Technovation CANADA

Technovation 2017 Hack Day

Hack Day Trainer: Ahmed

Carrer pathway through Engineering

- B. Eng. in Software Engineering
  Lakehead University
- Project Management Office
  IBM (Automation, Legal Deliverables…)
- M. Eng. in Technology Innovation Management
  Carleton University
- Cybersecurity Research and Development
  VENUS Cybersecurity Corp.

3rd year at Technovation!

Welcome to Technovation

- Technovation will teach the skills you need to emerge as tech entrepreneurs and leaders.
- Working in teams of 3-5 you identify a problem in your community, and build a technology business to solve it.
- Technovation takes you through 4 stages of launching a mobile app startup, inspired by the principles of design thinking:

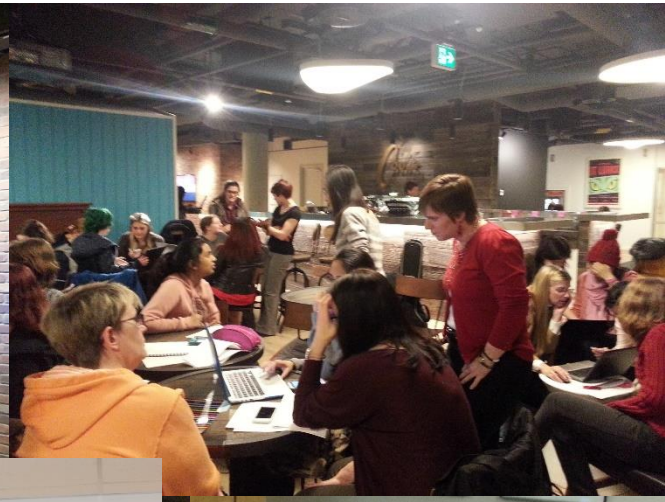
  Ideation - Identify a problem in the community
  Technology - Develop a mobile app solution
  Entrepreneurship - Build a business plan to launch the app
  Pitch - Take the business to market

Technovation Ottawa

Students from across Ottawa

January to April(12 Weeks)






What we are covering today

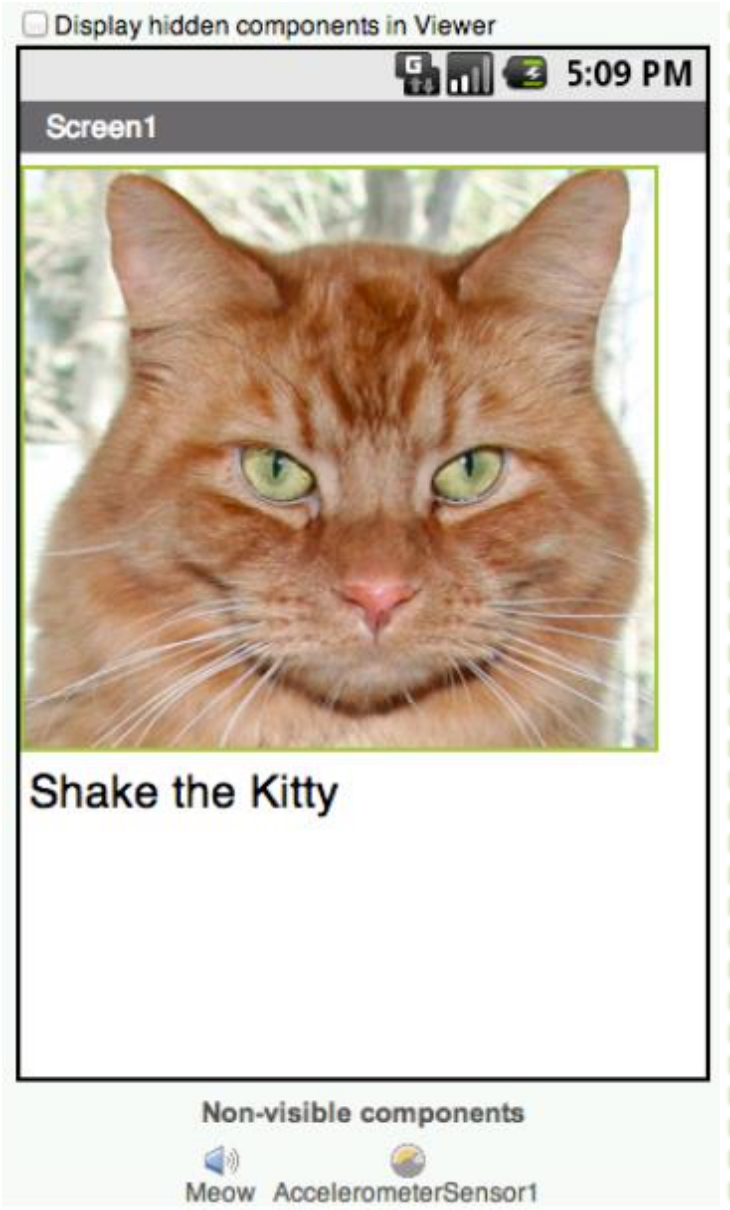
1. What you will need for today
2. Problem solving exercises
3. Application development cycle
4. Setting up your phone
   - Installing AI Companion
5. Tips for mobile development
6. Creating your first app “Talk to Me” (App 1)
   - Tutorial walkthrough
   - Downloading application
   - Save and distribute
7. ----------Lunch----------
8. Slide show (App 2)
9. Colored Dots (App 3)
10. Continue Learning
11. Technovation: Next Steps
12. Math App Challenge

