

# Introduction to MIT App Inventor 2

Dr. Farid Farahmand

Nov 18, 2015

# Getting Started

- AppInventor.org  
<http://www.appinventor.org/> →
  - LOGIN with your Google account
  - Follow the tutorial & build “I Have A Dream” App
  - Follow the course!

# Getting Started



appinventor.org

Learn ▾

Teach

About

🔍 Search Appinventor.org

← Modules

Introduction

📄 Course Overview

📄 Introduction

▶ "I Have a Dream" Tutorial

📄 "I Have A Dream" Worksheet

📄 Intro to App Inventor (lecture slides)

📄 What is App Inventor?

📄 Setup App Inventor

📄 App Maker Cards

## The App Inventor Course-in-a-Box

### Self-Directed Learners

Never coded but want to learn how to build apps? Then you've come to the right place. This course targets beginners of all ages and starts at the very beginning--setting up App Inventor. With video and text-based lessons, Professor Wolber will step you through building progressively more complex apps. You'll learn how to build many types of apps and you'll learn programming concepts and terminology.

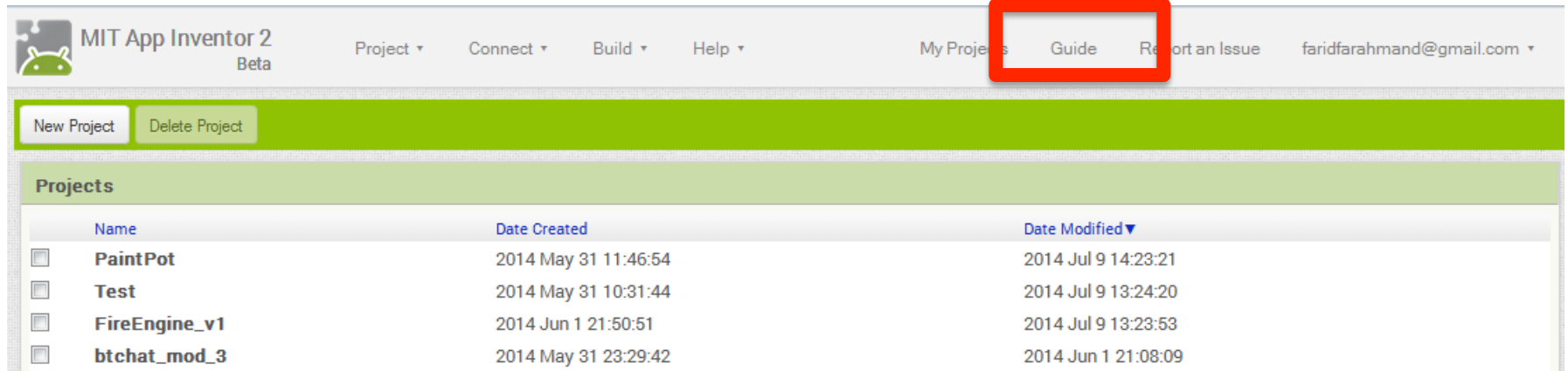
[Start learning with the "I Have Dream" Tutorial →](#)

### Modules

[Module 1: Introduction to AI2 and Event-Driven Programming](#)

# Your Project Site

More Example!



The screenshot shows the MIT App Inventor 2 Beta web interface. The top navigation bar includes the MIT App Inventor logo, the text "MIT App Inventor 2 Beta", and several menu items: "Project", "Connect", "Build", "Help", "My Projects", "Guide", "Report an Issue", and the user's email "faridfarahmand@gmail.com". The "Guide" link is highlighted with a red box. Below the navigation bar is a green bar with "New Project" and "Delete Project" buttons. The main content area is titled "Projects" and contains a table with the following data:

Name	Date Created	Date Modified
<input type="checkbox"/> PaintPot	2014 May 31 11:46:54	2014 Jul 9 14:23:21
<input type="checkbox"/> Test	2014 May 31 10:31:44	2014 Jul 9 13:24:20
<input type="checkbox"/> FireEngine_v1	2014 Jun 1 21:50:51	2014 Jul 9 13:23:53
<input type="checkbox"/> btchat_mod_3	2014 May 31 23:29:42	2014 Jun 1 21:08:09

