto1', 'Botão1', and 'Legenda1'. This panel is highlighted with a red box.
- Propriedades:** A panel showing the properties of the selected component, 'Botão1', including 'CorDoFundo', 'Fonte', 'TamanhoDaFonte', 'FamíliaDaFonte', 'Alinhamento', 'Imagem', and 'Mensagem'.

# Component designer - Components



Nesta coluna ficam armazenados todos os itens adicionados, sejam eles visíveis ou não na tela do programa.

Feito a atividade anterior, teríamos três componentes, agrupados na tela (screen1).

Aqui podemos também renomear ou deletar os componentes

Clicar sobre qualquer um dos itens da lista na coluna "Components" permite que você possa editar seus detalhes na quarta coluna, chamada de "Properties" (**Propriedades**).

# Component designer - Properties

The screenshot displays the MIT App Inventor Component Designer interface. The top navigation bar includes the MIT App Inventor logo, project management options (Projetos, Conectar, Compilar, Ajuda), and user information (Meus Projetos, Galeria, Guia, Reportar um problema, Português do Brasil, profsergioportari@gmail.com). The main workspace is divided into four panels: Paleta (Component Palette), Visualizador (Previewer), Componentes (Component Hierarchy), and Propriedades (Properties). The Properties panel is highlighted with a red border and shows the configuration for a selected Button component. The interface is in Portuguese.

**Paleta**

Interface de Usuário

- Botão
- CaixaDeSeleção
- EscolheData
- Imagem
- Legenda
- EscolheLista
- VisualizadorDeListas
- Notificador
- CaixaDeSenha
- Deslizador
- ListaSuspensa
- CaixaDeTexto
- EscolheHora
- NavegadorWeb

**Visualizador**

Mostrar os componentes ocultos no Visualizador  
 Marque para Previsualização no tamanho de um Tablet.

Screen1

Texto para Botão1

Texto para Legenda1

**Componentes**

- Screen1
  - CaixaDeTexto1
  - Botão1
  - Legenda1

Renomear Apagar

**Propriedades**

Botão1

CorDeFundo

■ Padrão

Ativado

FonteNegrito

Fonteltálico

TamanhoDaFonte

14.0

FamiliaDaFonte

padrão

Altura

Automático...

Largura

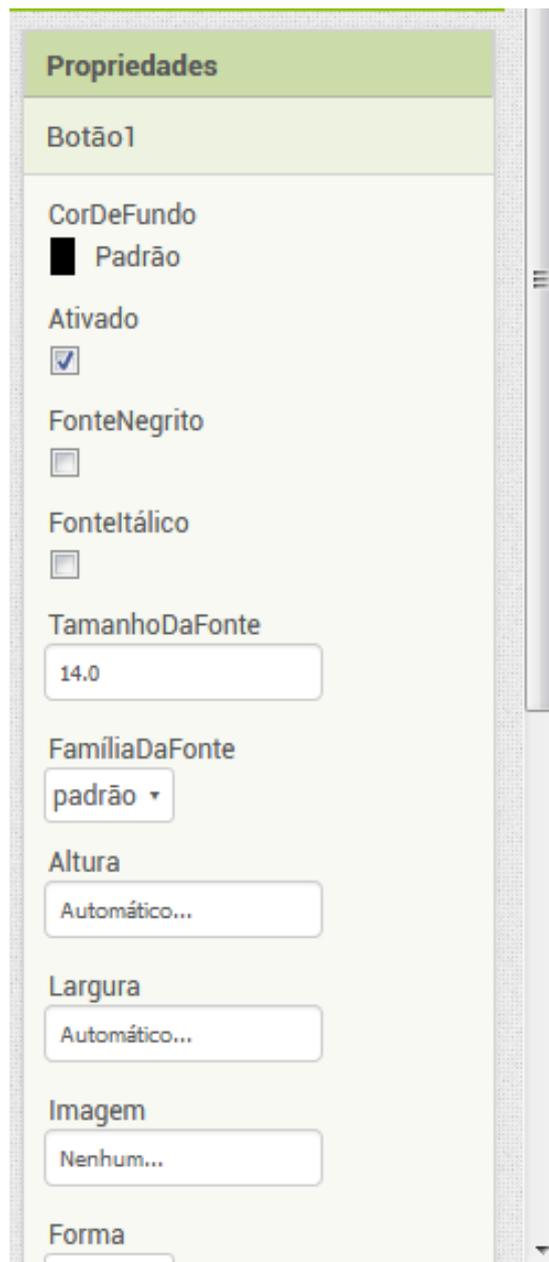
Automático...

Imagem

Nenhum...

Forma

# Component designer - Properties

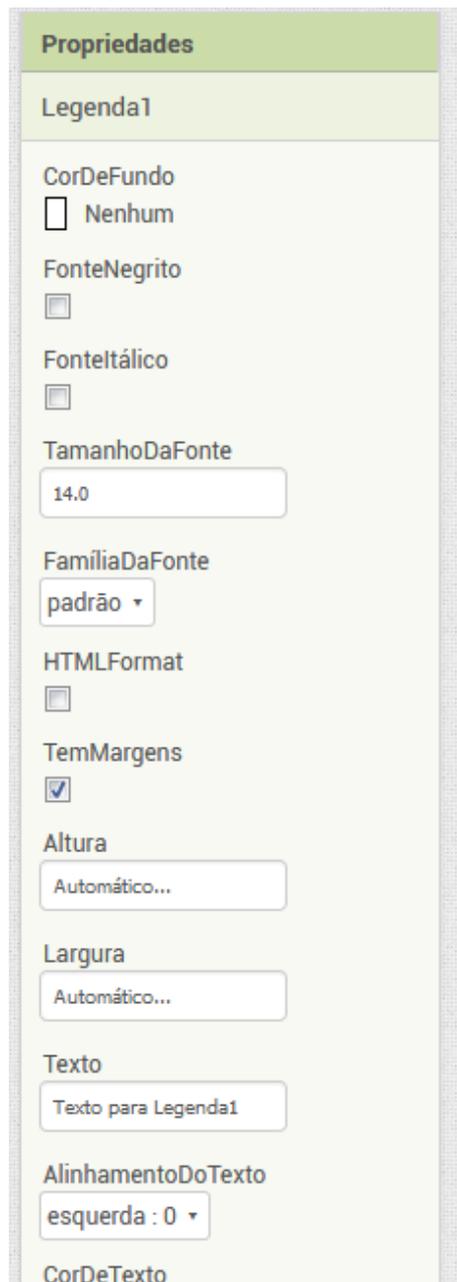


Propriedades são tipicamente associadas com componentes em um aplicativo, como Button, TextBox e Canvas .

Para um aplicativo, cada componente é completamente definido por um conjunto de propriedades

Ela permite definir os tamanhos e conteúdos dos textos de botões e caixas de informação, tamanho das imagens, cores de fundo e largura e altura de objetos.

# Component designer - Properties



**Atividade:** click no componente Label1 e atere suas seguintes propriedades:  
Aumente sua fonte (fontsize)  
Apague o texto na propriedade text.

# Propriedades e ações

As **propriedades** de diversos componentes de m aplicativo podem ser alteradas no momento da execução.

Para isso existe **ações** que mudam as propriedades dos objetos, como o texto, a cor, o tamanho ... .



Onde encontramos estas  
ações ?

Você encontrará as ações no block designer, é onde toda programação ocorre.



# Block Designer

Clique em  
"Blocks"

The screenshot shows the MIT App Inventor Block Designer interface. At the top left is the MIT App Inventor logo. The top navigation bar includes links for 'Projetos', 'Conectar', 'Compilar', 'Ajuda', 'Meus Projetos', 'Galeria', 'Inici', 'Reportar um problema', 'Introdução do Firebase', and 'problemas que não foram resolvidos por e-mail com'. Below the navigation bar is a green header for the current project, 'projeto', with tabs for 'Screen1', 'Adicionar Tela...', and 'Remover Tela'. On the right side of this header are 'Design' and 'Executar' buttons. The main workspace is divided into four panels: 'Paleta' (containing various UI components like Botão, CaixaDeSeleção, EscolheData, Imagem, Legenda, EscolheLista, VisualizadorDeListas, Notificador, CaixaDeSenha, Deslizador, ListaSuspensa, CaixaDeTexto, EscolheHora, and NavegadorWeb), 'Visualizador' (showing a preview of the app on a mobile device with 'Screen1' visible), 'Componentes' (showing a list of components, currently 'Screen1'), and 'Propriedades' (showing configuration options for 'Screen1', such as 'Título', 'AlinhamentoHorizontal', 'AlinhamentoVertical', 'NomeDoApp', 'CorDeFundo', 'ImagemDeFundo', 'AnimaçãoDeAberturaDeTela', 'Ícone', and 'AnimaçãoAberturaDeTela'). A red speech bubble points to the 'Blocks' button in the top right corner of the interface.

# Block Designer

Dividida em 2 colunas

The screenshot displays the MIT App Inventor Block Designer interface. At the top left is the MIT App Inventor logo. The top navigation bar includes links for 'Projetos', 'Conectar', 'Compilar', 'Ajuda', 'Meus Projetos', 'Galeria', 'Tela', 'Reportar um problema', 'Português do Brasil', and a user profile 'professor@ptbr@gmail.com'. Below this is a green header bar with the project name 'projeto' and buttons for 'Screen1', 'Adicionar tela...', and 'Remover tela'. On the right of this bar are 'Carregar' and 'Salvar' buttons. The main workspace is divided into two columns. The left column, titled 'Blocos', contains a tree view of components: 'Internos' (Control, Triggers, Matemática, Texto, Listas, Dados, Variáveis, Procedimentos), 'Screen1' (On Click, On Focus, On Scroll, On Tap), and 'Qualquer componente'. At the bottom of this column are 'Renomear' and 'Apagar' buttons. The right column, titled 'Visualizador', shows a large empty canvas with a teal backpack icon in the top right corner. On the right side of the canvas are zoom controls (a gear icon, '+', and '-') and a trash can icon. At the bottom left of the canvas, there are two status indicators (a yellow triangle with '0' and a red circle with '0') and a 'Mudar as Aparências' button.

# Block Designer - Blocks



Projetos + Conectar + Compilar + Ajuda + Meus Projetos + Galeria + Guia + Reportar um problema + Português do Brasil + professor@portel@gmail.com +

Screen1 + Adicionar tela... + Renomear tela + **Carregar** + **Excluir**

Blocos

Inferno

- Controle
- Lógica
- Matemática
- Texto
- Listas
- Classe
- Variáveis
- Procedimentos

Screen1

- Canvas1
- Botão1
- Legenda1

Outros componentes

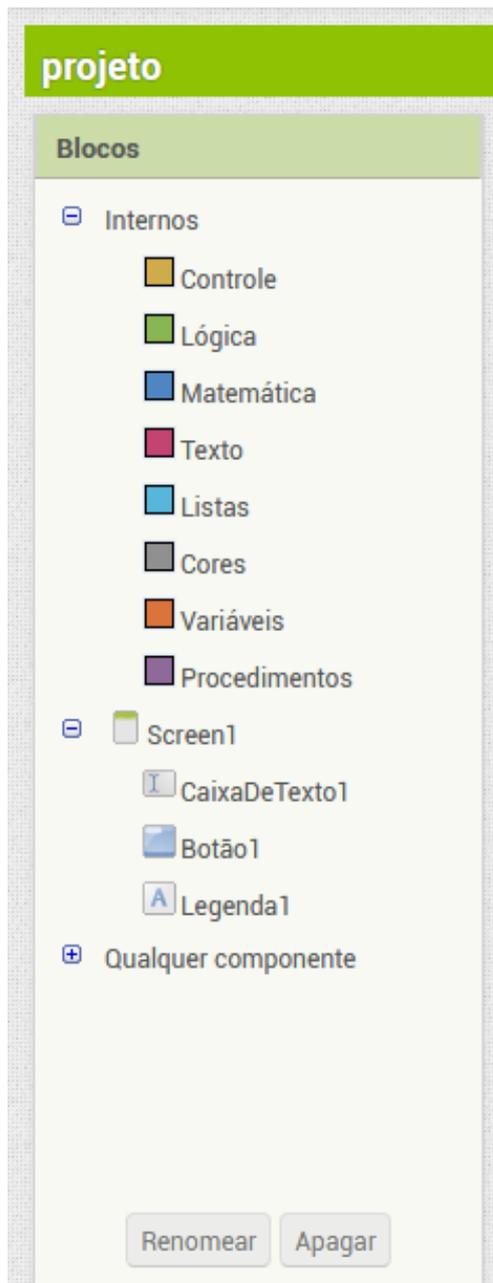
Renomear Apagar

Visualizador



0 0  
Mostrar Avisos

# Block Designer - Blocks



Aqui estão dispostos os diversos blocos utilizados para a programação. Em “Built-in” (Internos) blocos para comandos mais gerais, como operações lógicas, matemáticas e de controle. Em “Screen1” (Tela1) tem todos os objetos que você inseriu em seu programa. As ações para cada objeto serão encontradas aqui.

# Block Designer - Blocks

The image shows the Block Designer interface. On the left, under the 'Blocos' tab, there is a list of internal blocks categorized into 'Internos' and 'Qualquer componente'. The 'Internos' category includes: Controle, Lógica, Matemática, Texto, Listas, Cores, Variáveis, and Procedimentos. The 'Qualquer componente' category includes: Screen1, CaixaDeTexto1, Botão1, and Legenda1. At the bottom of the list are 'Renomear' and 'Apagar' buttons. The main area, labeled 'Visualizador', displays a sequence of logic blocks for a component named 'Legenda1'. The blocks are: 1. 'Legenda1' . CorDeFundo (property block); 2. 'ajustar Legenda1' . CorDeFundo para (action block); 3. 'Legenda1' . TamanhoDaFonte (property block); 4. 'ajustar Legenda1' . TamanhoDaFonte para (action block); 5. 'Legenda1' . TemMargens (property block); 6. 'ajustar Legenda1' . TemMargens para (action block); 7. 'Legenda1' . Altura (property block); 8. 'ajustar Legenda1' . Altura para (action block); 9. 'ajustar Legenda1' . PercentualDeAltura para (action block); 10. 'Legenda1' . Texto (property block); 11. 'Mostrar Avisos' (disabled block); 12. 'ajustar Legenda1' . Texto para (action block).

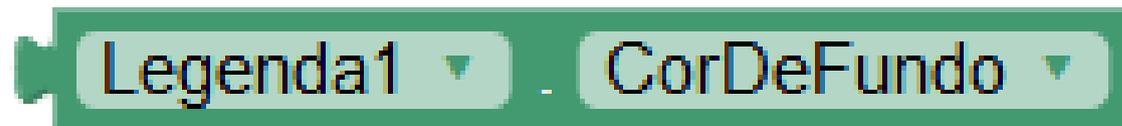
Ação que altera o tamanho da font de Label1

Ação que altera o texto mostrado em label1

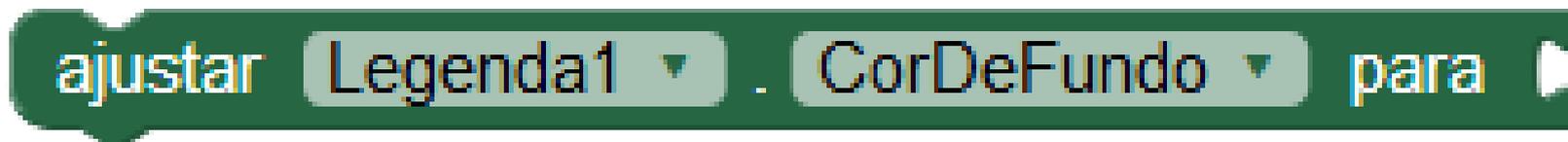
# Block Designer - Blocks

Em computação é comum chamar as ações que retorna uma propriedade de getters, e setters as ações que alteram uma propriedade.

Exemplo de um *getter*



Exemplo de um *setter*





Ok. Mas como estas ações  
são inicializadas ?

Além das ações, os objetos tem também eventos. Um aplicativo basicamente executa ações quando ocorre algum evento.