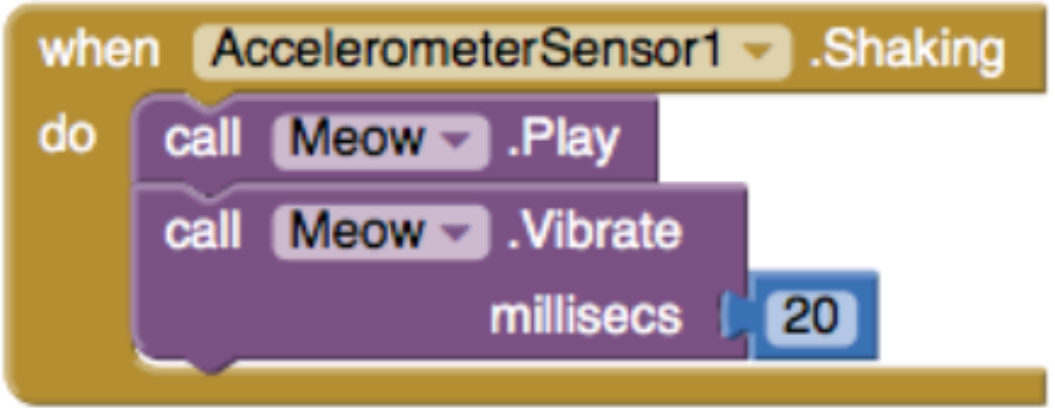
Why Learn App Inventor?

- Create your own mobile applications
- Use it for school projects
- Solve real world problems
- Create a prototype for your business

Problem solving exercises

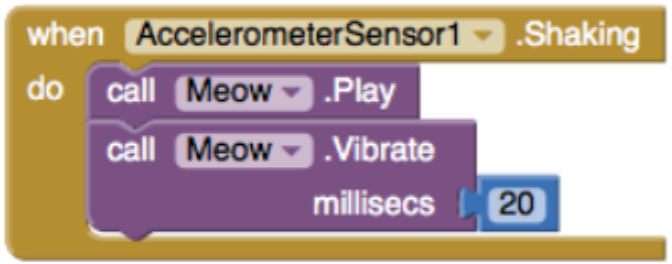
What Does This App do?





