

# Oficina MIT App Inventor 2 + Arduino: Resumo Básico

Ítalo Travenzoli (UFMG), Letícia Cherchiglia (UFMG) e Prof. Dr. Chico Marinho (EBA/UFMG)

24 a 27 de Junho de 2014

## 1. Objetivo

Essa oficina tem como objetivo despertar no público-alvo o interesse por programação e computação física através de ferramentas de código livre (software e hardware livres). Ao final, será produzido um aplicativo para dispositivos móveis Android que possa controlar um dispositivo elétrico/eletrônico (robô/carro).

## 2. Material necessário

- **Computador:** com Windows 7 + internet (wi-fi ou cabeada);
  - **MIT App Inventor 2** – requisitos: internet + conta gmail;
  - **Arduino** – requisitos: conhecimento intermediário de programação + componentes elétricos e eletrônicos (bateria, servos motores etc);
- **Smartphone** – Android (acima de 2.3) + cabo USB (se não houver internet wi-fi disponível);

## 3. Apresentação do MIT App Inventor 2

- Ambiente de programação gratuito e online, desenvolvido por pesquisadores do MIT (Instituto de Tecnologia de Massachusetts/EUA). Sua vantagem principal é tornar o desenvolvimento de aplicativos algo muito simples e intuitivo.
- **Básico:**
  - **Como entrar e criar/ver projetos:**
    1. Entre no site: <http://ai2.appinventor.mit.edu/>.
    2. Faça o login com seu email do gmail (se não tiver um, será necessário criá-lo).
    3. Clique no botão “Allow” para entrar no ambiente.
    4. Na primeira tela que aparecer, clique em “Never take survey”.
    5. Na seguinte, clique em “Continue”.
    6. Para criar um novo projeto, clique em “New Project”. Depois de escrever o nome do projeto, clique em “OK”.
    7. Para ver seus projetos, clique na aba superior em “Project” > “My projects” e selecione o projeto desejado da lista.
  - **Como sair:** clique no seu email, no canto superior direito do ambiente, e depois em “Sign Out”.
- **Como funciona o ambiente de programação - modo DESIGNER:**
  - Permite a escolha, personalização e colocação em tela dos componentes necessários para o aplicativo (botões, imagens etc).
  - **Lado esquerdo do ambiente (Palette):**
    - **User Interface:** possui componentes como botões, imagens etc;
    - **Layout:** possui componentes para alinhamento de outros componentes na tela;
    - **Media:** possui componentes para acesso de som, vídeo, câmera, reconhecimento de voz etc;
    - **Drawing and Animation:** possui componentes para desenho e movimento de componentes;
    - **Sensors:** possui componentes para acesso de acelerômetro, GPS etc;
    - **Social:** possui componentes para acesso de email, lista de contatos do telefone, twitter etc;
    - **Storage:** possui componentes para banco de dados e arquivos;
    - **Connectivity:** possui componentes para conexão com outros aplicativos, web ou Bluetooth;
    - **LEGO® MINDSTORMS®:** possui componentes para conexão com LEGO robótico;

- **Centro do ambiente (Viewer):**
  - Componentes visíveis (botões, imagens etc): ficam na própria tela do aplicativo.
  - Componentes invisíveis (reconhecimento de som, acelerômetro etc): são mostrados abaixo do escrito “Non-visible components”.
  - Para adicionar um componente, arraste-o do lado esquerdo do ambiente de programação para a tela do aplicativo (centro/Viewer).
  - Arraste componentes para alterar a posição desses na tela.
  - Quando um componente é selecionado, são mostradas as propriedades deste à direita (Properties). É possível alterar as propriedades de modo personalizado (cor, conteúdo, alinhamento do texto etc).
- **Lado direito do ambiente (Components):**
  - Lista de todos os componentes de seu projeto.
  - Para renomear algum, clique nele e em seguida no botão “Rename”.
  - Para deletar, selecione e clique no botão “Delete”.
- **Lado direito do ambiente (Properties):**
  - Propriedades do componente selecionado, que variam de acordo com o componente, permitindo sua personalização.
  - Ex: mudar cor de texto, mudar conteúdo de rótulo, mudar cor de fundo, carregar uma imagem, carregar um som etc.