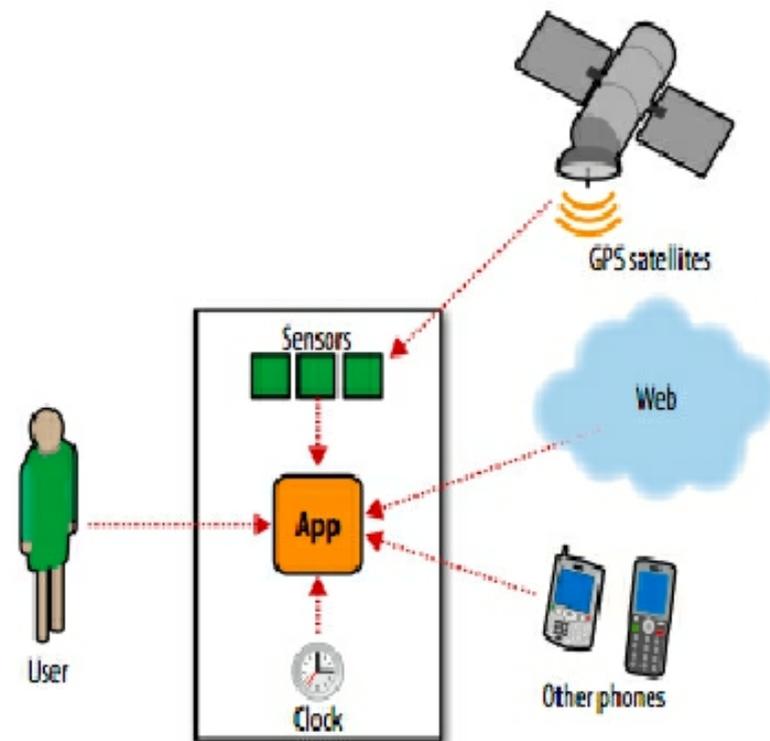
# Eventos

- O comportamento de um aplicativo é definido por um conjunto de manipuladores de eventos
- Cada manipulador de eventos executa funções específicas em resposta a um acontecimento particular

# Eventos

Um click em um botão é um exemplo de um evento iniciado pelo usuário. Contudo, existem outros tipos de eventos:

1. Timer events,
2. Sensor events,
3. Phone events,
4. Animation events,
5. Web events,
6. App (Screen) launch events.



# Conceito Chave

eventos + ações

Quando ocorrer um determinado evento, execute uma dada ação.

# Eventos + ações

## Exemplo de eventos

quando CaixaDeTexto1 .RecebeuFoco  
fazer

quando Screen1 .VoltarPressionado  
fazer

quando Botão1 .RecebeuFoco  
fazer

quando Screen1 .Inicializar  
fazer

## Exemplo de ações

Legenda1 . CorDeFundo

ajustar Legenda1 . CorDeFundo para



Ok. Já posso tentar  
começar a programar.

App é bem simples. O usuário vai digitar o seu nome (ex. Joao) na text box, clicar no botão e então será mostrado a seguinte mensagem: Ola Joao.



# Voltando ao editor de blocos ...

The screenshot displays a software interface for a project named "Aula01". At the top, a green header bar contains the project name "Aula01" on the left, a dropdown menu set to "Screen1", and two buttons: "Add Screen ..." and "Remove Screen". On the right side of the header, there are two tabs: "Designer" (which is active) and "Blocks".

The main workspace is divided into two sections:

- Blocks:** A vertical panel on the left side containing a tree view of components. It is organized into three categories:
  - Built-in:** Includes "Control" (orange square), "Logic" (green square), "Math" (blue square), "Text" (red square), "Lists" (light blue square), "Colors" (grey square), "Variables" (orange square), and "Procedures" (purple square).
  - Screen1:** A sub-category containing "TextBox1" (text input icon), "Button1" (blue button icon), and "Label1" (text icon).
  - Any component:** A catch-all category.
- Viewer:** A large, empty white rectangular area on the right, intended for visualizing the blocks. At the bottom left of this area, there are two warning indicators: a yellow triangle with "0" and a red triangle with "0", followed by a "Show Warnings" button. At the bottom right, there is a green trash can icon.

# Programando o comportamento

projeto

Screen1 Adicionar Tela ... Remover Tela

Blocos

Internos

- Controle
- Lógica
- Matemática
- Texto
- Listas
- Cores
- Variáveis
- Procedimentos

Screen1

- CaixaDeTexto1
- Botão1
- Legenda1

Qualquer componente

Renomear Apagar

Visualizador

quando Botão1 .Clique

fazer

quando Botão1 .RecebeuFoco

fazer

quando Botão1 .CliqueLongo

fazer

quando Botão1 .PerdeuFoco

fazer

quando Botão1 .ToqueParaBaixo

fazer

quando Botão1 .ToqueParaCima

fazer

1. Seleciona o componente Button1.
2. Arraste o evento "When Button1. click" para o visualizador de blocos.

# Programando o comportamento



3. Seleciona o componente Label.
4. Arraste a ação "set Label1.text" para o visualizador de blocos dentro do evento anterior.

Você precisar ter os blocos agrupados da seguinte maneira.



quando Botão1 .Clique

fazer ajustar Legenda1 . Texto para

A mensagem a ser exibida em label1 deve ser Ola + o nome digitado.



Ou seja, precisaremos  
juntar estes dois textos.



Existem o grupo de blocos para tratar informações textuais. Precisaremos de algo para juntar os textos.



Só pode ser esse bloco  
Join. Ao apontar para ele  
mostrou a seguinte  
mensagem.



A screenshot of a software interface, likely a visual programming environment. It features a central workspace with several colored blocks. A purple block with a gear icon and the text 'juntar' is highlighted with a yellow border. To its right, a brown block contains the text 'quando Botão1' and a dropdown menu. Below these, a green block is partially visible. A yellow tooltip box is overlaid on the 'juntar' block, containing the text: 'Justapõe todas entradas para formar uma única cadeia de texto. Se não houver entradas, forma uma cadeia de texto vazia.' Other blocks in the background include a purple block with a video camera icon and a purple block with the text 'comprimento'.

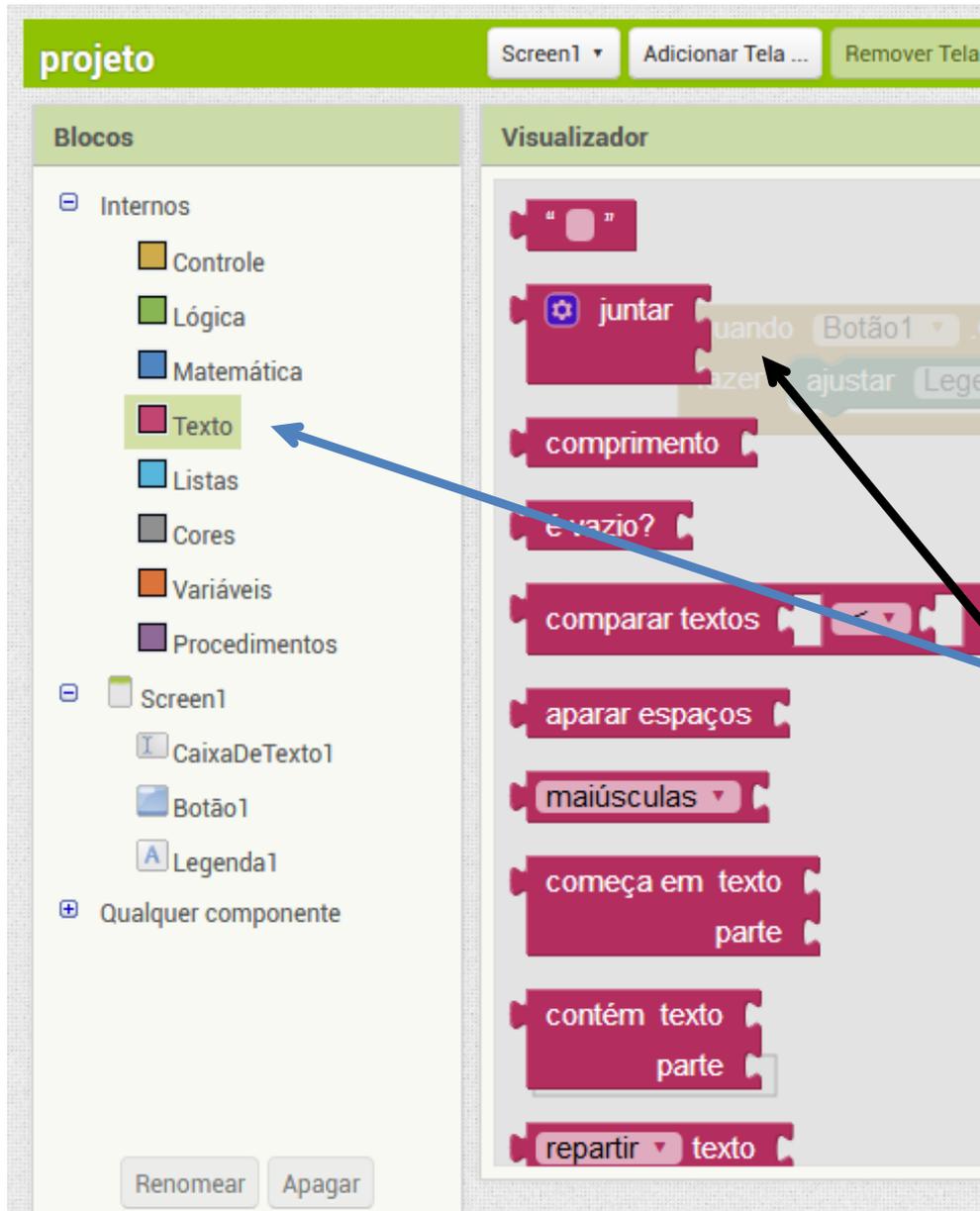
Isso mesmo, esta é uma boa  
forma para explorar e  
conhecer o App Inventor.



# Anatomia de um bloco



# Programando o comportamento



5. Seleciona o grupo de blocos Text.

6. Arraste a ação Join para o visualizador de blocos. A saída da Join será a entrada do "Set Label1. Text".

Você precisar ter os blocos agrupados da seguinte maneira.



```
quando Botão1 .Clique  
fazer  
  ajustar Legenda1 . Texto para [configurar] juntar
```

# Programando o comportamento

The image shows a programming environment interface with two main panels: 'Blocks' and 'Viewer'.

**Blocks Panel:**

- Built-in
  - Control
  - Logic
  - Math
  - Text** (highlighted)
  - Lists
  - Colors
  - Variables
  - Procedures
- Screen1
  - TextBox1
  - Button1
  - Label1
- Any component

**Viewer Panel:**

- “ ” (highlighted with a red arrow)
- join
- length
- is empty
- compare texts
- trim
- upcase
- starts at text

7. Selecciona o grupo de blocos Text.

8. Arraste o bloco que cria um texto.

9. Escreva a mensagem “Ola “

Você precisar ter os blocos agrupados da seguinte maneira.



```
quando Botão1 .Clique
fazer
  ajustar Legenda1 . Texto para
  juntar " Ola "
```

# Programando o comportamento

10 . Selecciona o componente  
TextBox

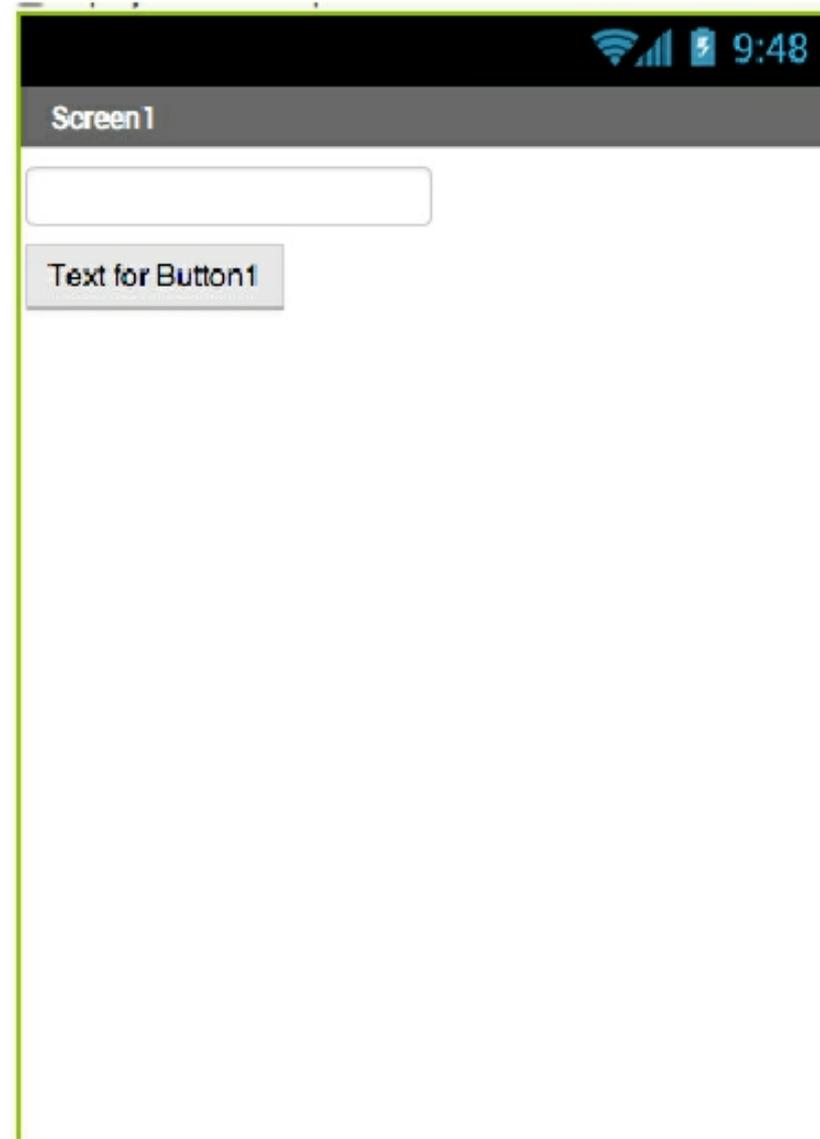
11. Arraste o bloco que  
retorna o texto para dentro  
do visualizado, como  
entrada do bloco Join.

Por fim. Você precisar ter os blocos agrupados da seguinte maneira.



```
quando Botão1 .Clique  
fazer  
  ajustar Legenda1 . Texto para  
  juntar " Ola "  
  CaixaDeTexto1 . Texto
```

Relembrando a tela do  
nosso aplicativo. Vamos  
testar ?



Instalem o MIT AI2 Companion em seu smartphone



**MIT AI2 Companion**  
MIT Center for Mobile Learning - 5 de julho de 2014  
Educação

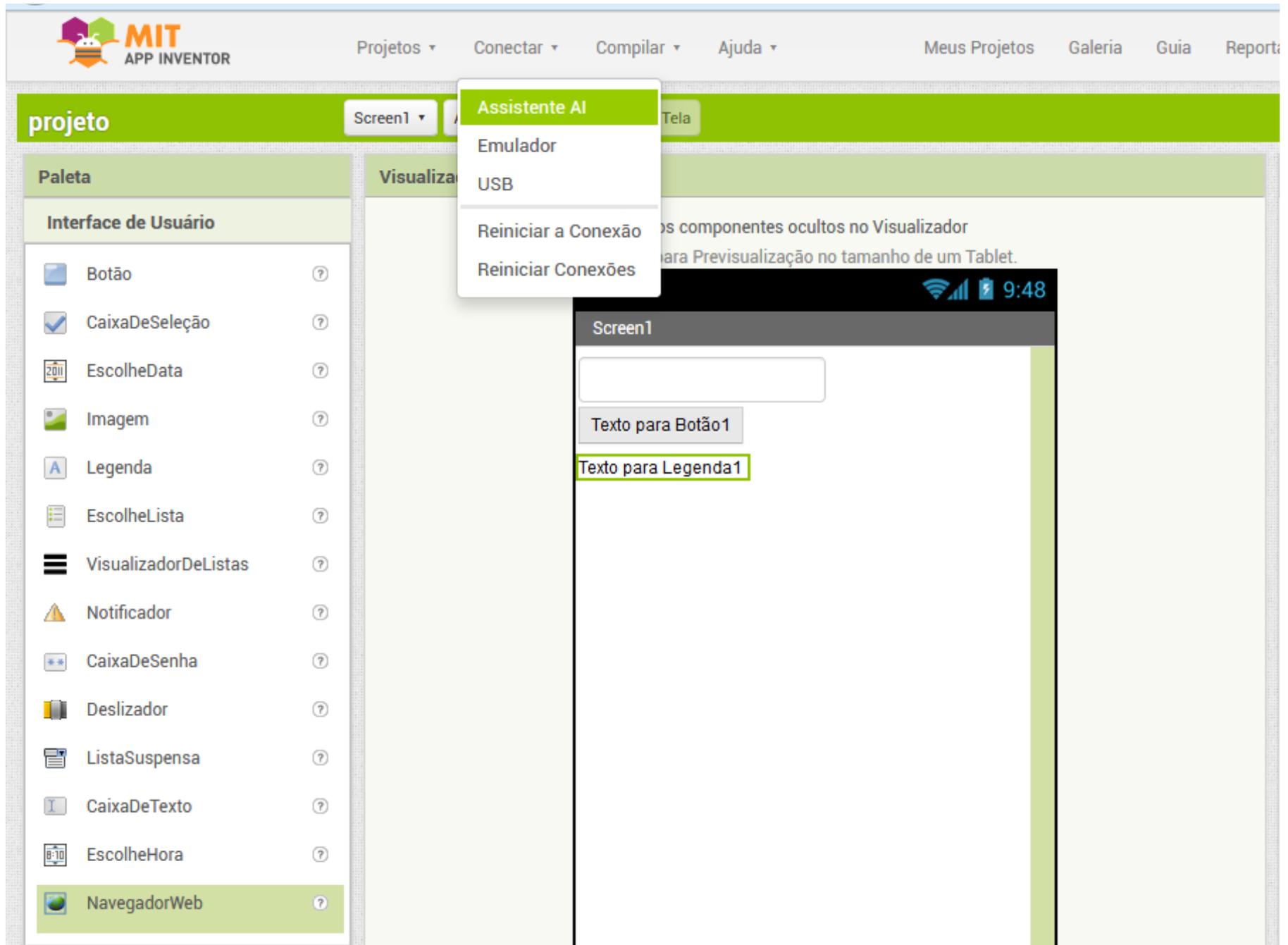
**Instalado**

Este aplicativo é compatível com todos os seus dispositivos.