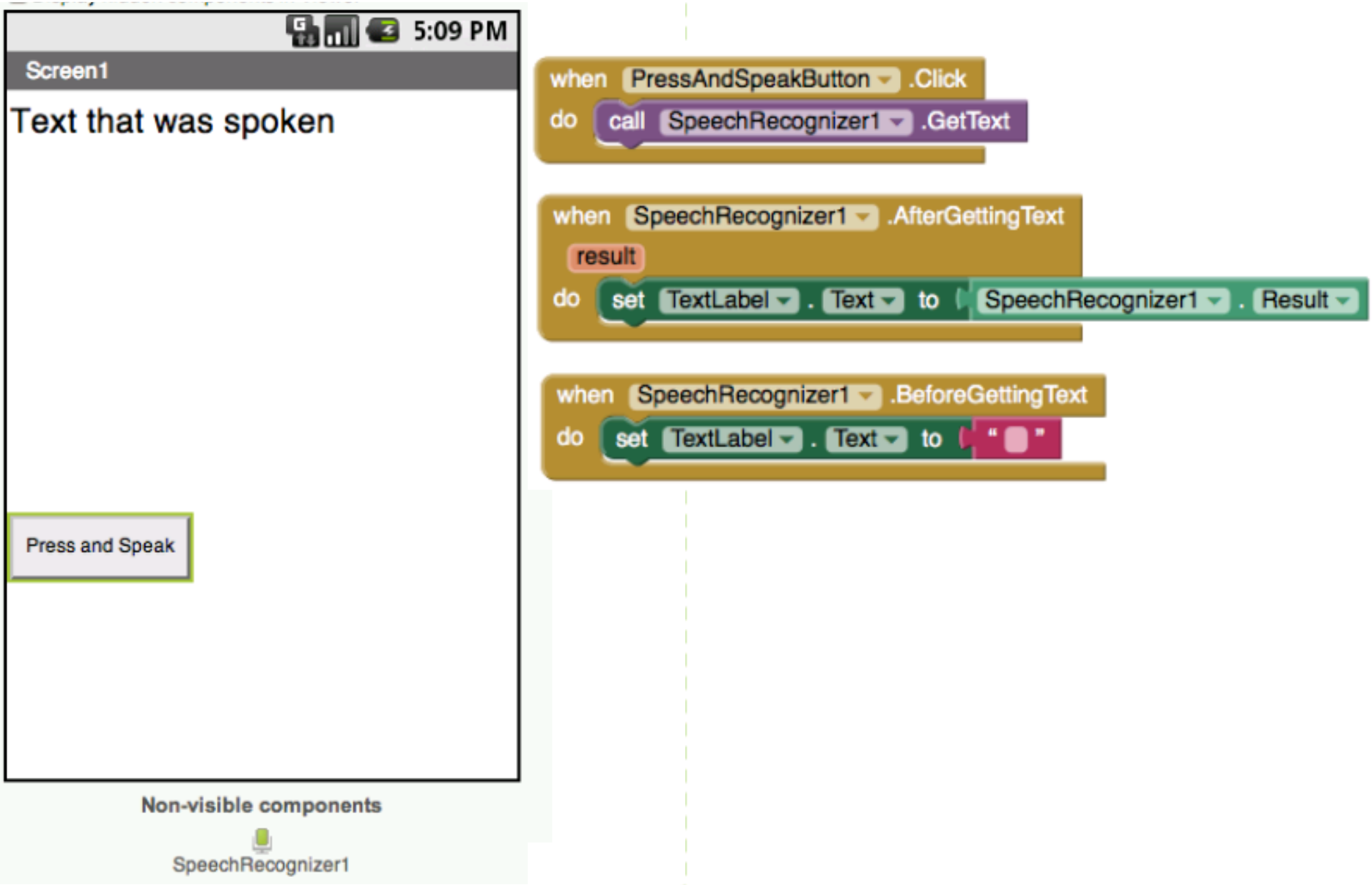
Shaking Phone

Make something happen when you shake your phone.