# Start A New Project - TalktoMe

The screenshot displays the MIT App Inventor 2 Beta web interface. At the top left, the MIT App Inventor logo and 'Beta' version are visible. The top navigation bar includes 'Projects', 'Connect', 'Build', and 'Help' menus. On the right, there are links for 'My Projects', 'Gallery', 'Guide', 'Report an Issue', 'English', and the user's email 'faridfarahmand@gmail.com'. The main workspace is divided into four panels: 'Palette', 'Viewer', 'Components', and 'Properties'. The 'Palette' panel on the left shows a 'User Interface' section with various widgets like Button, TextBox, ListView, DatePicker, TimePicker, CheckBox, Label, ListPicker, Slider, PasswordTextBox, Notifier, Image, WebViewer, and Spinner. The 'Viewer' panel in the center shows a mobile device preview with a status bar at the top displaying 'Screen1' and the time '9:48'. The 'Components' panel on the right shows a single component named 'Screen1'. The 'Properties' panel on the far right lists properties for 'Screen1', including 'AboutScreen', 'AlignHorizontal' (set to 'Left'), 'AlignVertical' (set to 'Top'), 'AppName' (set to 'TalktoMe'), 'BackgroundColor' (set to 'White'), 'BackgroundImage' (set to 'None...'), 'CloseScreenAnimation' (set to 'Default'), and 'Icon' (set to 'None...').

MIT App Inventor 2  
Beta

Projects ▾ Connect ▾ Build ▾ Help ▾

My Projects Gallery Guide Report an Issue English ▾ faridfarahmand@gmail.com ▾

**TalktoMe** Screen1 ▾ Add Screen ... Remove Screen

Designer Blocks

**Palette**

User Interface

- Button
- TextBox
- ListView
- DatePicker
- TimePicker
- CheckBox
- Label
- ListPicker
- Slider
- PasswordTextBox
- Notifier
- Image
- WebViewer
- Spinner

**Viewer**

Display hidden components in Viewer  
 Check to see Preview on Tablet size.

Screen1

9:48

**Components**

Screen1

**Properties**

Screen1

AboutScreen

AlignHorizontal  
Left ▾

AlignVertical  
Top ▾

AppName  
TalktoMe

BackgroundColor  
 White

BackgroundImage  
None...

CloseScreenAnimation  
Default ▾

Icon  
None...

# Complete the Interface – Building the Designer

Designer

Blocks

Screen1 Add Screen ... Remove Screen

Palette

User Interface

- Button
- TextBox
- ListView
- DatePicker
- TimePicker
- CheckBox
- Label
- ListPicker
- Slider
- PasswordTextBox
- Notifier
- Image
- WebView
- Spinner

Viewer

Display hidden components in Viewer

Check to see Preview on Tablet size.

Screen1

My Name

Sonoma State University

SEAWOLVES

SONOMA STATE UNIVERSITY

Components

- Screen1
  - Button1
  - Button2
  - Image1
  - AccelerometerSensor1
  - TextToSpeech1

Button2

BackgroundColor

Default

Enabled

FontBold

FontItalic

FontSize

14.0

FontTypeface

default

Height

Automatic...

Width

Automatic...

Image

None...

9:48

Add an image

Add a button

Your Components

Component Property

# Build A Quick App - Blocks

**TalktoMe2\_V2** Screen1 Add Screen ... Remove Screen

**Blocks**

- Built-in
  - Control
  - Logic
  - Math
  - Text
  - Lists
  - Colors
  - Variables
  - Procedures
- Screen1
  - Button1
  - AccelerometerSensor1
  - TextToSpeech1
- Any component
  - Any AccelerometerSensor
  - Any Button

**Viewer**

when AccelerometerSensor1 .Shaking  
do call TextToSpeech1 .Speak message " Stop shaking me! "

when Button1 .Click  
do call TextToSpeech1 .Speak message " My Name is Fareed "

This is the Design

Click on TextToSpeech1 under Screen1, and drag this here!

Get this from Button1 (under Screen1) and drag it here!

This is just a "text String". Go to TEXT under Built-in and pick the first block. You can change the text to anything you like!

# Creating the QR Code for the Package

The screenshot shows the MIT App Inventor interface for an application named "TalktoMe2\_V2". At the top, there are buttons for "Screen1" and "Add Screen ...". A red box highlights a dropdown menu with two options: "App ( provide QR code for .apk )" and "App ( save .apk to my computer )".

The interface is divided into two main sections: "Blocks" on the left and "Viewer" on the right.