★★★★☆ (4.530) [+665](#) [Recomende isto no Google](#)

The screenshot shows three mobile device screens displaying the MIT App Inventor 2 Companion interface. The first screen shows the main menu with options to 'connect with code' and 'scan QR code', along with the user's IP address (192.168.1.107) and version (2.11a2ex1). The second screen shows the same interface but with a different IP address (128.31.24.195) and version (2.05a7ex1). The third screen, titled 'Screen 1', displays a photo of an orange cat and the text 'Pet the Kitty! This is a screen I got after downloading Zentao! created via the MIT AI2 Companion'.

# Testando usando o MIT App Inventor 2 Companion



The screenshot displays the MIT App Inventor 2 Companion web interface. At the top, the MIT App Inventor logo is visible on the left, and navigation links for 'Projetos', 'Conectar', 'Compilar', and 'Ajuda' are in the center. On the right, there are links for 'Meus Projetos', 'Galeria', 'Guia', and 'Reportar'. Below the navigation bar, the main workspace is titled 'projeto' and contains a 'Screen1' dropdown menu. A dropdown menu is currently open, listing the following options: 'Assistente AI', 'Emulador', 'USB', 'Reiniciar a Conexão', and 'Reiniciar Conexões'. The 'Assistente AI' option is highlighted in green. To the left of the workspace is a 'Paleta' (palette) titled 'Interface de Usuário' (User Interface), which lists various UI components such as 'Botão', 'CaixaDeSeleção', 'EscolheData', 'Imagem', 'Legenda', 'EscolheLista', 'VisualizadorDeListas', 'Notificador', 'CaixaDeSenha', 'Deslizador', 'ListaSuspensa', 'CaixaDeTexto', 'EscolheHora', and 'NavegadorWeb'. The main workspace area shows a preview of the app running on a mobile device. The preview displays a screen titled 'Screen1' with a white background, a text input field, a button labeled 'Texto para Botão1', and a text label labeled 'Texto para Legenda1'. The mobile device's status bar at the top shows the time as 9:48 and various icons for Wi-Fi, signal strength, and battery.

# Testando usando o MIT App Inventor 2 Companion

1 install and launch

2 connect computer and phone to SAME WiFi

3 Click Connect and choose AI Companion

4 Scan

MIT App Inventor 2 Companion

MIT App Inventor 2

type in the 6-digit code  
-or-  
scan the QR code

Six Digit Code

connect with code

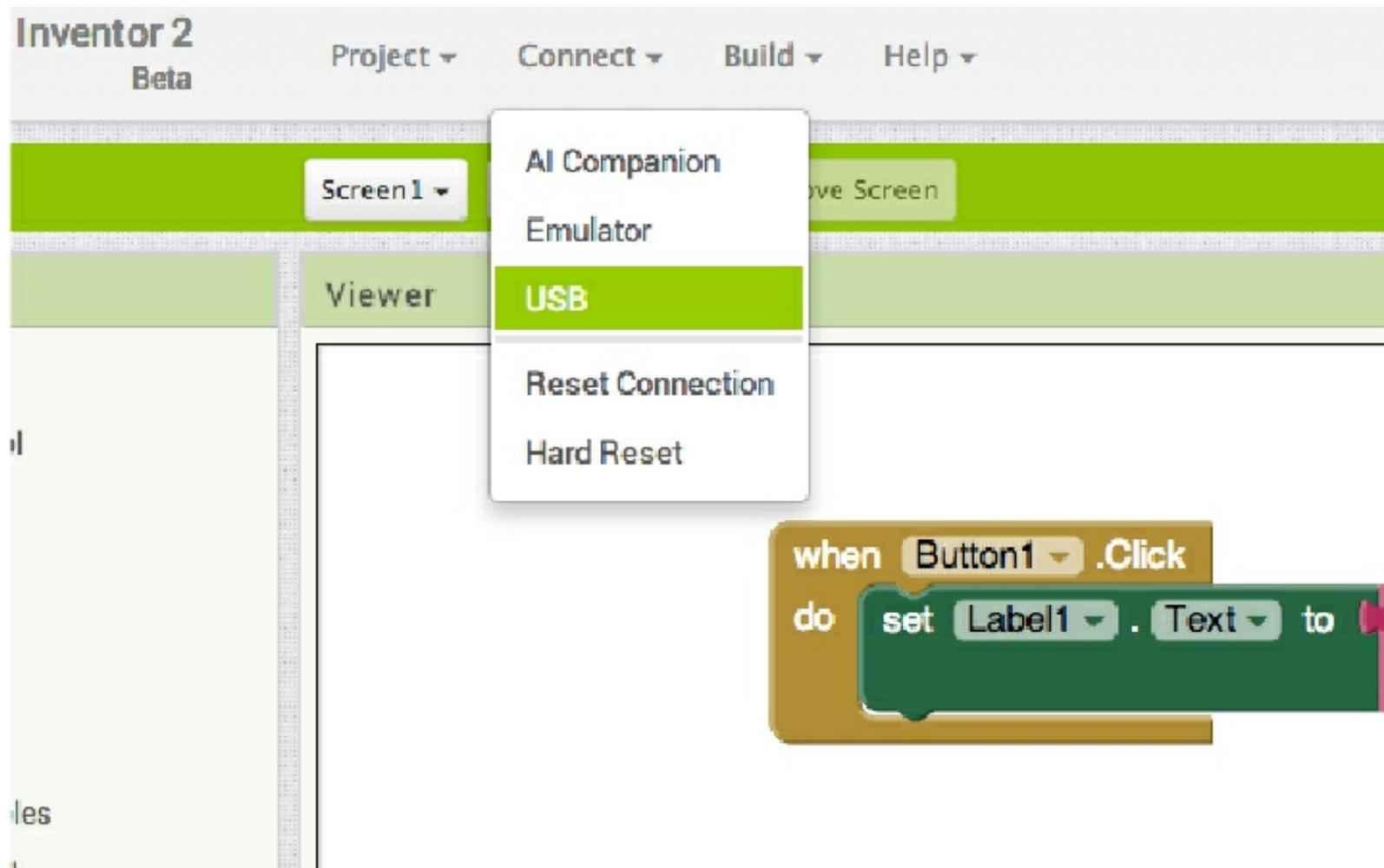
scan QR code

Your IP Address is: 192.168.1.111  
Version: 2.07nb9zx1

<http://www.appinventor.org/appInventor2Changes>

OU ...

Conecte seu smartphone android a porta usb do computador, e seleciona Connect e depois em USB.



OU ...

# Baixar o emulador para PC

<http://appinventor.mit.edu/explore/ai2/windows.html>

Podemos utilizar o emulador para computador. No endereço acima encontramos instruções para instalar em seu PC.

Desta forma, não precisamos de um dispositivo Android para realizar os testes.

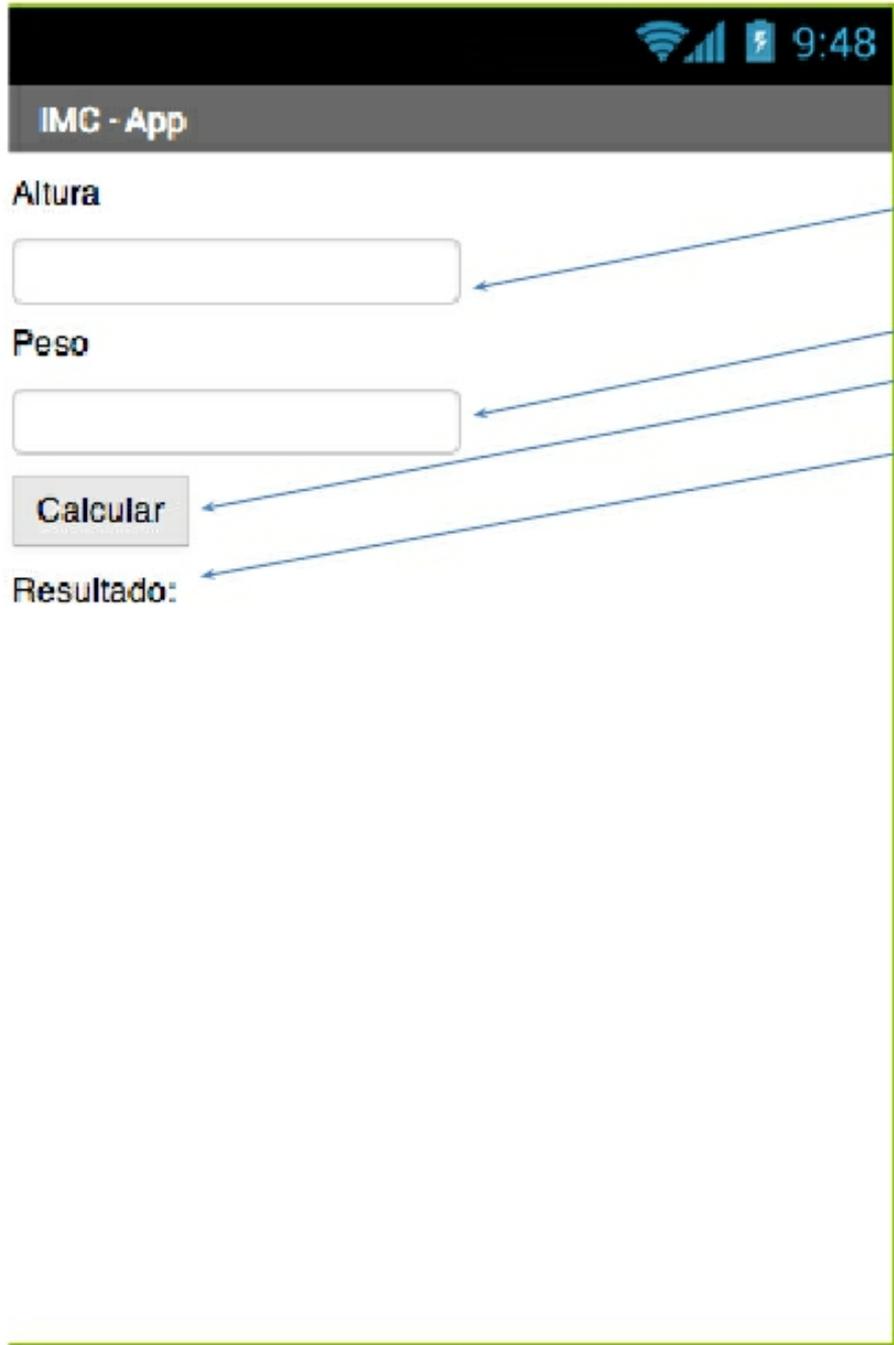


# Atividade guiada

Criar um aplicativo para calcular o IMC (Índice de Massa Corpórea).

1. Crie um novo projeto chamado IMC-APP
2. Monte a interface do aplicativo como a seguir

# Atividade guiada



Lembre-se de renomear os componentes.

# Atividade guiada

The image shows a programming environment interface with two main panels: 'Blocks' on the left and 'Viewer' on the right.

**Blocks Panel:**

- Built-in:**
  - Control
  - Logic
  - Math (highlighted)
  - Text
  - Lists
  - Colors
  - Variables
  - Procedures
- Screen1:**
  - Label1
  - txtAltura
  - Label2
  - txtPeso
  - btnCalcular
  - lblResultado
- Any component**

**Viewer Panel:**

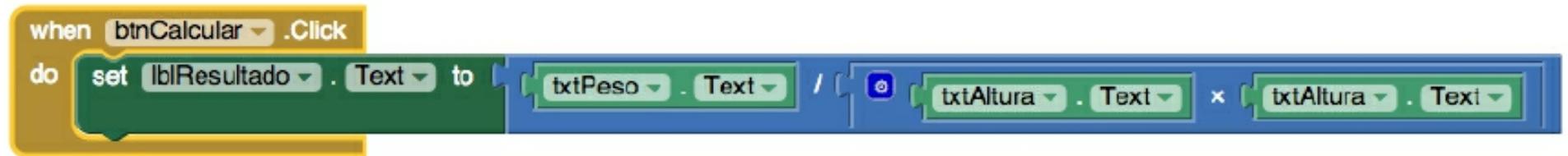
The script area contains several blocks:

- A '0' block.
- An '=' (assignment) block.
- A '+' (addition) block.
- A '-' (subtraction) block.
- A 'x' (multiplication) block, with a blue arrow pointing to it from the right.
- A '/' (division) block, highlighted with a yellow border and a blue arrow pointing to it from the right. Below it is a tooltip that reads: "Return the quotient of the two numbers."
- An '^' (power) block.
- A 'random integer from 1 to 100' block.
- A 'random fraction' block.

Para fazer o cálculo do IMC basta dividir seu peso em quilogramas pela altura ao quadrado (em metros). Então precisaremos dos blocos para dividir e multiplicar.

# Atividade guiada

Os blocos devem estar como na figura abaixo.



IMC - App

Altura

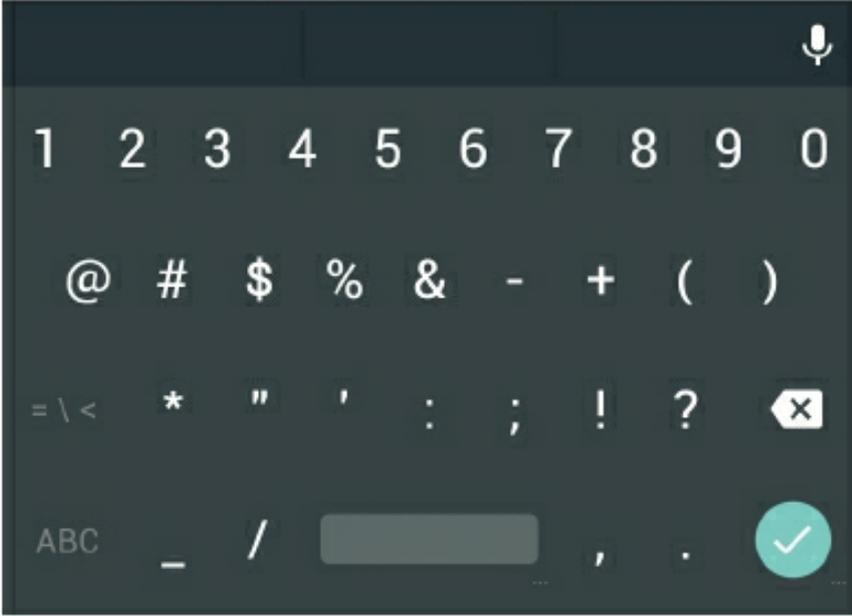
1.68

Peso

64

Calcular

22,67574





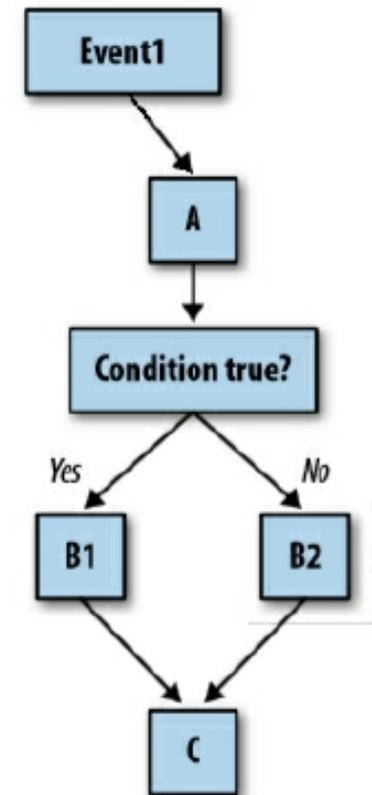
Estou tendo dificuldades  
em codificar este App.