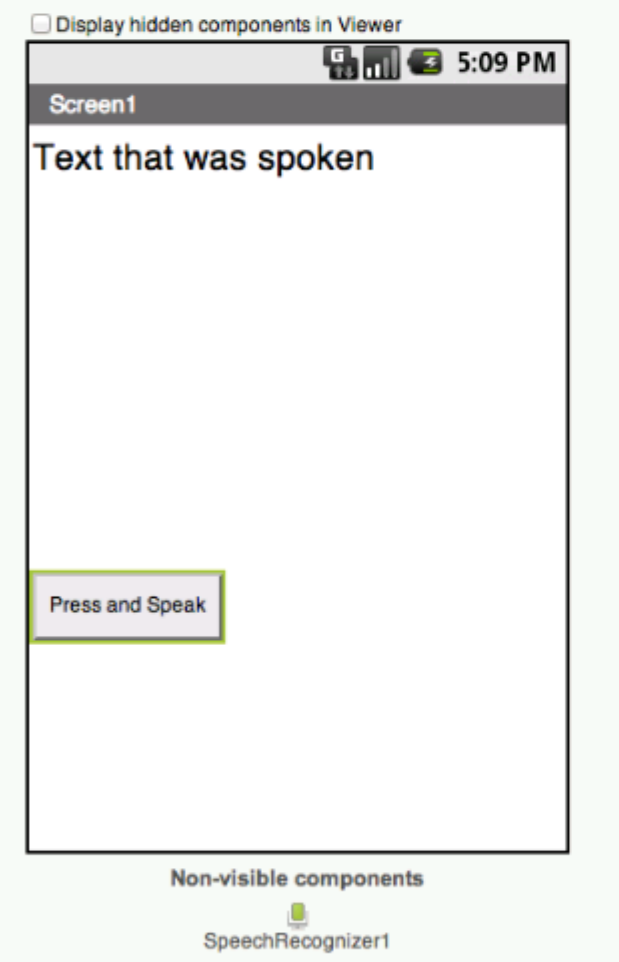
The **AccelerometerSensor.Shaking** event will detect when the phone is shaking and then the Meow sound will play and the phone will vibrate for 20 milliseconds.

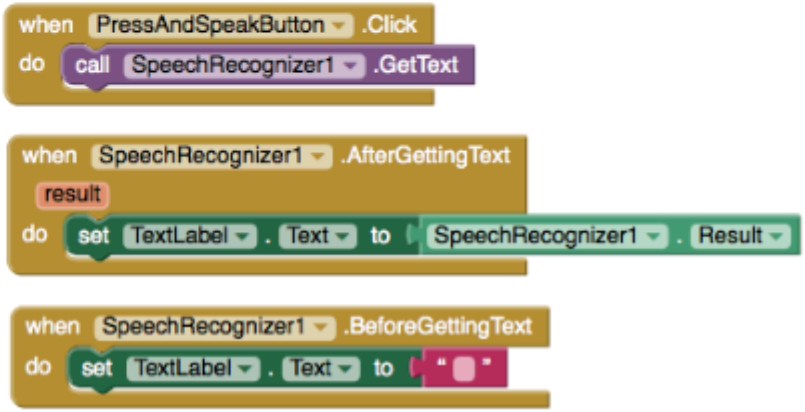
What Does This App do?



Speech Recognition

Display the text of what is being said on the phone screen.