**Blocks Panel:**

- Built-in
  - Control
  - Logic
  - Math
  - Text
  - Lists
  - Colors
  - Variables
  - Procedures
- Screen1
  - Button1
  - AccelerometerSensor1
  - TextToSpeech1

**Viewer Panel:**

The code blocks in the viewer are:

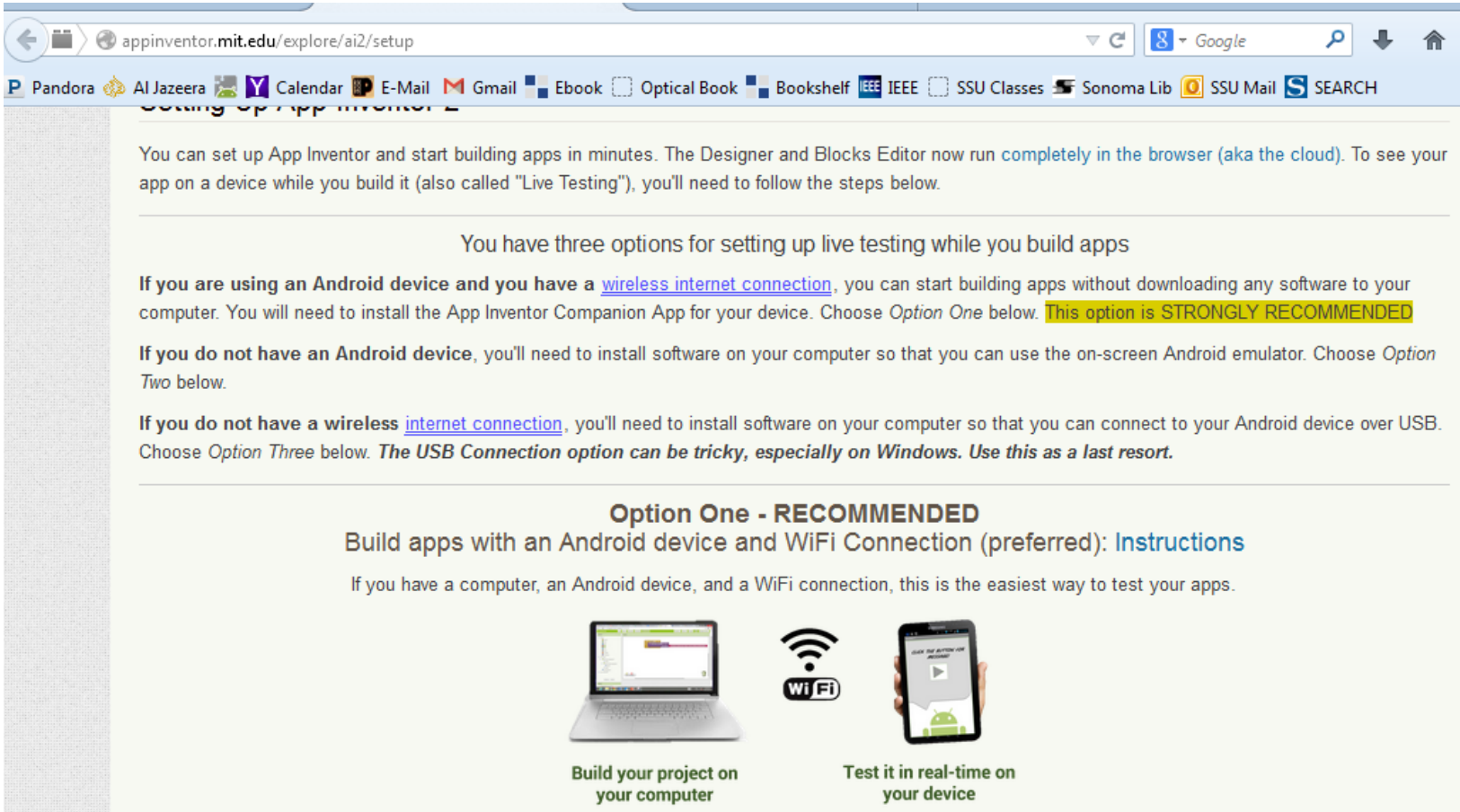
```
when AccelerometerSensor1 .Shaking
do
  call TextToSpeech1 .Speak
  message " Stop shaking me! "
```

```
? when Button1 .Click
do
  call TextToSpeech1 .Speak
  message " My Name is Fae-Reed "
```



# Programming Your Device

Must download MIT AI@ Companion on your mobile device



The screenshot shows a web browser window with the URL `appinventor.mit.edu/explore/ai2/setup`. The browser's address bar and search bar are visible. Below the browser window, the page content is displayed on a light green background. It starts with a heading "Setting up App Inventor" and a paragraph explaining that the Designer and Blocks Editor run in the browser. It then lists three options for setting up live testing. Option One is highlighted as recommended and includes icons for a laptop, a Wi-Fi symbol, and a smartphone.