Era esperado, pois vimos apenas eventos e ações. Contudo, mesmo em um dado evento, podemos ter diferentes comportamentos.



# Eventos e condições

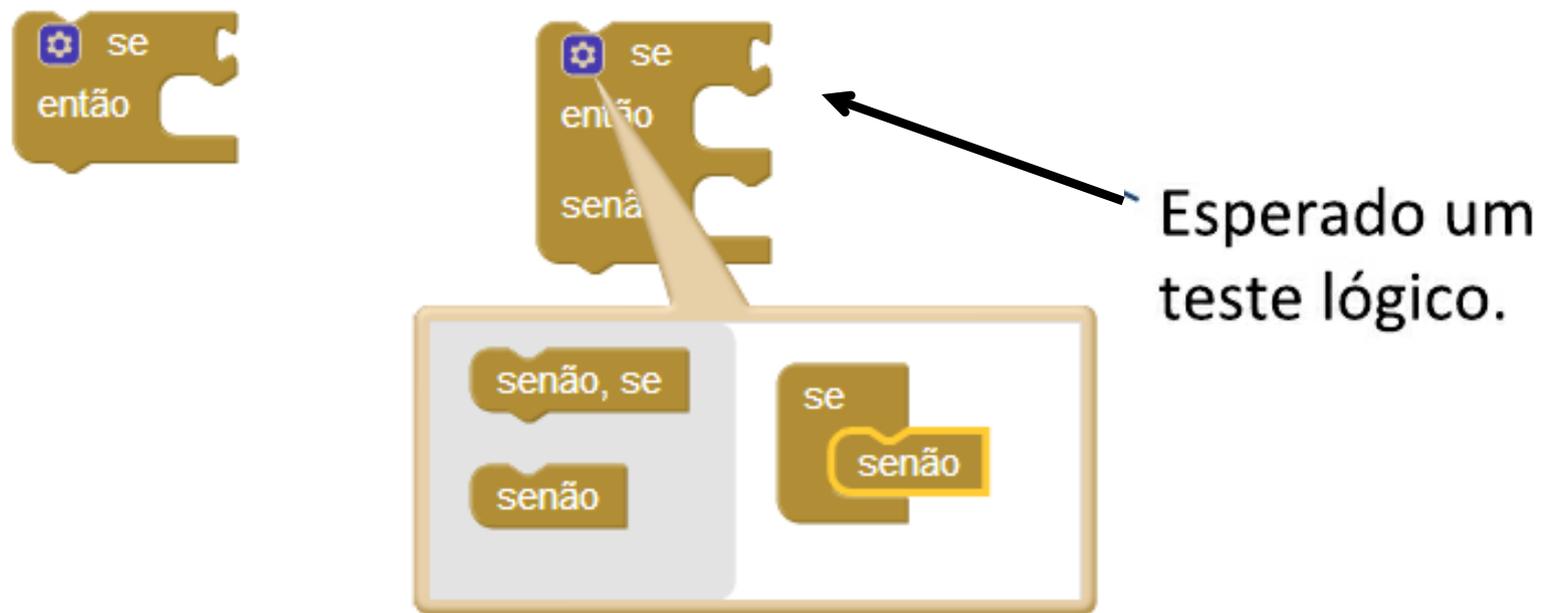
- Quando o evento 1 ocorre, a função A é executada
- Em seguida, um teste de condição executada
  - Se o teste é verdadeiro, B1 é executada
  - Se for falso, B2 é executado
- Em qualquer situação, o restante dos eventos (C) são executados completando o fluxo do programa



# Blocos condicionais

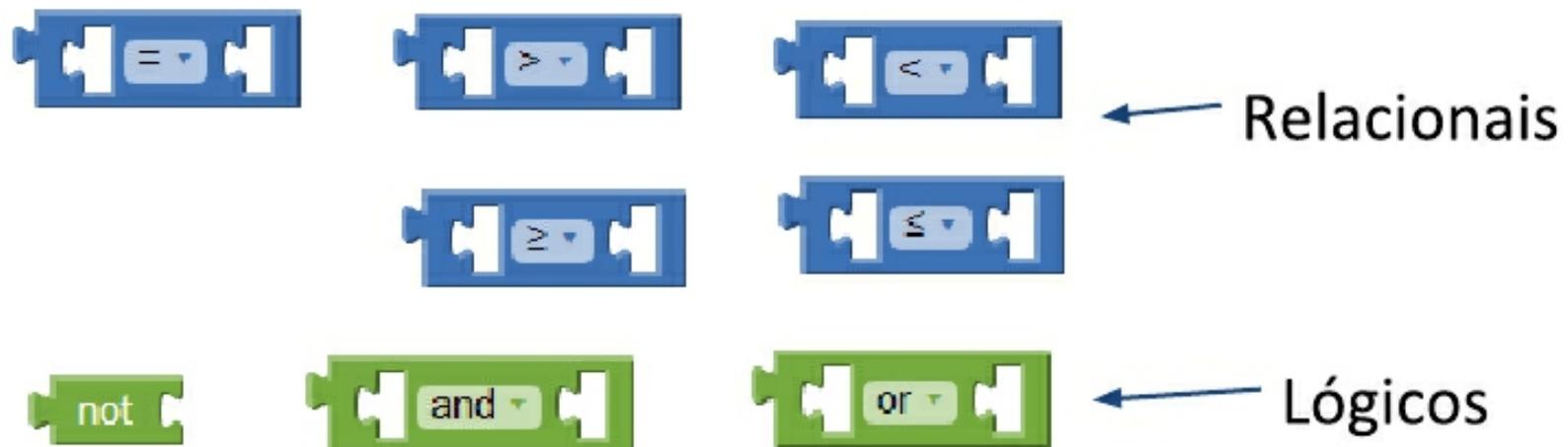
O App Inventor oferece dois tipos de blocos condicionais if e else if na opção control do editor de blocos.

O bloco if da opção pode conter vários “elses”, clicando no ícone azul e os adicionando para dentro do bloco.



# Testes lógicos

Um teste lógico é uma expressão lógica que inclui blocos relacionais e lógicos, que retornam um resultado verdadeiro ou falso.



Os blocos relacionais estão em Built-in - Math e os lógicos em Built-in Logics

# Blocos condicionais

Tanto para *if* quanto para *else if*, os blocos que você colocou dentro do encaixe *then* só será executado se o teste for verdadeiro

Para um bloco *if*, se o teste for falso, o aplicativo passa para os blocos abaixo dele

Se os testes *else if* forem falsos, os blocos dentro do *else* são executados



Não entendi muito bem  
esta última informação.

Ok. Vamos fazer uns exemplos,  
que lhe ajudarão a entender  
melhor.



Considerem um aplicativo simples, que ao clicar no botão “acende” uma lampada.



Na verdade, são diferentes  
imagens que serão mostradas  
em um dado momento. Click  
nas imagens para baixar cada  
imagem.

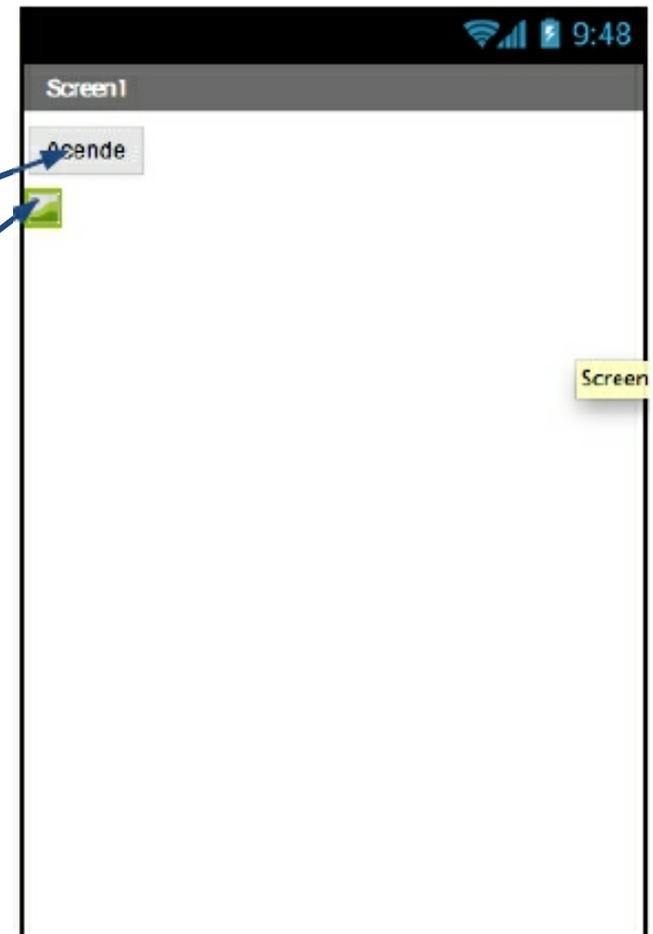


[www.sergioportari.com.br/lampada\\_apagada.png](http://www.sergioportari.com.br/lampada_apagada.png)  
[www.sergioportari.com.br/lampada\\_acesa.png](http://www.sergioportari.com.br/lampada_acesa.png)

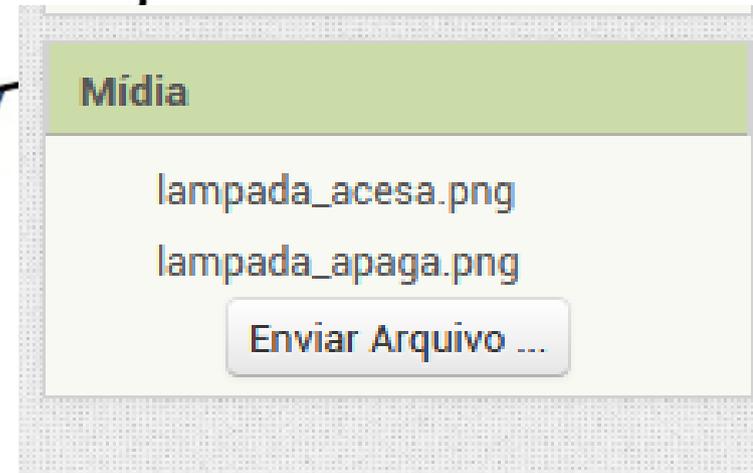
Criem um novo projeto, e arrastem um botão e uma imagem para o visualizador.



- Button
- CheckBox
- DatePicker
- Image
- Label
- ListPicker
- ListView
- Notifier
- PasswordTextBox
- Slider



Façam o upload das duas imagens das lampadas, clicado em upload file. Abaixo da coluna components.



Selecione a imagem, e na propriedade "picture" selecione a imagem da lampada apagada.



Components	Properties
<ul style="list-style-type: none"><li>Screen1<ul style="list-style-type: none"><li>Button1</li><li><b>Image1</b></li></ul></li></ul>	<p>Image1</p> <p>Picture lampada_apaga.png</p> <p>Visible showing</p> <p>Width Automatic...</p> <p>Height Automatic...</p>

No editor de blocos, façam a programação abaixo. Observem o nome das figuras.

Os blocos relacionais (comparações) estão em Built-in - Math.



```
quando Botão2 .Clique
fazer
  se Imagem1 . Imagem = " lampada_apaga.png "
  então
    ajustar Imagem1 . Imagem para " lampada_acesa.png "
    ajustar Botão2 . Texto para " Apagar "
  senão
    ajustar Imagem1 . Imagem para " lampada_apaga.png "
    ajustar Botão2 . Texto para " Acender "
```

Deu tudo certo, muito legal.



Acho que já posso  
incrementar o aplicativo  
do IMC :)



Sim, mas vamos aproveitar e ver o conceito de variáveis. Esse conceito irá nos ajudar nesse e outros Apps.



As variáveis são espaços de memória representados por um nome que não estão associados com um determinado componente. E



Existem dois tipos de variáveis,  
globais e locais. Falaremos  
primeiro das globais.



# Definindo variáveis globais

Você define uma nova variável explicitamente no editor de blocos arrastando um bloco *global variable* (variável global)

Você pode nomear a variável clicando no texto “*name*” dentro do bloco

Você pode especificar um valor inicial usando um bloco *number, text, color* ou *make a list*

Reabra o projeto IMC, e faça os passos a seguir para criar 2 variáveis globais.



Passos que você seguir para criar uma variável chamada *score* com um valor inicial de 0

1. Arraste o bloco *initialize global* da opção *Variables* em *Built-in*

A screenshot of a Scratch 'initialize global' block. The block is orange with a white outline. It contains the text 'inicializar global' in orange, 'nome' in a light blue box, and 'para' in white. A small white arrow is on the right side.

2. Altere o nome da variável clicando em "*name*" e digitando "*imc*"

A screenshot of a Scratch 'initialize global' block. The block is orange with a white outline. It contains the text 'inicializar global' in orange, 'imc' in a light blue box, and 'para' in white. A small white arrow is on the right side.

3. Defina o valor inicial para um número arrastando o bloco *number* presente na opção e conectando-o à definição de variável

A screenshot of a Scratch 'initialize global' block. The block is orange with a white outline. It contains the text 'inicializar global' in orange, 'imc2' in a light blue box, 'para' in white, and a blue '0' block connected to the right side.

Você ainda precisar criar uma variável msg que armazenará o texto a ser mostrado, como nos blocos abaixo:



inicializar global `imc2` para `0`

inicializar global `msg` para `" "`

# Obtendo e alterando variáveis

Quando você define uma variável o App Inventor cria dois blocos para ele, um *set* (definir) e um *get* (obter)

Esses blocos são visíveis passando o mouse sobre o nome da variável no bloco de inicialização



Concluem os blocos como na figura abaixo:



```
initialize global imc to 0
initialize global msg to ""
when btnCalcular .Click
do
  set global imc to (txtPeso . Text / (txtAltura . Text * txtAltura . Text))
  if
    get global imc < 18.5
  then
    set global msg to "Você está abaixo do peso ideal "
  else if
    get global imc < 25
  then
    set global msg to "Parabéns — você está em seu peso normal! "
  else if
    get global imc < 30
  then
    set global msg to "Você está acima de seu peso (sobrepeso) "
  else if
    get global imc < 35
  then
    set global msg to "Obesidade grau I "
  else if
    get global imc < 40
  then
    set global msg to "Obesidade grau II "
  else
    set global msg to "Obesidade grau III "
  set lblResultado . Text to get global msg
```

The image shows a Scratch script for calculating BMI. It starts with two global variables: 'imc' (BMI) initialized to 0 and 'msg' (message) initialized to an empty string. A 'when clicked' event triggers a 'do' block containing a 'set' block for 'imc' calculated as (txtPeso . Text / (txtAltura . Text \* txtAltura . Text)). This is followed by an 'if-then-else' chain with conditions: < 18.5 (below ideal weight), < 25 (normal weight), < 30 (overweight), < 35 (Obesity grade I), < 40 (Obesity grade II), and else (Obesity grade III). The final 'set' block updates 'lblResultado . Text' with the value of 'get global msg'.

# Hora de testar novamente

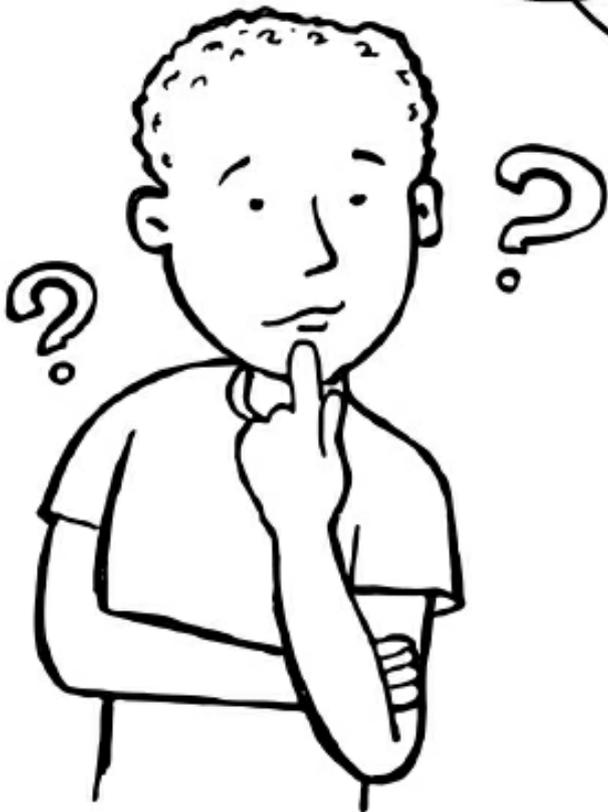
The image shows the MIT App Inventor web interface. At the top, there is a navigation bar with the MIT App Inventor logo and various menu items like 'Projetos', 'Conectar', 'Compilar', 'Ajuda', 'Meus Projetos', 'Galeria', 'Guia', 'Reportar um problema', 'Português do Brasil', and 'partici@gmail.com'. Below this is a green header bar with the project name 'lampada' and buttons for 'Screen1', 'Adicionar tela...', and 'Remover tela'. On the right side of this bar are 'Executar' and 'Atualizar' buttons.