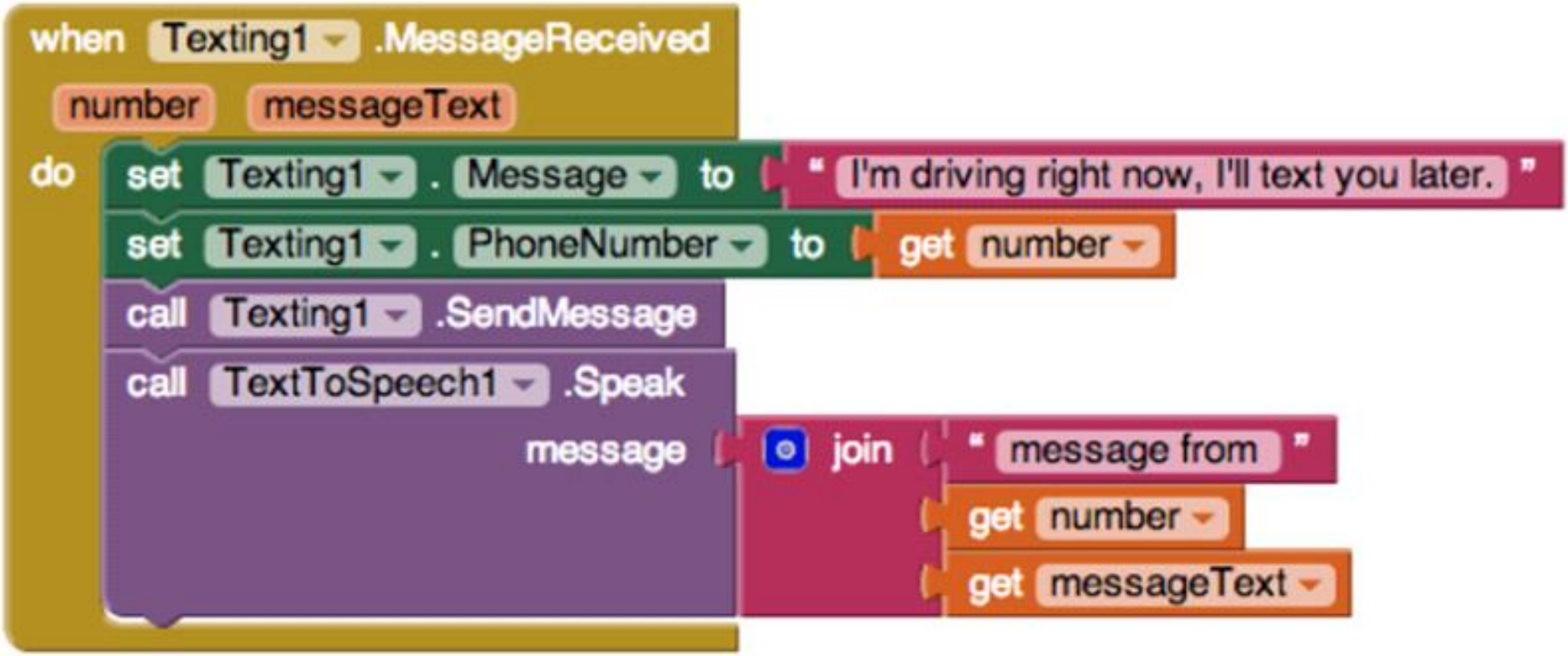


When the **PressAndSpeakButton** is clicked the **SpeechRecognizer** event is called and is ready for you to speak.

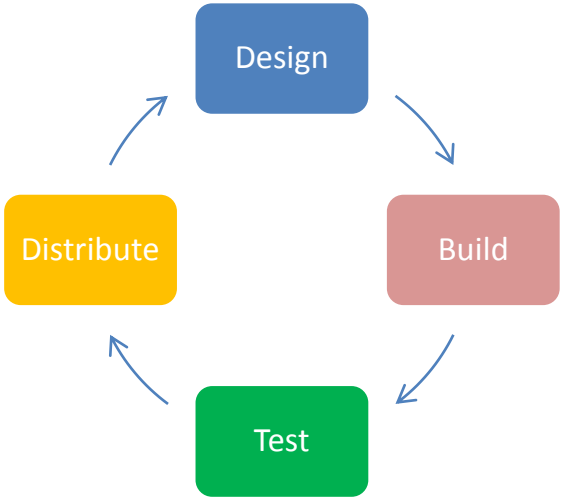
The **BeforeGettingText** event will be triggered before speech has been received and recognized. Then the **Label** will display no text on the screen.

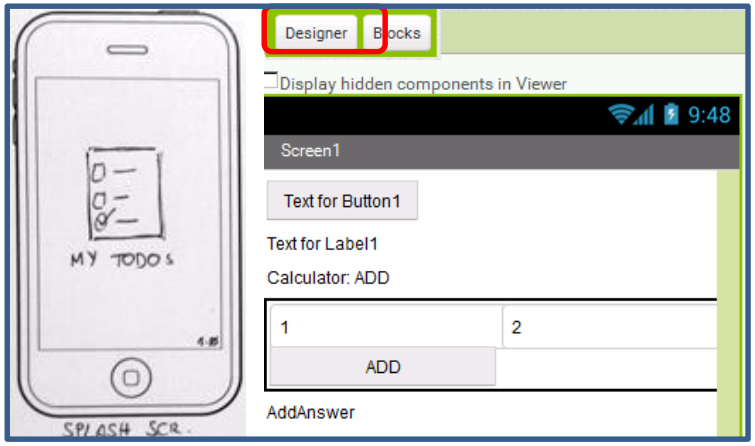
The **AfterGettingText** event will be triggered once speech has been received and recognized. Then the **Label** will display the text on the screen.