You can set up App Inventor and start building apps in minutes. The Designer and Blocks Editor now run **completely in the browser (aka the cloud)**. To see your app on a device while you build it (also called "Live Testing"), you'll need to follow the steps below.

You have three options for setting up live testing while you build apps




If you are using an Android device and you have a [wireless internet connection](#), you can start building apps without downloading any software to your computer. You will need to install the App Inventor Companion App for your device. Choose *Option One* below. **This option is STRONGLY RECOMMENDED**

If you do not have an Android device, you'll need to install software on your computer so that you can use the on-screen Android emulator. Choose *Option Two* below.

If you do not have a wireless [internet connection](#), you'll need to install software on your computer so that you can connect to your Android device over USB. Choose *Option Three* below. *The USB Connection option can be tricky, especially on Windows. Use this as a last resort.*

**Option One - RECOMMENDED**  
Build apps with an Android device and WiFi Connection (preferred): [Instructions](#)

If you have a computer, an Android device, and a WiFi connection, this is the easiest way to test your apps.

Build your project on your computer      Test it in real-time on your device

# Program Your Device

When you start the Companion on your phone, it will look like (1). Back in App Inventor, choose "Connect" (2) then "AI Companion". This will cause a QR code to appear (3) You can then scan the QR code (4) with your phone to see your app live. NOTE: for live testing to work, **both your computer and phone/tablet must be connected to the same WiFi station**. Using WiFi is the easiest way to connect, but if you're at school/work you may have firewall issues.



# Creating the QR Code for the Package

The screenshot shows the MIT App Inventor interface for an application named "TalktoMe2\_V2". The interface is divided into three main sections: a top toolbar, a left-hand "Blocks" palette, and a central "Viewer" area.

**Top Toolbar:** Contains a dropdown menu set to "Screen1", and two buttons: "Add Screen ..." and "Remove Screen".

**Blocks Palette (Left):** Lists various built-in components categorized by color: Control (orange), Logic (green), Math (blue), Text (pink), Lists (light blue), Colors (grey), Variables (orange), and Procedures (purple). Below these are specific components for the current screen: Screen1 (grey), Button1 (blue), AccelerometerSensor1 (grey), and TextToSpeech1 (blue).

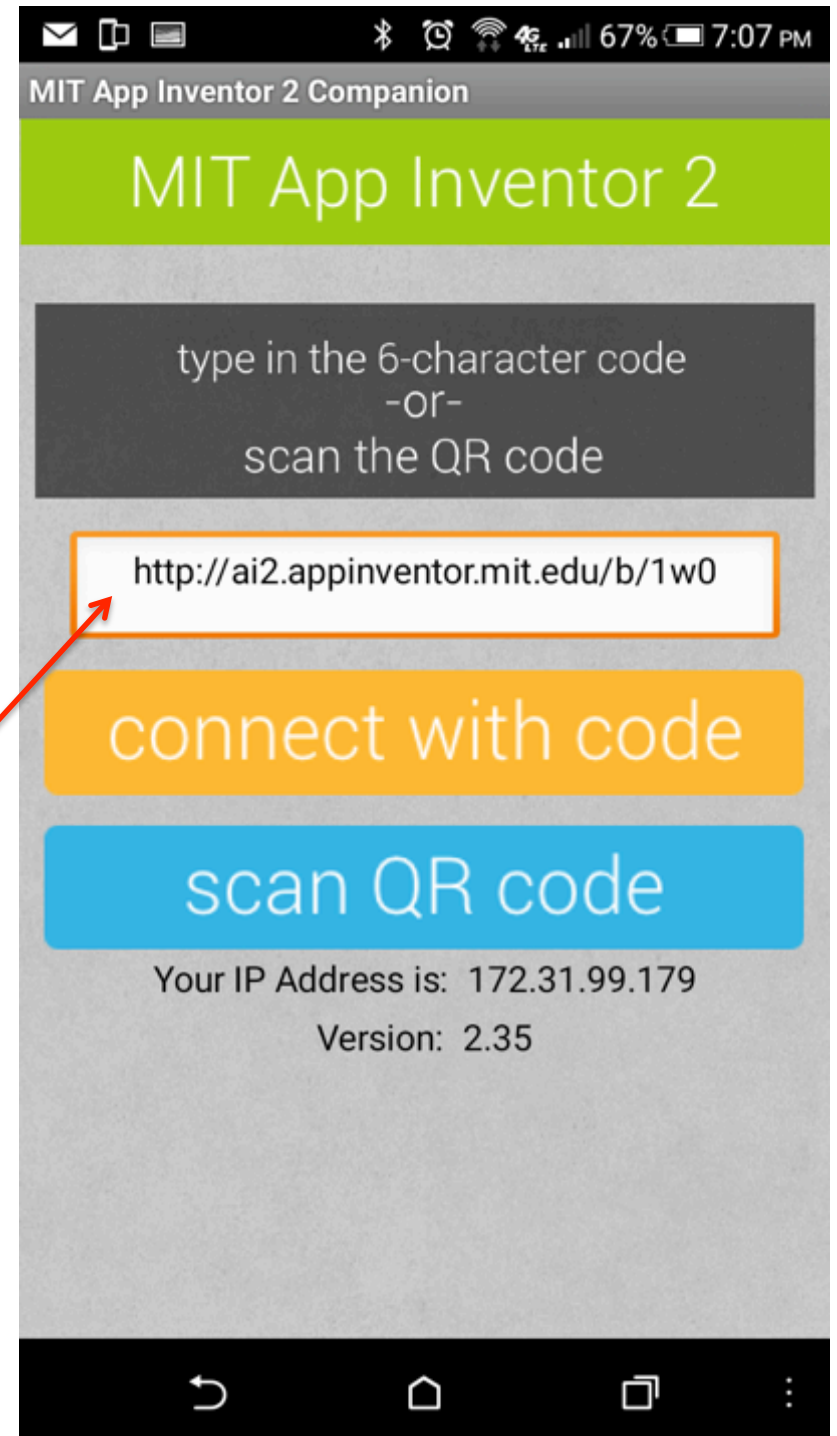
**Viewer Area (Center):** Displays the visual representation of the app's logic. It shows two event-driven code blocks:

- The top block is a "when AccelerometerSensor1 .Shaking" event. The "do" block contains a "call TextToSpeech1" procedure.
- The bottom block is a "when Button1 .Click" event. The "do" block contains a "call TextToSpeech1" procedure.

**QR Code Dialog (Right):** A modal dialog box titled "Barcode link for TalktoMe2\_V2" is open. It features a large QR code in the center. Below the QR code is an "OK" button. At the bottom of the dialog, there is a note: "Note: this barcode is only valid for 2 hours. See [the FAQ](#) for info on how to share your app with others."

# On Your Android

- Google Play →  
Download MIT AI2  
Companion
- Open the application
- Make sure you get a URL  
address
- Press **Scan QR Code**
- Then press **Connect with  
Code**



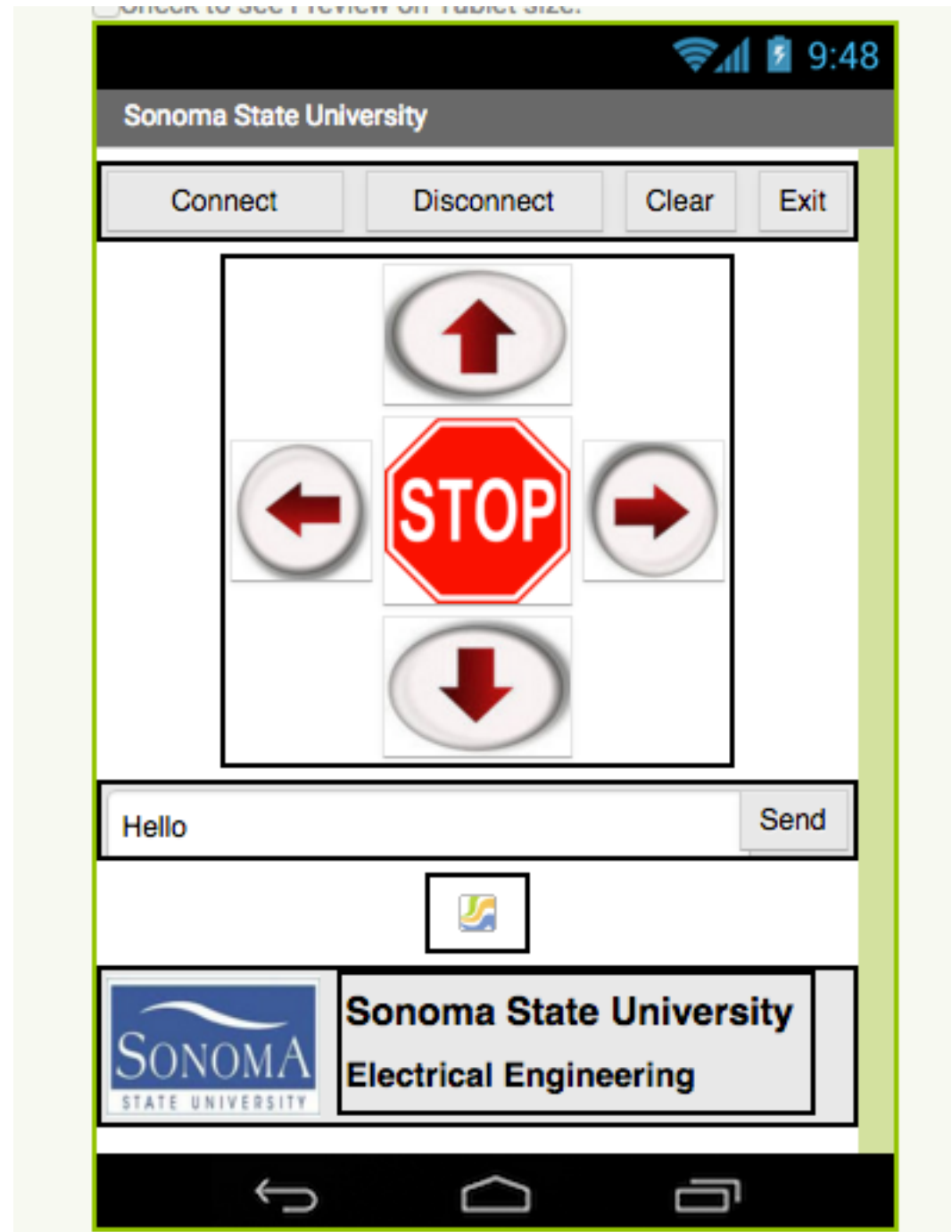
# Your APP

- Make sure your phone is setup to download applications from non-secure sources!
- Find the downloaded APP on your phone and run it!
- This is how your app looks like:



Another Example....

# USB Interface



# The Designer

Viewer

```
initialize global boolServer to false
when Screen1.Initialize do
  when lipConnect .AfterPick...
  when btnSend .Click do cal...
  to init do call
  when BluetoothServer1 .Con...
  when lipConnect .BeforePic...
  to send message do if get...
  when Clock1 .Time to print textToPrint do s...
  to connected trueOrFalse ...
  when Screen1 .ErrorOccurs
  to disconnect do call
  when btnDisconnect .Click ...
  when btnExit .Click do cal...
```

when btnLeft .Click do call send message " L "

when btnRight .Click do call send message " R "

when btnUp .Click do call send message " U "

when btnDown .Click do call send message " D "

when btnStop .Click do call send message " S "