The main workspace is divided into four panels:

- Paleta:** A list of UI components under the heading 'Interface de Usuário'. Components include Botão, CaixaDeSeleção, EscolhaData, Imagem, Legenda, EscolhaLista, VisualizadorDeListas, Notificador, CaixaDeSenha, Decodificador, ListaSuspensa, CaixaDeTexto, EscolhaHora, and NavegadorWeb.
- Visualizador:** A preview window showing the app's current state. A dialog box is overlaid on this panel.
- Componentes:** A tree view showing the hierarchy of components on the screen, including Screen1, Notific1, and Imagem1.
- Propriedades:** A panel for configuring the selected component (Imagem1). It shows properties like 'Alinhamento' (Automatico...), 'Legenda' (Automatico...), 'Imagem' (Imagem\_espigada.png...), 'RotacionarAngle' (0), 'RedimensionarPorCabeceira' (checked), and 'Visivel' (checked).

The dialog box in the center is titled 'Conectar ao Assistente' and contains the following text: 'Launch the MIT AI2 Companion on your device and then scan the barcode or type in the code to connect for live testing of your app. Need help finding the Companion App?'. Below the text is a QR code and the text 'Seu código é: yokrpy'. A 'Cancelar' button is at the bottom left of the dialog.

Legal esses apps, mas  
espera algo que usasse  
mais os recursos do meu  
smartphone.



Sim, mas agora podemos  
explorar mais estes recurso.  
Vamos começar com a tela e o  
acelerometro..



Um app que permite desenhar diretamente na tela, e você pode apagar apenas balançando (shaking) seu smartphone.



Crie um novo projeto e adicione um canvas e um AccelerometerSensor na view do Aplicativo.



Desenho

Paleta

- Interface de Usuário
- Organização
- Mídia
- Desenho e Animação
- Bola
- Pinball
- SpriteImagem

Sensores

Social

Armazenamento

Conectividade

LEGO® MINDSTORMS®

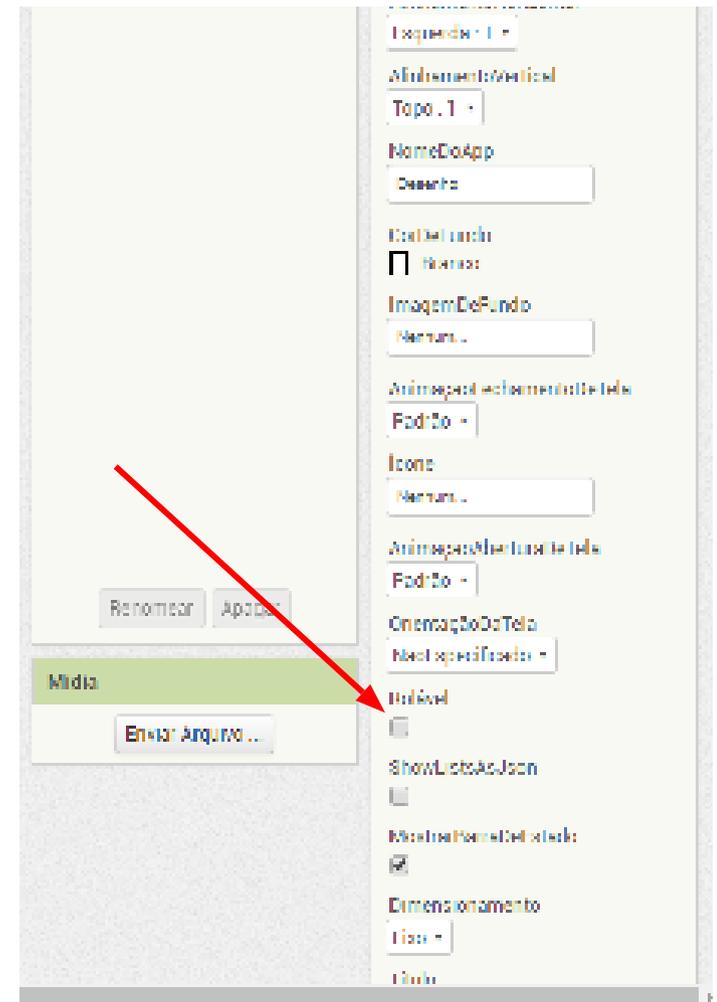
Experimental

Extension

Paleta

- Interface de Usuário
- Organização
- Mídia
- Desenho e Animação
- Sensores
  - SensorAcelerômetro
  - CódigoDeBarras
  - Temporizador
  - GyroscopeSensor
  - SensorDeLocalização
  - NearField
  - SensorDeOrientação
  - Pedometer
  - SensorDeProximidade

Desmarque a propriedade scrollable do objeto Screen



Escolha a opção Fill Parent para a largura (width) e altura (height).



The screenshot displays the Android Studio interface. On the left, the 'Componentes' (Components) panel shows a hierarchy: 'Screen1' containing 'Pintura1' and 'SensorAcelerômetro1'. A red arrow points from the 'Pintura1' component to the 'Propriedades' (Properties) panel on the right. The 'Propriedades' panel lists various attributes for the 'Pintura1' component, including 'CorDeFundo' (White), 'ImagemDeFundo' (None), 'TamanhoDaFonte' (14.0), 'Altura' (Fill parent), 'Largura' (Fill parent), 'LarguraDaLinha' (2.0), 'CorDePintura' (Black), 'AlinhamentoDoTexto' (center), and 'Visível' (checked). At the bottom of the components panel are 'Renomear' and 'Apagar' buttons.

Basta fazer a seguinte programação no editor de blocos.



```
quando SensorAcelerômetro1 ▾ .Sacudindo  
fazer chamar Pintura1 ▾ .Apagar
```

```
quando Pintura1 ▾ .Arrastado  
  xInicial yInicial xAnterior yAnterior xAtual yAtual algumSpriteArrastado  
fazer chamar Pintura1 ▾ .DesenharLinha  
  x1 obter xAnterior ▾  
  y1 obter yAnterior ▾  
  x2 obter xAtual ▾  
  y2 obter yAtual ▾
```

Agora começou a ficar interessante.



Existem diversos recursos,  
fique a vontade de explorá-  
los. Por exemplo em média  
você encontrará blocos para  
sintetizar e reconhecer  
vozes.



Iremos ver agora, como  
organizar melhor os objetos  
da interface gráfica.



O AppInventor não tem muita flexibilidade para o desenvolvimento da interface gráfica, porém é possível definir como os objetos serão alinhados.



Monte a seguinte interface,  
com um label, textbox e um  
button.



Screen2

Nome

Adiciona

Observe, que por padrão, os objetos são alinhados verticalmente.



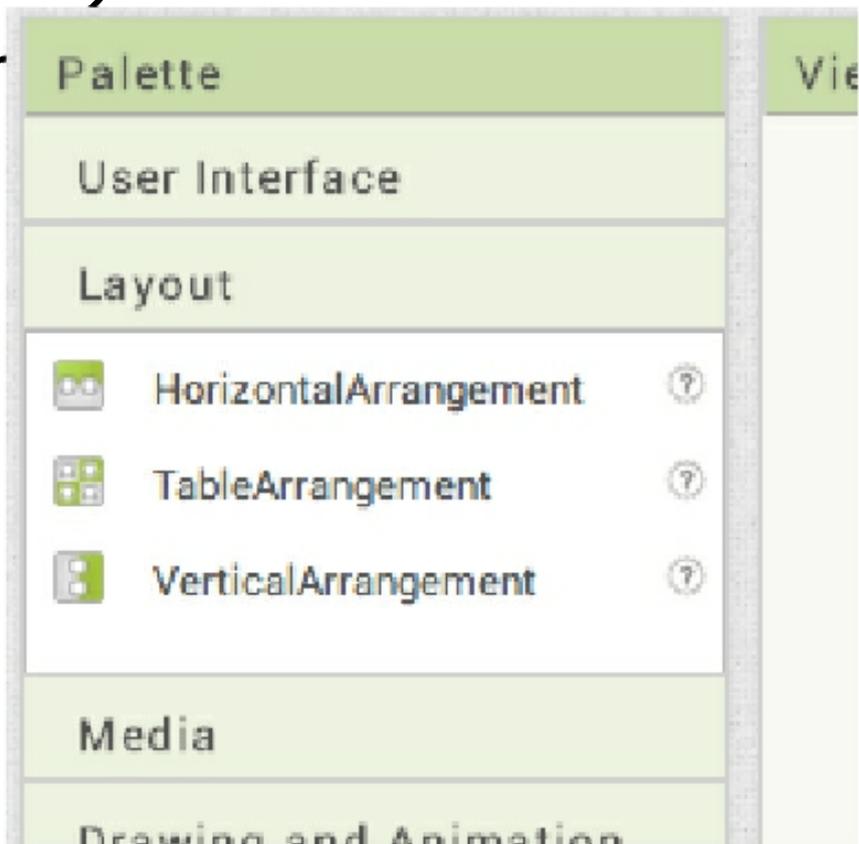
Screen2

Nome

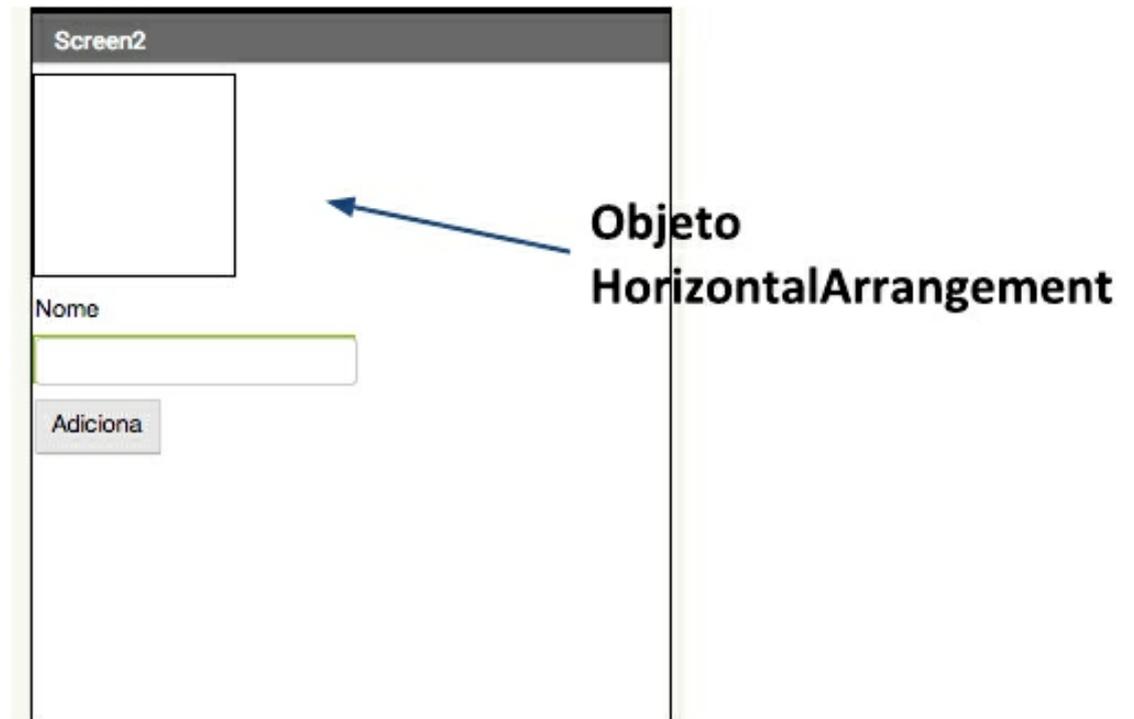
Adiciona

A screenshot of an Android application interface. At the top, there is a dark grey header bar with the text "Screen2" in white. Below the header, the word "Nome" is displayed in a dark grey font. Underneath "Nome" is a white text input field with a thin grey border. Below the input field is a rectangular button with a light grey background and a thin green border, containing the text "Adiciona" in a dark grey font.

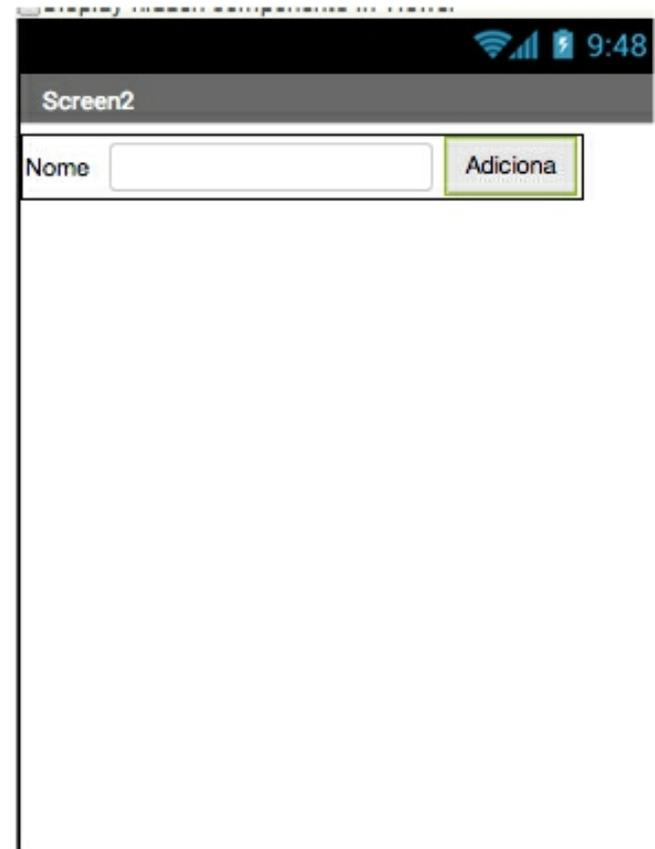
Caso queiram alinhá-los horizontalmente, adicione o objeto HorizontalArrangement na tela. Este objeto está em Palete, grupo Layout.



Depois de adicionado, sua tela  
ficará similar a seguinte.



Então, arrastem e soltem os  
objetos (label, textbox e button)  
no objeto  
HorizontalArrangement.



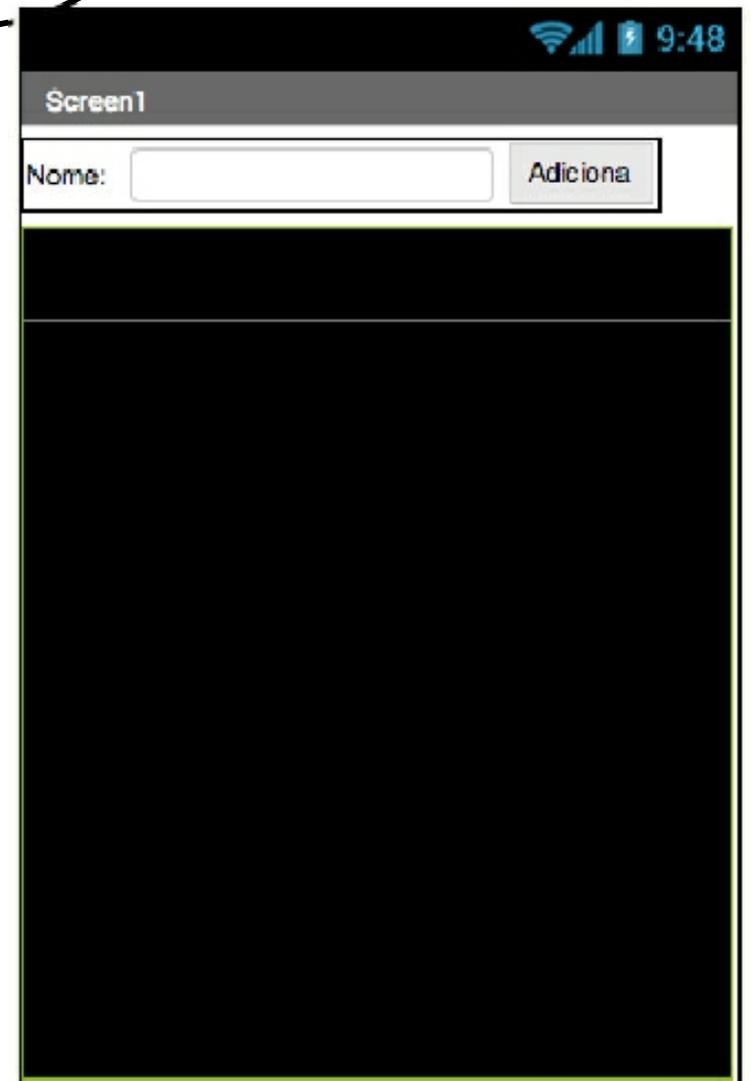
Podemos fazer diferentes layouts de tela utilizando os objetos `HorizontalArrangement`, `VerticalArrangement` e `TableArrangement`. Experimentem.