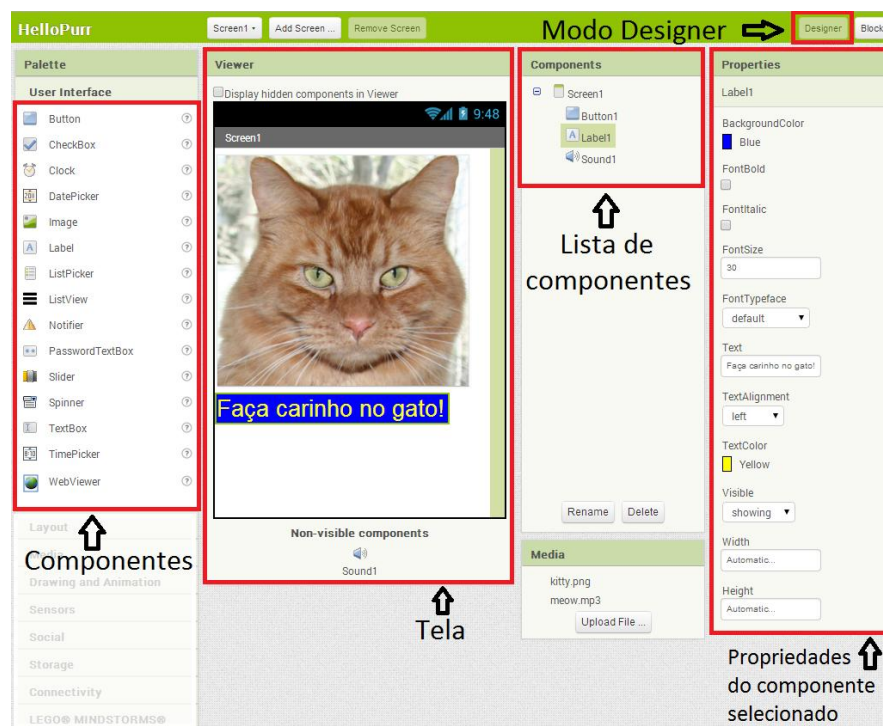


Figura 1- Vista do ambiente de programação no modo Designer

- **Componentes mais utilizados nessa oficina:**
  - **Botão** (Button em User Interface): propriedade de mudar texto (Text), cor de fundo (BackgroundColor) e cor do texto (TextColor);
  - **Rótulo** (Label em User Interface): propriedade de mudar texto (Text), cor de fundo (BackgroundColor), cor do texto (TextColor) e tamanho do texto (FontSize);
  - **Bluetooth** (Bluetooth Client em Connectivity);
  - **Lista de objetos** (ListPicker em User Interface): propriedade de mudar texto (Text);
  - **Mensagem de notificação** (Notifier em User Interface);
  - **Reconhecimento de fala** (SpeechRecognizer em Media);
  - **Sensor de aceleração** (AccelerometerSensor em Sensors);

➤ **Como funciona o ambiente de programação - modo BLOCKS:**

- Permite a escolha de blocos de programação, que podem ser ligados de forma lógica, para definir o comportamento dos componentes do modo DESIGNER.
- **Lado esquerdo do ambiente (Blocks):**
  - **Built-in:** blocos gerais de programação.
    - **Control:** possui blocos para controle (desvio condicional, loop, eventos);
    - **Logic:** possui blocos lógicos (resultado sempre será verdadeiro ou falso);
    - **Math:** possui blocos para matemática (aritmética, geometria etc);
    - **Text:** possui blocos para tratamento de texto (cortar texto, comparar texto etc);
    - **Lists:** possui blocos para criação e manipulação de listas de elementos;
    - **Colors:** possui blocos para cores;
    - **Variables:** possui blocos para criação e manipulação de variáveis;
    - **Procedures:** possui blocos para criação e manipulação de funções.
  - **Screen1:** lista os componentes do modo DESIGNER que estão na tela Screen1. Ao clicar em um deles é possível ver os blocos de programação disponíveis especificamente para aquele componente. Ex: componente “button1” (botão1) tem um bloco para clique que só funcionará quando o botão1 for clicado etc.
  - **Any component:** lista os componentes genéricos dos tipos utilizados na sua tela Screen1. Ao clicar em um deles é possível ver os blocos de programação disponíveis de modo geral para aquele tipo de componente. Ex: componente “any Button” (qualquer botão) tem bloco para clique que definirá um comportamento para quando todos os botões forem clicados, seja o botão1 ou o botão2 etc.
  - **Media:** lista as mídias disponíveis no projeto (vídeos, imagens, sons etc).
- **Centro do ambiente (Viewer):**
  - Mostra os blocos de programação existentes e suas conexões.
  - Para adicionar um bloco, arraste-o do lado esquerdo do ambiente de programação (Blocks) para o espaço em branco no centro da tela (Viewer).
  - Arraste blocos para ligá-los, o que irá definir o comportamento lógico dos componentes.
  - **IMPORTANTE:** os blocos são ligados de acordo com suas formas (é como montar um quebra-cabeça). Quando um bloco pode ser conectado a outro, o local da conexão fica amarelo/alaranjado. Se não ficar, não será conectado.
  - Para remover um bloco, selecione-o e aperte “delete” no teclado do computador ou arraste-o para o ícone de lixeira presente do lado inferior direito.