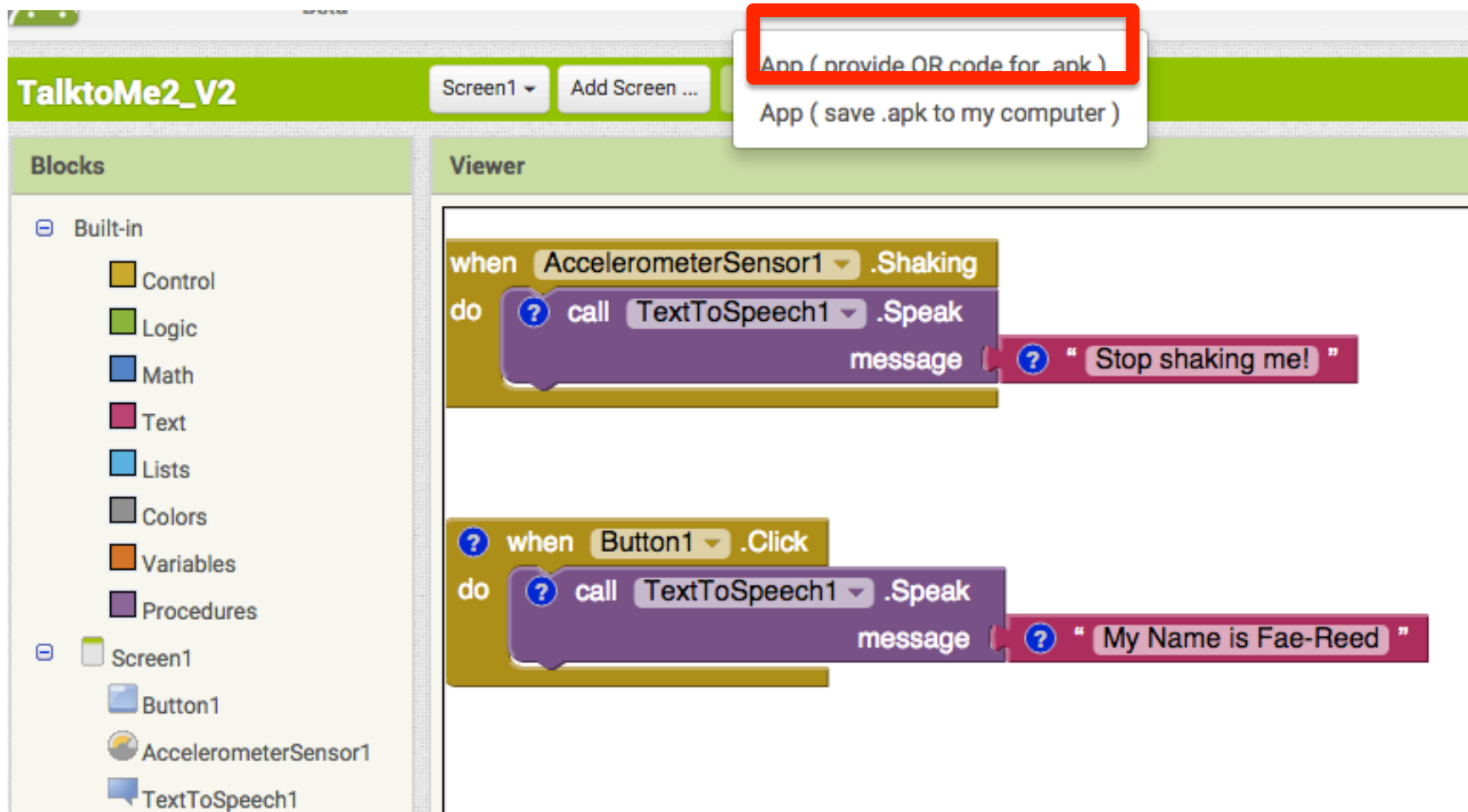
Event raised when an error occurs. Only some errors will raise this condition. For those errors, the system will show a notification by default. You can use this event handler to prescribe an error behavior different than the default.

0 0  
Hide Warnings



# Download the App on your phone!

- 1-Create the .apk
- 2-Upload to your phone using USB cable
- 3-Run the application on the phone