Agora, veremos como usar um objeto muito comum nos apps, o ListView.



Adicione um listview a sua tela, e altere as propriedades Width e Height para Fill Parent.



Agora é só programar no  
block designer e testar.



```
initialize global lstNomes to create empty list

when Button1 .Click
do
  add items to list list
  item get global lstNomes
  TextBox1 . Text
  set ListView1 . Elements to get global lstNomes
```

No último App, a lista é montada com informações dadas pelo usuário.



Certo. Então vamos aproveitar e aprender usar múltiplas telas.

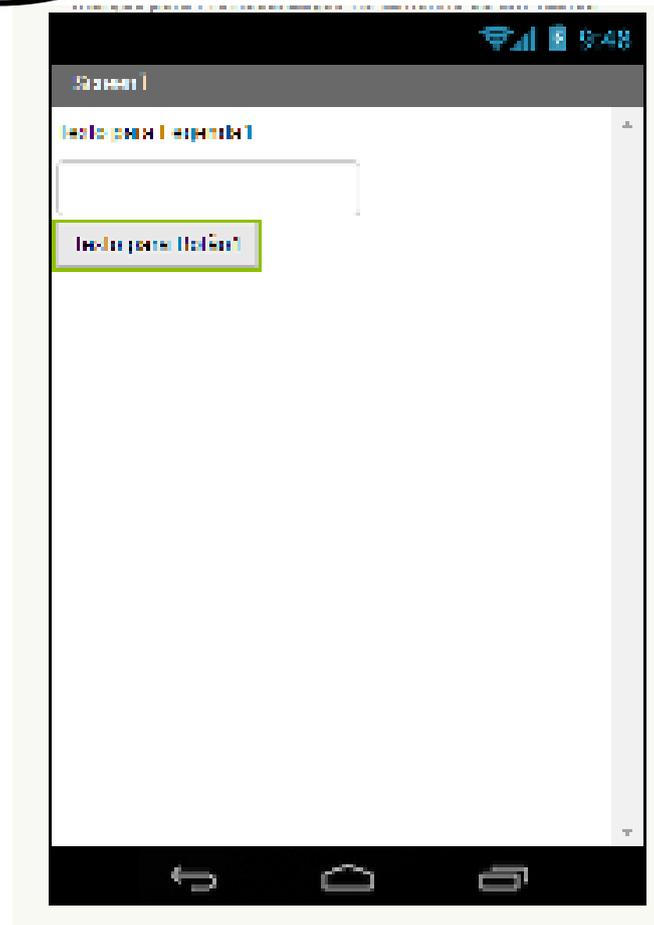


Crie um novo projeto

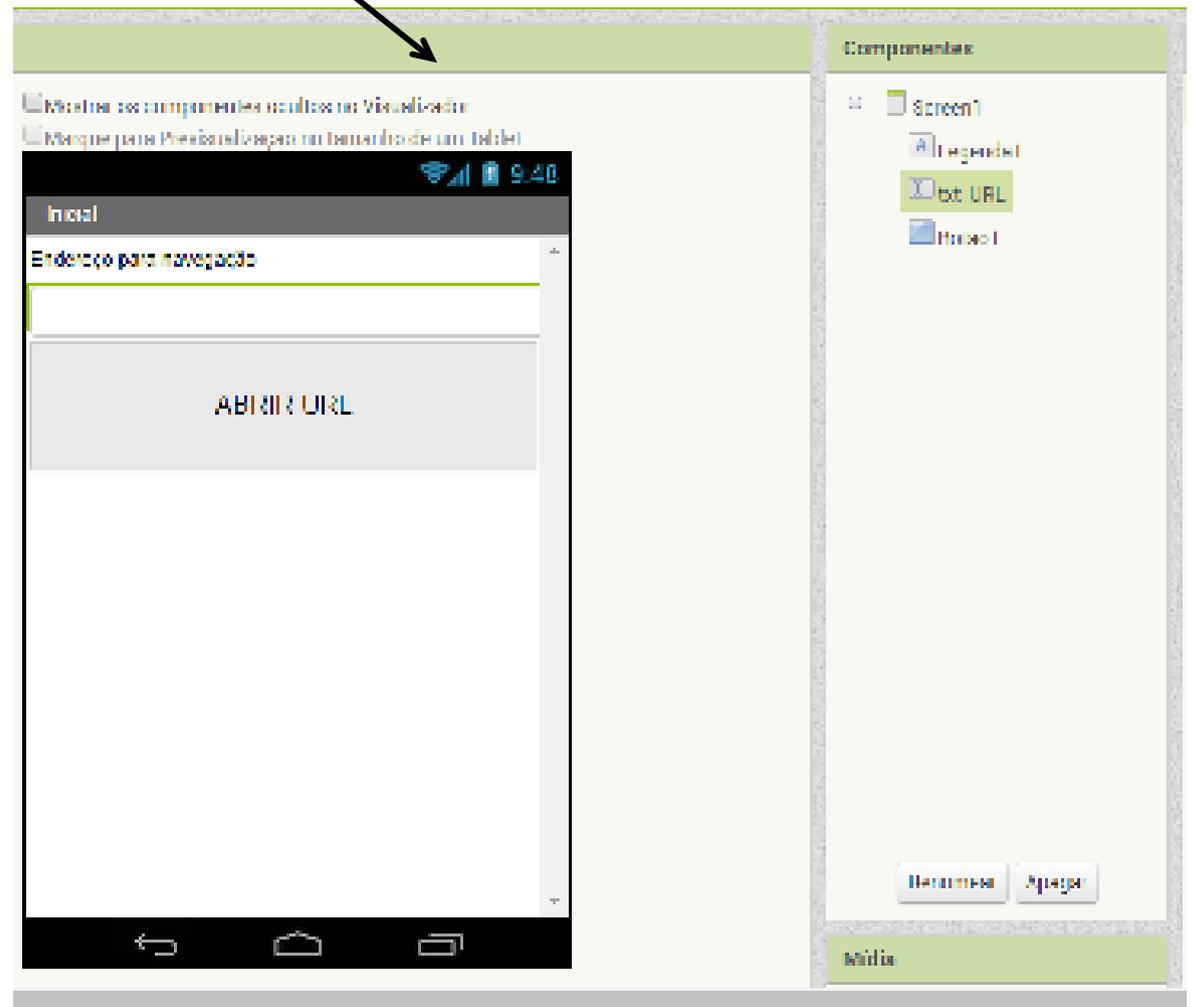


A screenshot of the MIT App Inventor web interface. The top navigation bar includes the MIT App Inventor logo and menu items: 'Projetos', 'Conectar', 'Compilar', and 'Ajuda'. The main workspace shows a project named 'teste2'. On the left, there is a 'Paleta' (palette) for the 'Interface de Usuário' (User Interface) with various components like 'Botão', 'CaixaDeSeleção', 'EscolheData', 'Imagem', 'Legenda', 'EscolheLista', 'VisualizadorDeListas', 'Notificador', and 'CaixaDeSenha'. On the right, the 'Projetos' dropdown menu is open, listing options such as 'Meus projetos', 'Iniciar novo projeto ...', 'Importar projeto (.aia) do meu computador...', 'Importar projeto (.aia) de um repositório ...', 'Apagar Projeto', 'Salvar projeto', 'Salvar projeto como...', 'Ponto de controle', 'Exportar o projeto selecionado (.aia) para o meu computador', 'Exportar todos os projetos', 'Importar keystore', 'Exportar a keystore', and 'Apagar keystore'. The 'Iniciar novo projeto ...' option is highlighted in green. An arrow from the text bubble above points to the 'Projetos' menu.

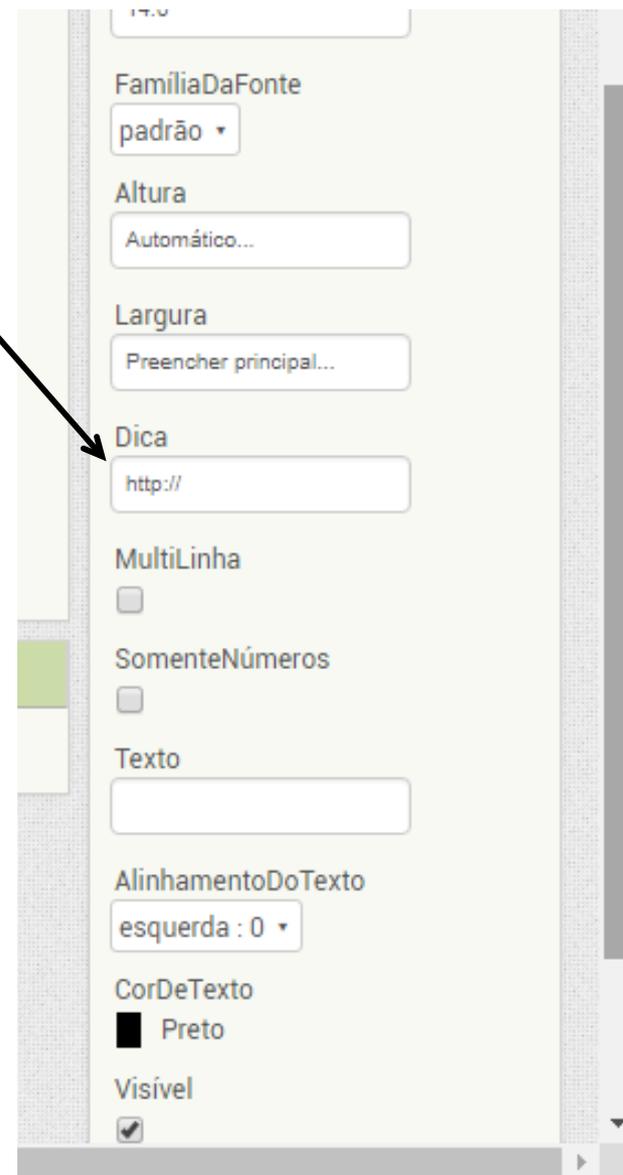
Adicione uma  
Legenda, uma Caixa  
de Texto e um  
Botão



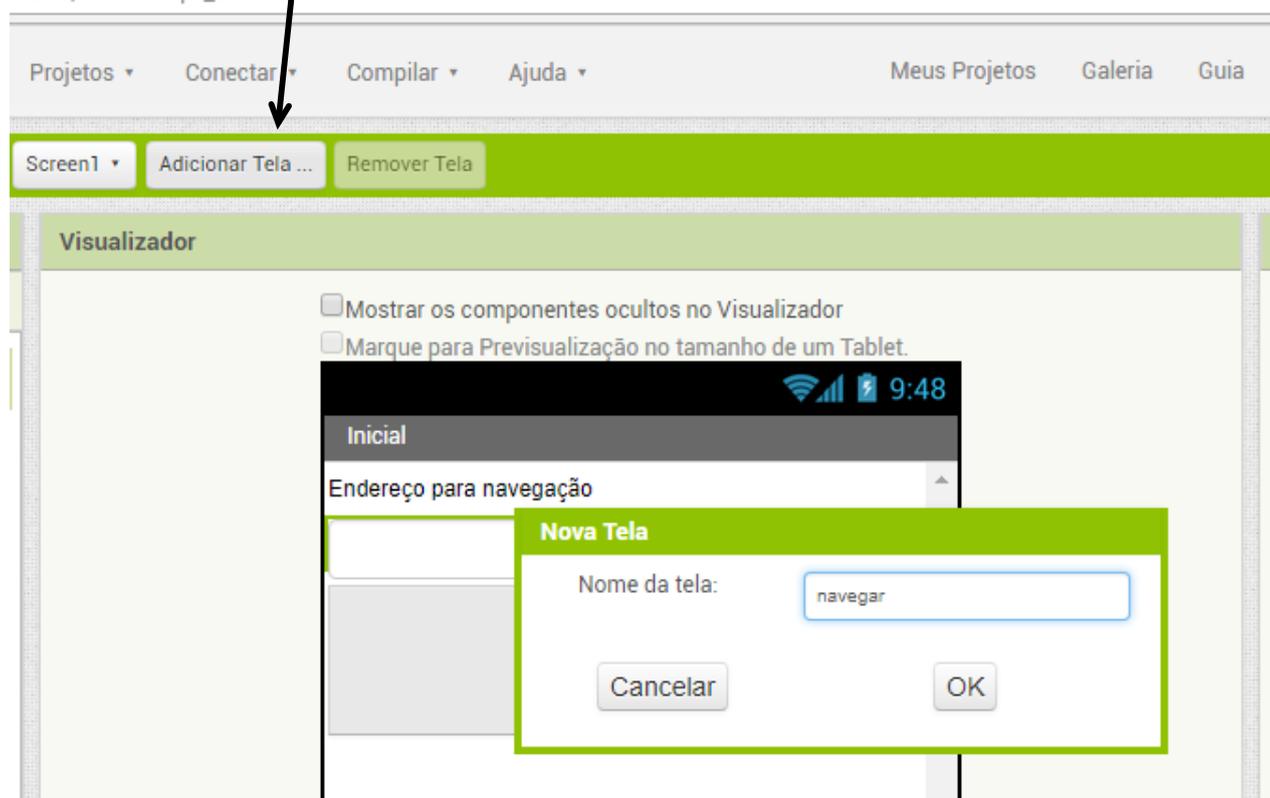
Modifique os nomes dos componentes e as propriedades para ficar assim:



No txt\_URL, coloque  
na propriedade Dica:  
http://



Adicione uma nova tela ao seu projeto. Dê o nome de navegar.

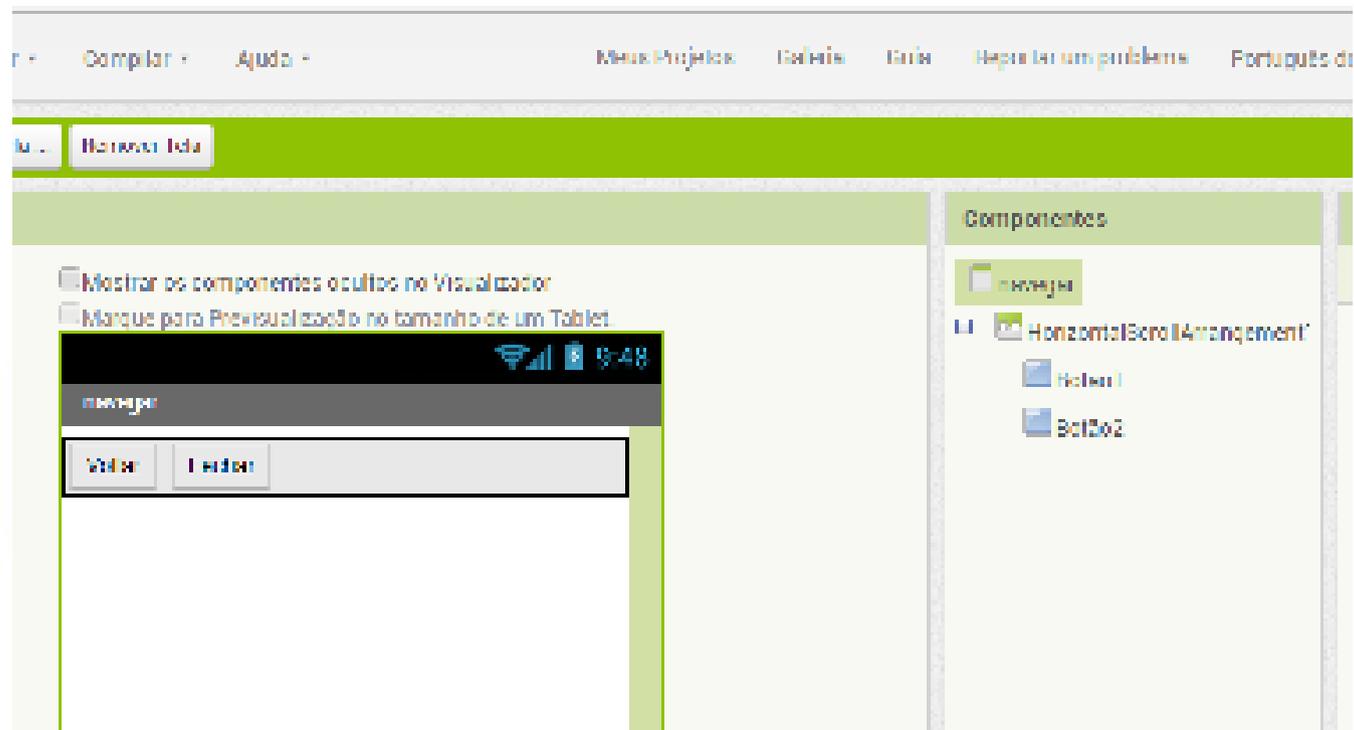


Adicione um  
HorizontalScrollArrangement  
e ajuste a largura para  
preencher principal