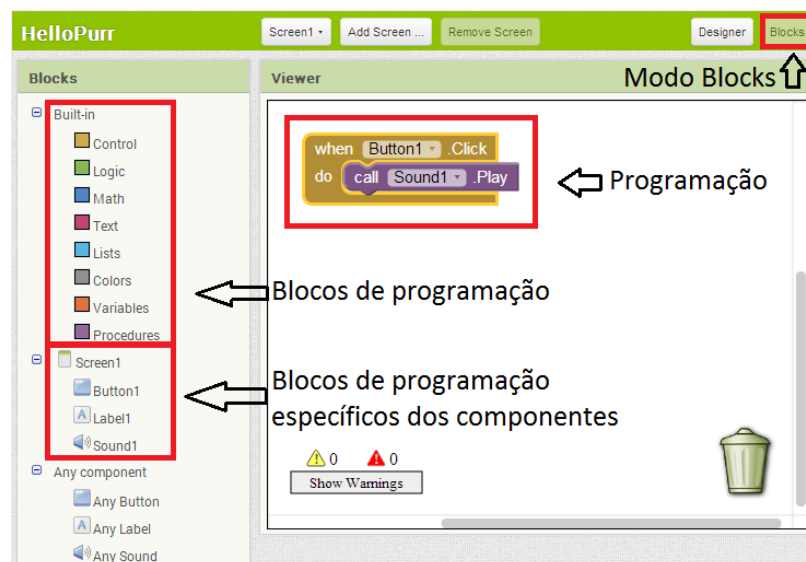
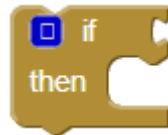
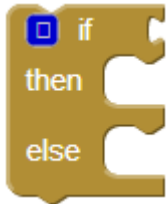


Figura 2 - Vista do ambiente de programação no modo Blocks

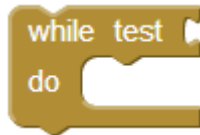
- Lista de blocos de programação mais utilizados nessa oficina:



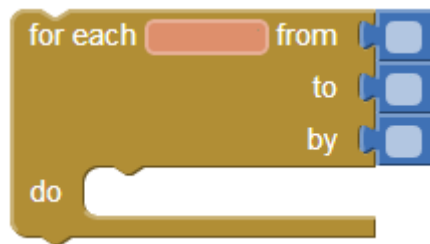
**Se** (condição for verdadeira ou falsa) **então** (faça instrução)



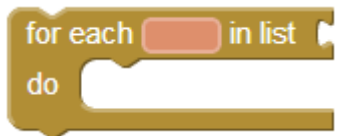
**Se** (condição for verdadeira ou falsa) **então** (faça instrução) **senão** (faça outra instrução)



**Enquanto o teste** (de comparação, igualdade etc ocorrer) **faça** (instrução)



**Para cada** (variável auxiliar) **de** (valor1) **até** (valor2) **variando por** (valor3) **faça** (instrução)



**Para cada** (variável auxiliar) **na lista** (lista) **faça** (instrução)



**Quando** (componente).(algum evento daquele componente ocorrer) **faça** (instrução)



**Mude** (componente).(alguma propriedade do componente) **para** (nova propriedade)



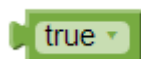
**Chame** (função).(evento)



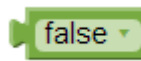
**Inicialize variável** (nome da variável) **com o valor** (valor)



**Retorne** (o valor guardado na variável)



**Verdadeiro**



**Falso**



**Texto**



**Valor**

- **Como testar o aplicativo no computador (modo EMULATOR):**
  1. Baixe o emulador em: [http://appinv.us/aisetup\\_windows](http://appinv.us/aisetup_windows)
  2. Abra esse arquivo e instale-o
  3. Abra o programa “aiStarter.exe” (geralmente fica em "C:\Arquivo de Programas\ApplInventor")
  4. No ambiente de programação clique em “Connect” e depois em “Emulador”
  5. Aguarde até que seja aberta uma janela simulando a tela de seu smartphone.
  6. Obs: pode ser que apareça uma janela no ambiente de programação atrás da janela do emulador. Caso positivo, clique no botão “Not now”
  7. Depois de fechar a janela do emulador, clique em “Connect” e depois em “Reset Connection”.
  