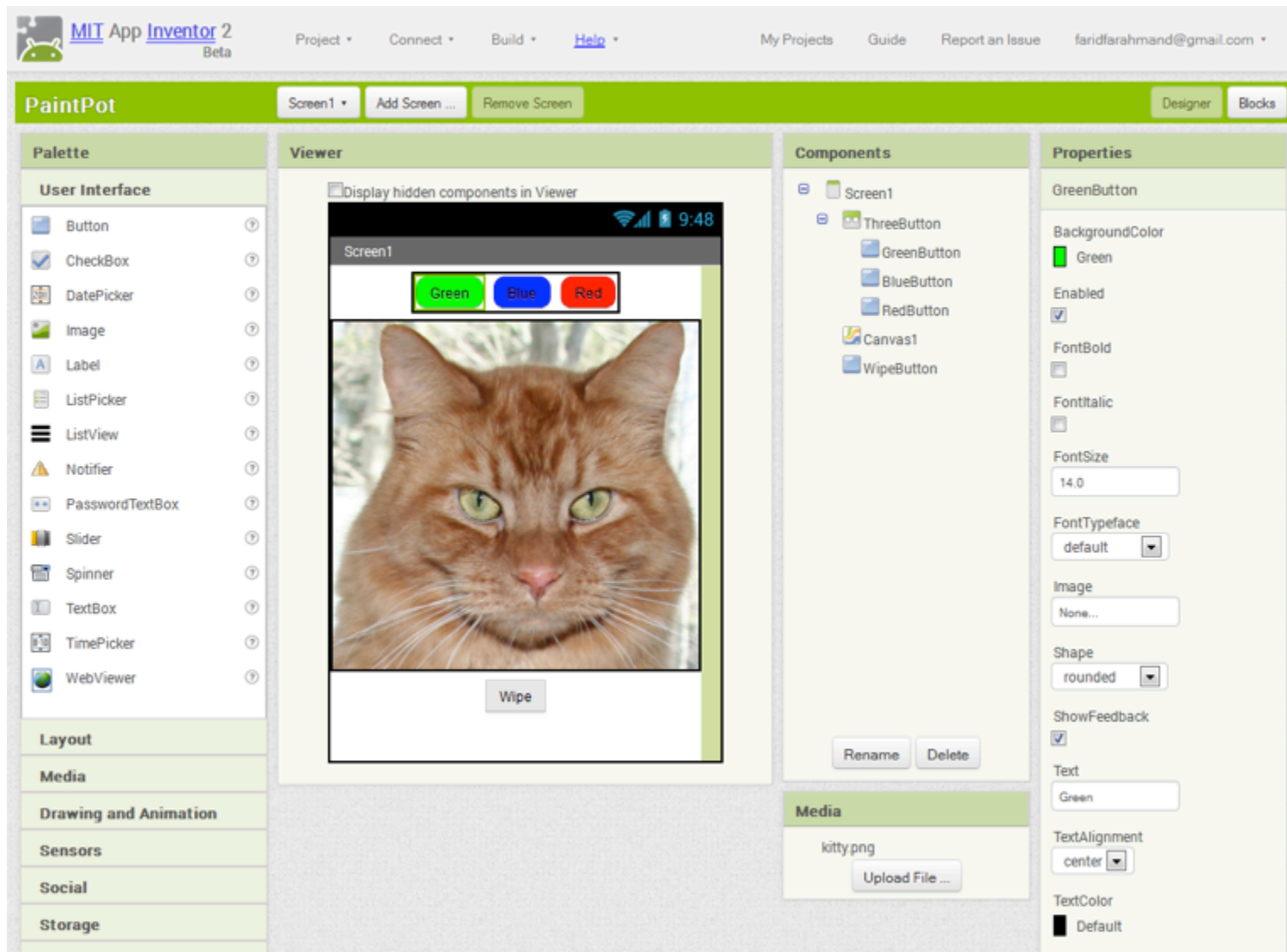
# Pairing Your Phone & Computer

- Remove the phone device from your BT connections
- Make sure your phone's BT is on and it is discoverable
- Make sure your computer is not paired with any other devices (mouse, keyboard, etc.)

# Basic Idea

Making your first APP:

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



# Block Architecture

The screenshot displays the MIT App Inventor 2 Beta interface for a project named "PaintPot". The interface is divided into several sections:

- Top Bar:** Includes the MIT App Inventor 2 Beta logo, navigation links for Project, Connect, Build, and Help, and user information for My Projects, Guide, Report an Issue, and faridfarahmand@gmail.com.
- Project Bar:** Shows "Screen1" as the active screen, with options to "Add Screen ..." and "Remove Screen". It also includes "Designer" and "Blocks" tabs.
- Blocks Panel (Left):** A categorized list of components:
  - Built-in: Control, Logic, Math, Text, Lists, Colors, Variables, Procedures.
  - Screen1: ThreeButton, GreenButton, BlueButton, RedButton, Canvas1, WipeButton.
  - Any component: A catch-all category.Buttons for "Rename" and "Delete" are located below the list.