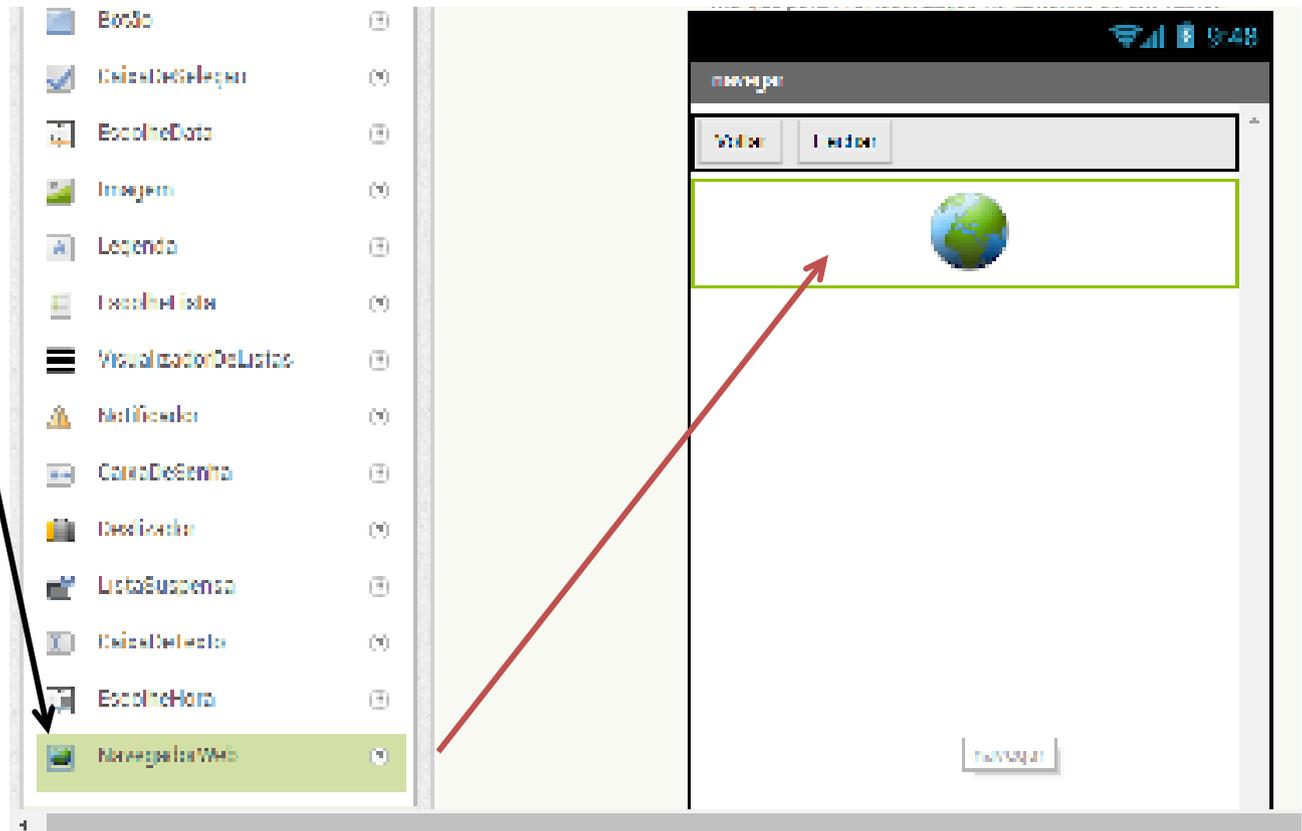
The screenshot displays the Android Studio interface with four main panels: Paleta, Visualizador, Componentes, and Propriedades. The Paleta (Palette) on the left is set to 'Organização' (Layout) and shows 'HorizontalScrollArrangement' selected. The Visualizador (Preview) shows a mobile device screen with a 'navegar' button and a large grey rectangular area representing the scroll arrangement. The Componentes (Components) panel shows the 'HorizontalScrollArrangement' widget added to the 'navegar' component. The Propriedades (Properties) panel on the right shows the configuration for 'HorizontalScrollArrangement 1', including 'AlinhamentoHorizontal' (Left), 'AlinhamentoVertical' (Top), and 'CorDeFundo' (Padrão).



Acione a propriedade Rolável da nova tela. Depois insira dois botões no HorizontalScrollArrangement



Abaixo do  
HorizontalScrolArrangement (fora  
dele) insira um NavegadorWeb



Nos blocos desta  
tela navegador  
coloque o seguinte:



```
quando navegar ▾ .Inicializar
fazer chamar NavegadorWeb1 ▾ .IrParaUrl
      url obter valor inicial

quando Botão1 ▾ .Clique
fazer chamar NavegadorWeb1 ▾ .IrParaTrás

quando Botão2 ▾ .Clique
fazer fechar tela
```

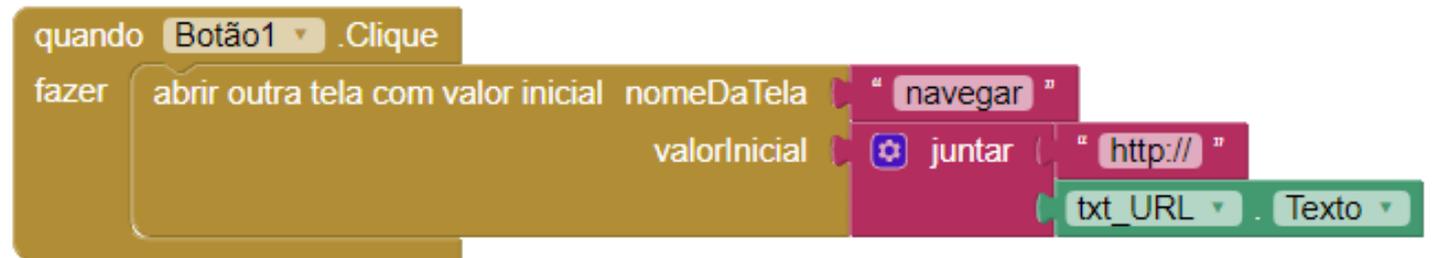
Selecione a Screen1  
para voltarmos à  
tela inicial do  
projeto



The image shows the MIT App Inventor web interface. At the top, there is a menu bar with options: "Projetos", "Conectar", "Compilar", "Ajuda", and "Novo Projeto". Below this is a green header bar labeled "Navegador" with three buttons: "navegar", "Adicionar tela...", and "Remover tela". An arrow from the text above points to the "navegar" button. A dropdown menu is open under "navegar", showing "Screen1" and "navegar". To the left is a "Blocos" (Blocks) palette with categories like "Interna", "Controle", "Lógica", "Matemática", "Texto", "Listas", "Cores", "Variáveis", "Procedimentos", "navegar", and "Inicialização/Encerramento". The "navegar" category is expanded, showing a "Botão1" block. On the right, the code editor shows three event-driven blocks:

- quando **navegar** for inicializado  
fazer chamar **NavegadorWeb1** .InParaUrl  
url: obter valor inicial
- quando **Botão1** for clicado  
fazer chamar **NavegadorWeb1** .InParaTela
- quando **Botão2** for clicado  
fazer fechar tela

Coloque agora estes blocos na Screen1



Agora é só testar...



E se eu quisesse  
algo um pouco mais  
avançado?



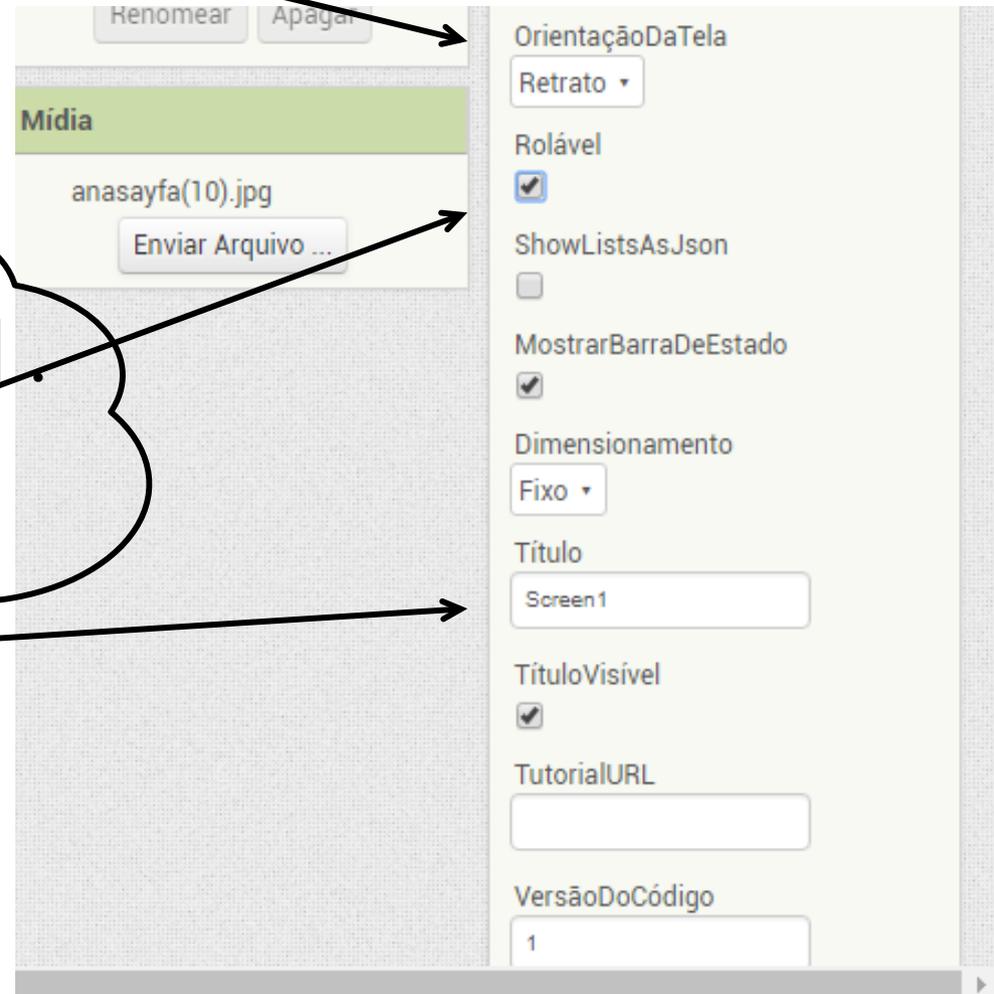
Crie um novo projeto.  
Vamos criar um Bate Papo online



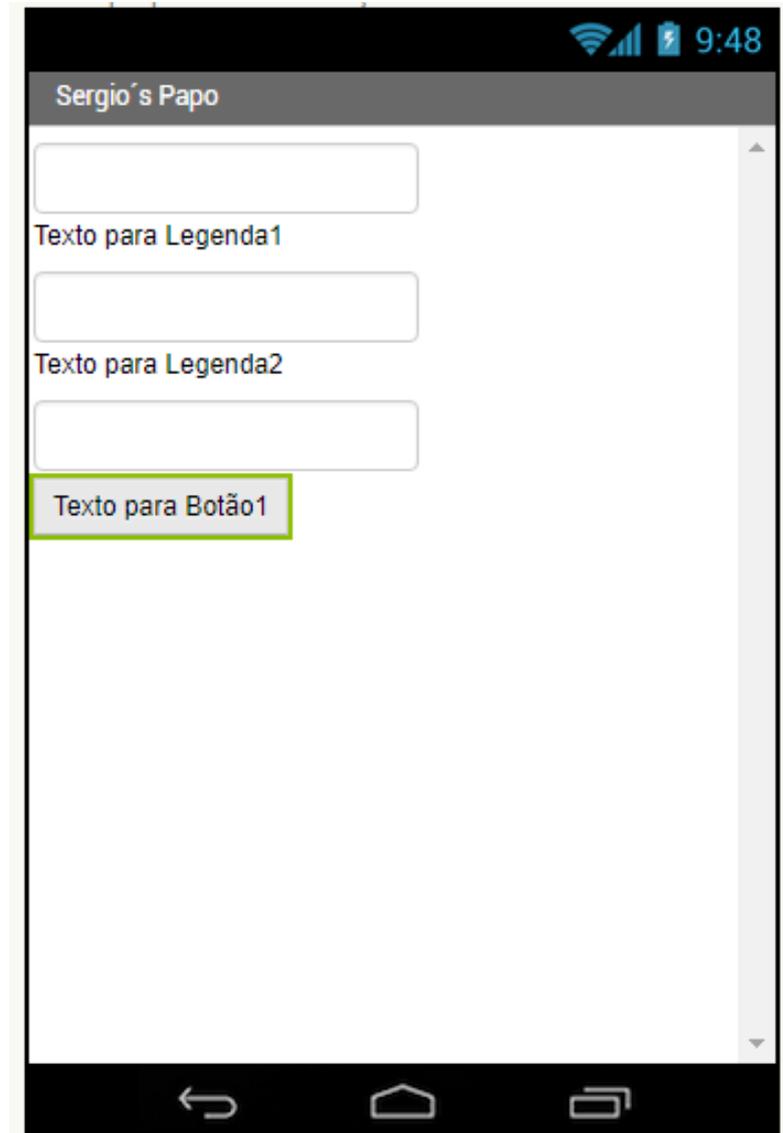
A screenshot of the MIT App Inventor web interface. The top navigation bar includes the MIT App Inventor logo and menu items: 'Projetos', 'Conectar', 'Compilar', and 'Ajuda'. The left sidebar is titled 'Desenho' and contains a 'Paleta' section with 'Interface de Usuário' components: Botão, CaixaDeSeleção, EscolheData, Imagem, Legenda, EscolheLista, VisualizadorDeListas, and Notificador. A dropdown menu is open under the 'Projetos' menu, with the 'Iniciar novo projeto ...' option highlighted in green. Other options in the menu include 'Importar projeto (.aia) do meu computador...', 'Importar projeto (.aia) de um repositório ...', 'Apagar Projeto', 'Salvar projeto', 'Salvar projeto como...', 'Ponto de controle', 'Exportar o projeto selecionado (.aia) para o meu computador', 'Exportar todos os projetos', 'Importar keystore', 'Exportar a keystore', and 'Apagar keystore'. Two black arrows originate from the text in the thought bubble above: one points to the 'Projetos' menu, and the other points to the 'Iniciar novo projeto ...' option in the dropdown.

Altere a propriedade  
Orientação da  
Screen para Retrato  
(Portrait)

Marque a opção Rolável.  
Coloque um título na  
Screen1 (Bate Papo por  
exemplo)



Monte a estrutura  
com 3 Caixas de  
texto,  
2 Legendas, 1 botão





Na caixa de texto 1, desmarque a opção Ativado. Em Largura, selecione Preencher principal

Apague o conteúdo da caixa Dica. Acione a opção MultiLinha

Troque a CorDeFundo para Cinza e a CorDeTexto para Laranja

CaixaDeTexto1

- CorDeFundo:  Padrão
- Ativado:
- FonteNegrito:
- Fonteltálico:
- TamanhoDaFonte: 14.0
- FamiliaDaFonte: padrão ▾
- Altura: Automático...
- Largura: Preencher principal
- Dica:
- MultiLinha:
- SomenteNúmeros:
- Texto:

Midia

anasayfa(10).jpg

Enviar Arquivo ...

Renomeie a  
CaixadeTextos 1 para  
txt\_Mensagens



The screenshot displays the Android Studio IDE interface. On the left, a preview window shows a mobile application screen with a text input field. A dialog box titled "Renomear Componente" is open, allowing the user to rename a selected component. The dialog has two input fields: "Nome antigo" (Old Name) containing "CaixaDeTextos 1" and "Nome novo" (New Name) containing "txt\_Mensagens". Below the fields are "Cancelar" and "OK" buttons. On the right, the "Componentes" (Components) panel lists the UI elements on the screen, including "CaixaDeTextos1", "Legenda1", "CaixaDeTextos2", "Legenda2", "CaixaDeTextos3", and "Botao1". At the bottom of this panel are "Renomear" and "Apagar" buttons. A green box highlights the dialog, and two black arrows originate from the text in the thought bubble above: one points to the "Nome novo" field, and the other points to the "Renomear" button in the Components panel.