- **Como testar o aplicativo no dispositivo móvel (modo USB):**
  1. Conecte o smartphone ao computador através de um cabo USB
  2. No smartphone, clique em Configurações e deixe selecionadas as “Opções de Desenvolvedor”. Tenha certeza de que a opção "USB Debugging" está selecionada (Android 3.2+: Configurações > Aplicações > Desenvolvimento; Android 4.0+: Configurações > Opções de Desenvolvedor; Android 4.2+: Configurações > Sobre o telefone > clicar 7x em Build number > retornar para Opções de Desenvolvedor)
  3. No ambiente de programação clique em “Connect” e depois em “USB”
  4. Aguarde até que o aplicativo apareça na tela de seu smartphone

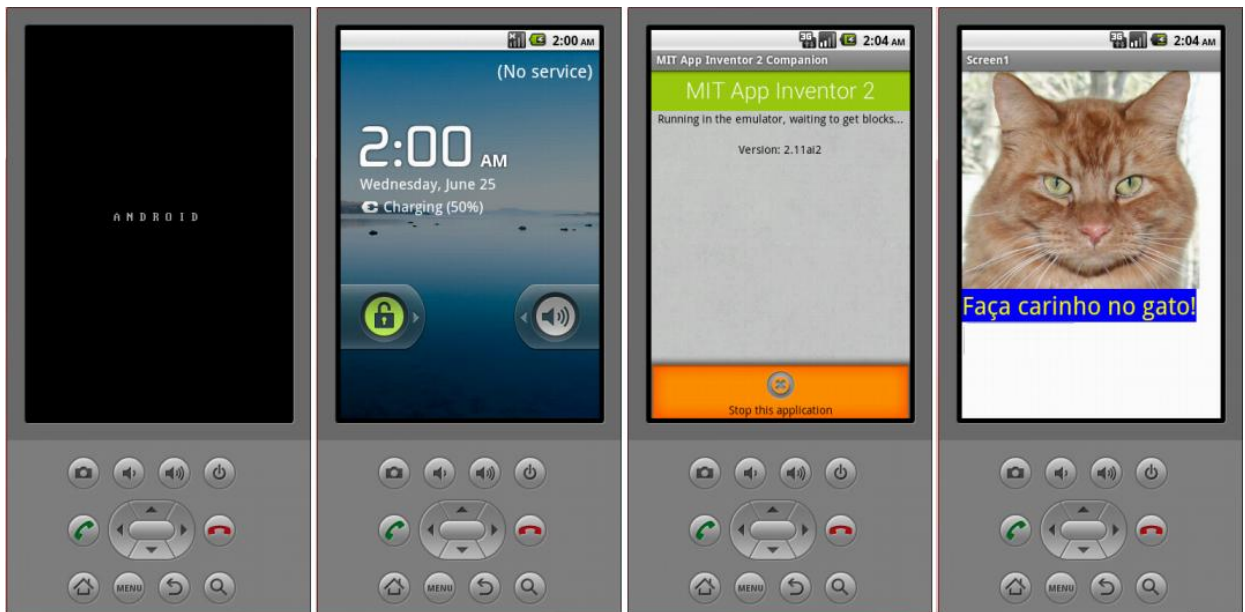


Figura 3 - Progressão de telas que irão aparecer na janela no modo Emulador

- **Como testar o aplicativo no dispositivo móvel (modo AI Companion):**
  1. Baixe o aplicativo Android MIT AI2 Companion na Google Play para o seu smartphone: ([https://play.google.com/store/apps/details?id=edu.mit.appinventor.aicompanion3&hl=pt\\_BR](https://play.google.com/store/apps/details?id=edu.mit.appinventor.aicompanion3&hl=pt_BR))
  2. Abra o MIT AI2 Companion no smartphone
  3. No ambiente de programação clique em “Connect” e depois em “AI Companion”
  4. Será gerado um código de 6 dígitos e um QR code
  5. No smartphone é possível entrar com o código ou simplesmente com o QR code, bastando selecionar esta opção (scan QR code) e apontar a câmera do celular para a tela do computador
  
- **Como gerar o aplicativo para o dispositivo móvel (QR Code):**
  1. Clique em “Build” e depois em “App (provide QR code for .apk)”
  2. Aponte a câmera do celular para a tela do computador e faça download do aplicativo

- **Como gerar o aplicativo para o dispositivo móvel (.apk):**
  1. Clique em “Build” e depois em “App (save .apk to my computer)”
  2. Transfira o arquivo .apk para seu smartphone
  3. No seu smartphone, instale o aplicativo. Obs: é necessário habilitar a opção de instalação de fontes desconhecidas no Android (Configurações > Segurança > Fontes desconhecidas)
- **Links úteis (em inglês):**
  - **Resoluções de problemas:** <http://appinventor.mit.edu/explore/ai2/support/troubleshooting.html>
  - **Tutoriais:** <http://appinventor.mit.edu/explore/ai2/tutorials.html>

#### 4. Apresentação do Arduino

- Plataforma de prototipagem eletrônica de hardware livre, projetada na Itália. Possui uma linguagem de programação padrão, que é essencialmente C/C++. O objetivo do projeto é criar ferramentas que são acessíveis, com baixo custo, flexíveis e fáceis de serem usadas por artistas e amadores.
- Download: <http://arduino.cc/en/Main/Software>