- Viewer (Right):** A workspace for visualizing the app's components and their logic. It shows:
  - Buttons: RedButton, GreenButton, BlueButton, and WipeButton. Each has a "Click" event block with a "do" block containing a "set Canvas1 . PaintColor to" block with a color swatch (red, green, blue, and a clear icon respectively).
  - Canvas1: A "Touched" event block with a "do" block containing a "call Canvas1 . DrawCircle" block. The "x" and "y" inputs are connected to "get x" and "get y" blocks, and the "r" input is set to 5.
  - Canvas1: A "Dragged" event block with a "do" block containing a "call Canvas1 . DrawLine" block. The "x1", "y1", "x2", and "y2" inputs are connected to "get prevX", "get prevY", "get currentX", and "get currentY" blocks.
- Media Panel (Bottom Left):** Shows a file named "kitty.png" and an "Upload File ..." button.
- Warning Panel (Bottom Center):** Displays "0" warnings and a "Show Warnings" button.
- Trash Can (Bottom Right):** A trash can icon for deleting components.

# References:

- AppInventor.org  
<http://www.appinventor.org/> → LOGIN with your Google account
- Other Tutorials:  
<http://appinventor.mit.edu/explore/ai2/beginner-videos.html>
- Exploring AppInventor  
<http://appinventor.mit.edu/explore/>