Selecione a Legenda1 e  
escreva, em Texto,  
Nome:



The screenshot displays the Android Studio interface. On the left, the 'Screen1' view shows a list of UI components: txt\_Mensagens, Legenda1 (highlighted), CaixaDeTexto2, Legenda2, CaixaDeTexto3, and Botão1. On the right, the 'Legenda1' properties panel is open, showing various settings for the text view. The 'Nome' field is currently empty, and an arrow from the text bubble above points to it. Other visible settings include CorDeFundo (set to 'Nenhum'), FonteNegrito (unchecked), Fonteltálico (unchecked), TamanhoDaFonte (14.0), FamíliaDaFonte (padrão), HTMLFormat (unchecked), TemMargens (checked), Altura (Automático...), Largura (Automático...), AlinhamentoDoTexto (esquerda : 0), CorDeTexto (Preto), and Visível (checked). At the bottom, there are buttons for 'Renomear' and 'Apagar', and a 'Mídia' section with 'anasayfa(10).jpg' and an 'Enviar Arquivo ...' button.

Em CaixaDeTexto2,  
apague o texto em Dica,  
na Largura, selecione  
Preencher principal e  
renomeie para  
txt\_Nome



The screenshot displays the Android Studio interface. On the left, a mobile app preview shows a form with fields for 'Nome', 'Texto para Legenda3', and 'Texto para Botão1'. A 'Renomear Componente' dialog box is open, showing the current name 'CaixaDeTexto2' and the new name 'txt\_Nome'. On the right, the 'CaixaDeTexto2' component inspector is visible, showing various properties such as 'Conteúdo', 'Preencher', 'Através', 'FonteNegrito', 'FontesBásico', 'TamanhoDeFonte', 'FamiliaDeFonte', 'Alinhamento', 'Largura', 'Dica', 'MultiLinha', and 'SequenciaDeFormas'. The 'Largura' property is set to 'Preencher principal'. The 'Dica' property is currently empty. The 'Renomear' button is highlighted.

Selecione a Legenda2 e escreva, em Texto, Mensagem:



The screenshot displays the Android Studio interface. On the left, the 'Screen1' layout is shown with several components: 'txt\_Mensagens', 'Legenda1', 'CaixaDeTexto2', 'Legenda2', 'CaixaDeTexto3', and 'Botão1'. The 'Legenda1' component is currently selected. On the right, the 'Properties' panel for 'Legenda1' is visible, showing various settings such as 'CorDeFundo', 'FonteNegrito', 'FonteTálico', 'TamanhoDaFonte' (set to 14.0), 'FamíliaDaFonte' (set to 'padrão'), 'HTMLFormat', 'TemMargens' (checked), 'Altura' (set to 'Automático...'), 'Largura' (set to 'Automático...'), 'Texto' (set to 'Mensagem:'), 'AlinhamentoDoTexto' (set to 'esquerda : 0'), 'CorDeTexto' (set to 'Preto'), and 'Visível' (checked). An arrow points from the text in the thought bubble to the 'Texto' property field.

Em CaixaDeTexto3,  
apague o texto em Dica,  
na Largura, selecione  
Preencher principal e  
renomeie para  
txt\_Mensagem



The image shows a screenshot of an Android IDE interface. On the left, a mobile emulator displays a form with fields for 'Nome:', 'Texto para Legenda3', and 'Texto para Botão1'. A dialog box titled 'Renomear Componente' is overlaid on the emulator, showing the current name 'CaixaDeTexto3' and the new name 'Txt\_Mensagem'. On the right, the IDE's design view shows a tree of components with 'CaixaDeTexto2' selected. The properties panel for 'CaixaDeTexto2' is visible, with the 'Largura' property set to 'Preencher principal'. Below the design view, the 'Midia' section shows 'android:src="@drawable/..."' and an 'Enviar Arquivo...' button. Arrows from the text in the thought bubble point to the 'CaixaDeTexto2' component in the design view, the 'Largura' property in the properties panel, and the 'Renomear' button in the design view.

No Botão1, Selecione Preencher principal em Largura, coloque em Texto, Enviar, e renomeie para btn\_Enviar



A screenshot of an Android IDE interface. The main window shows a mobile app preview with a form containing fields for 'Nome:' and 'Mensagem:'. A 'Renomear Componente' (Rename Component) dialog box is open in the foreground. The dialog has two input fields: 'Nome anterior' (Previous name) containing 'Código' and 'Novo nome' (New name) containing 'btn\_Enviar'. Below the fields are 'Cancelar' and 'OK' buttons. To the right, the 'Design' (Design) view shows a list of UI components. A 'Botão1' (Button1) component is highlighted in green. Below the component list, the 'Propriedades' (Properties) panel is visible, showing various settings for the selected button, including 'Largura' (Width) set to 'Preencher principal' (Fill main) and 'Texto' (Text) set to 'Enviar'. Arrows from the text in the thought bubble point to the 'Novo nome' field, the 'Botão1' component, and the 'Enviar' text property.

Em Sensores, adicione um Temporizador ao projeto



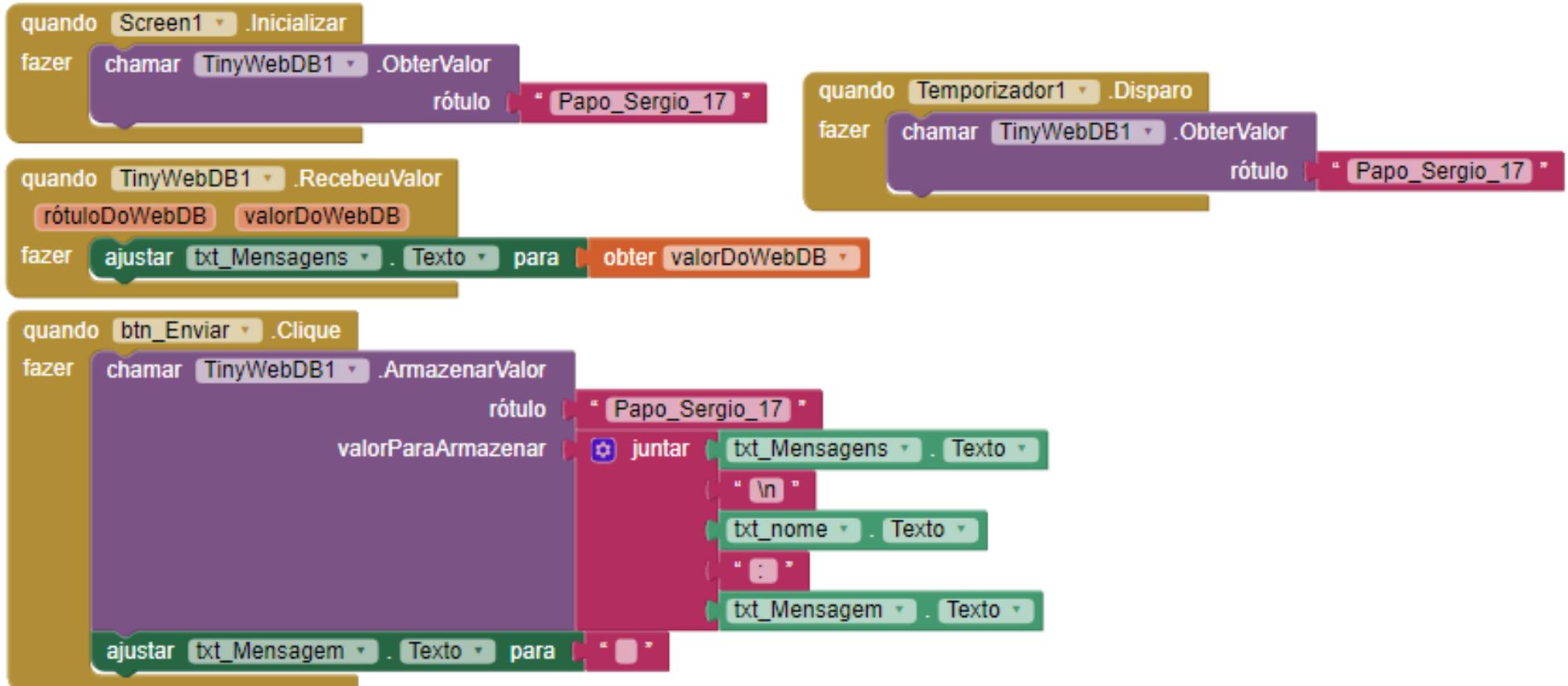
A screenshot of the Android Studio IDE interface. The left sidebar shows the 'Sensores' (Sensors) category selected, with the 'Temporizador' (Timer) component highlighted. An arrow points from this component to the central preview window. The preview window shows a mobile app interface with a form containing 'Nome:' and 'Mensagem:' labels, input fields, and an 'Enviar' button. The right sidebar shows a list of components, with 'Temporizador' also highlighted. At the bottom, the 'Componentes invisíveis' (Invisible Components) palette shows the 'Temporizador' component being added to the project.

Em Armazenamentos,  
adicione um  
TinyWebDB que será  
um banco de dados  
na WEB



A screenshot of an IDE interface, likely for Android development. The interface is divided into three main sections: a left-hand 'Paleta' (Palette) with various component categories, a central 'Visualizador' (Viewer) showing a mobile app preview, and a right-hand 'Componentes' (Components) list. In the 'Paleta', the 'Armazenamento' (Storage) category is expanded, and 'TinyWebDB' is highlighted with a green background. An arrow points from this component to the 'Visualizador', which displays a mobile app screen with a title bar 'Sergio's Page', a 'Nome' field, a 'Mensagem' field, and a 'Enviar' button. The 'Componentes' list on the right shows a hierarchy starting with 'Screen1', followed by several UI components like 'txt\_Mensagem', 'Legenda1', 'txt\_nome', 'Legenda2', 'txtidDetecta', 'btn\_Enviar', and 'temporizador1'. At the bottom of the 'Componentes' list, 'TinyWebDB1' is added with a green arrow icon. The 'Visualizador' also shows a status bar at the top with signal, Wi-Fi, and battery icons, and a time of 9:40. The bottom of the screen shows the Android navigation bar with back, home, and recent apps buttons.

Vamos aos códigos..  
. De blocos é claro ;)



Podemos melhorar a  
atualização das  
mensagens



Infelizmente estão atualizando a plataforma e pode sair do ar. Foram introduzidas limitações: ~~250~~ campos até 500 char.

## App Inventor for Android: Tiny WebDB Service



### **NOTE: This service is being modified. It might go offline without notice**

This demonstration Web service is designed to work with [App Inventor for Android](#) and the [TinyWebDB](#) component. The site is designed for use by applications running on the phone (via JSON requests). You can also invoke the [get](#) and [store](#) operations by hand from this Web page to test the API, and also delete individual entries.

This service is only a demo. The database will store at most 250 entries; adding entries beyond that will cause the oldest entries to be deleted. Also, individual data values are limited to at most 500 characters.

The source code for this service, designed to run on Google AppEngine, is included in the [App Inventor](#) documentation. You can use this implementation as a model for deploying your own services, with larger capacity and additional features, and build applications that use the [TinyWebDB](#) component to talk to your service.

### Available calls:

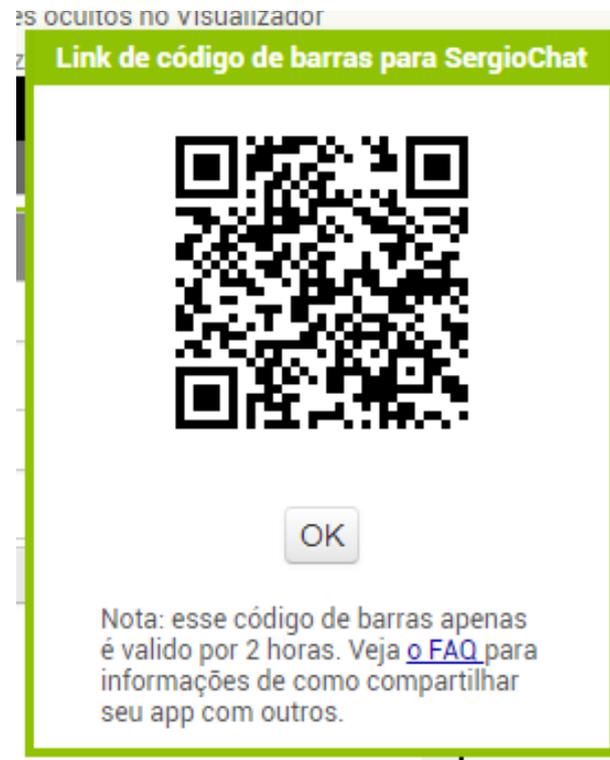
- [/storeavalue](#): Stores a value, given a tag and a value
- [/getvalue](#): Retrieves the value stored under a given tag. Returns the empty string if no value is stored

Para finalizar,  
vamos ver como  
distribuir nosso APK

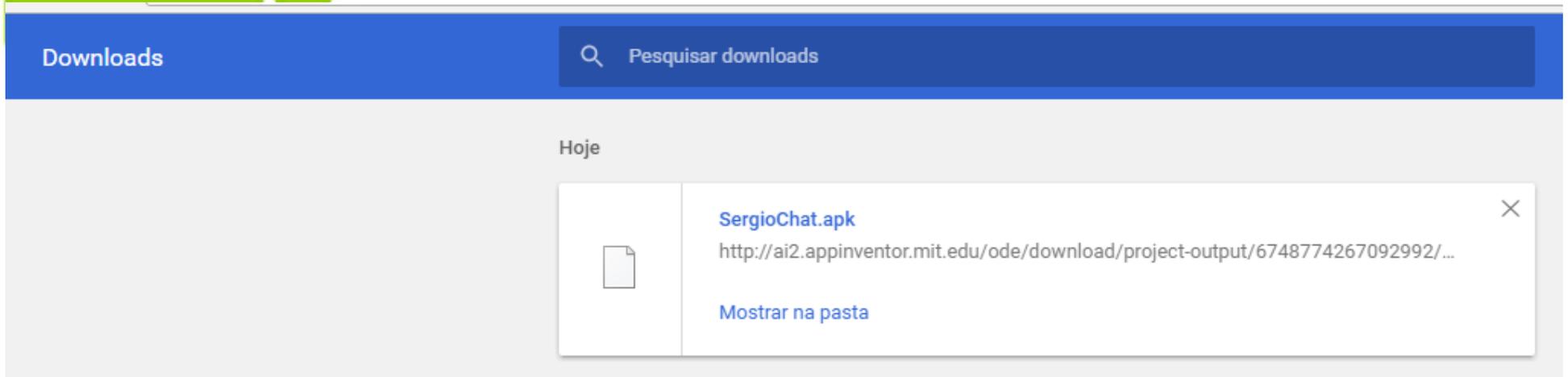
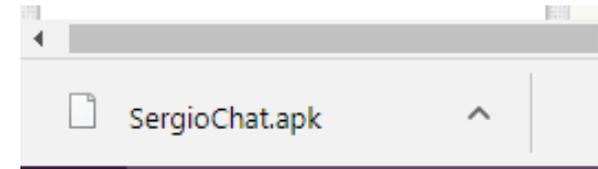


A screenshot of the MIT App Inventor web interface. The top navigation bar includes the MIT App Inventor logo, 'Projetos', 'Conectar', 'Compilar', and 'Ajuda'. The 'Compilar' menu is open, showing two options: 'App ( fornecer o QR code para o .apk )' and 'App ( salvar .apk em meu computador )'. The main workspace is titled 'SergioChat' and is divided into a 'Paleta' (left) and a 'Visualizador' (right). The 'Paleta' contains a list of UI components: Botão, CaixaDeSeleção, EscolheData, Imagem, Legenda, EscolheLista, VisualizadorDeListas, Notificador, and CaixaDeSenha. The 'Visualizador' shows a preview of the chat application with a header 'Sergio's Papo', a text input field for 'Nome:', another for 'Mensagem:', and an 'Enviar' button. A status bar at the top of the preview shows the time as 9:48.

Existem duas opções:  
A primeira fornece um  
QRCode para download  
direto no aparelho



A segunda é fazendo o download do APK para o PC e distribuir posteriormente



Espero que tenham aproveitado o mini-curso.

Qualquer dúvida:

[portari@gmail.com](mailto:portari@gmail.com)

<https://www.facebook.com/sergioportari>