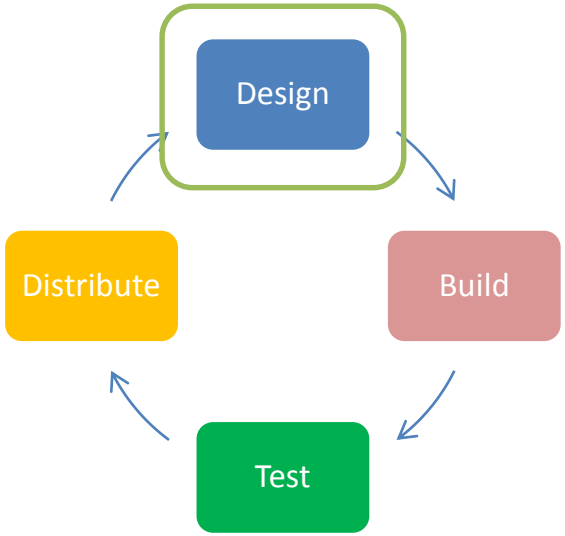
What Does this App do?

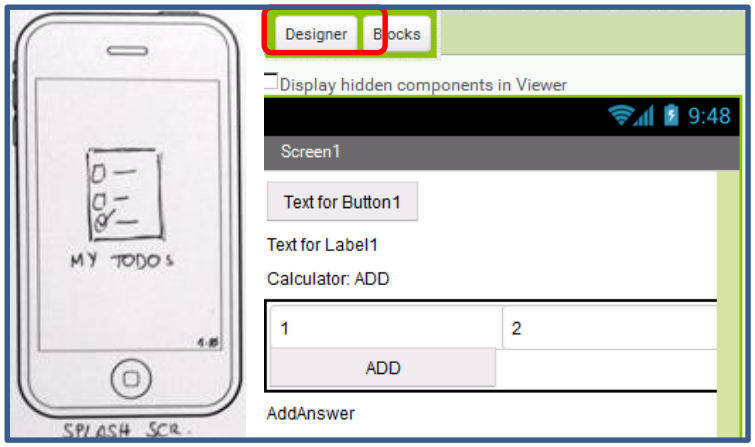


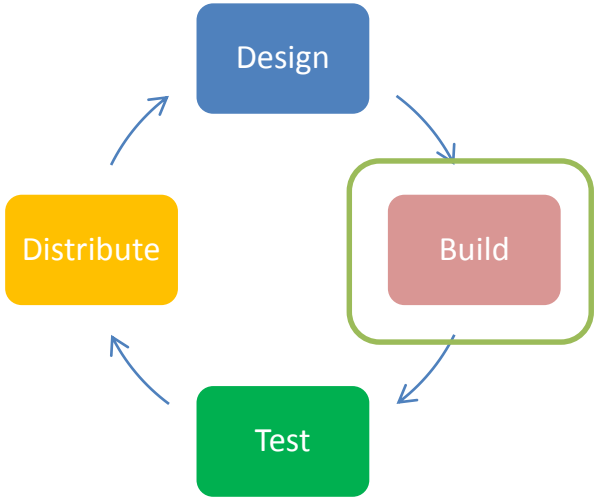
Process for creating apps

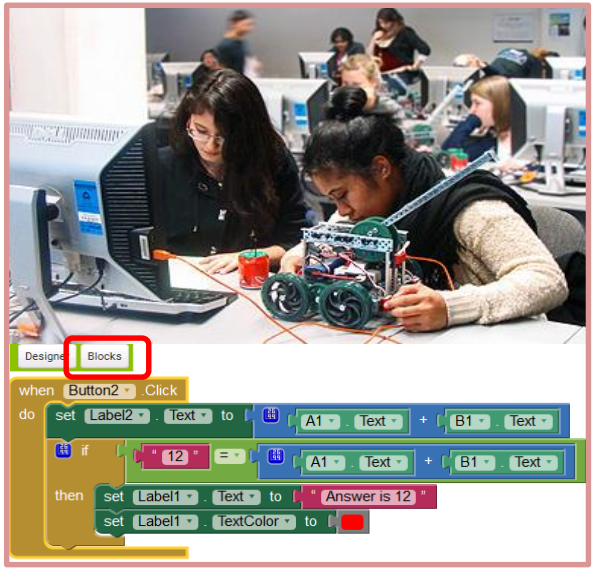

Design
Build
Test
Distribute

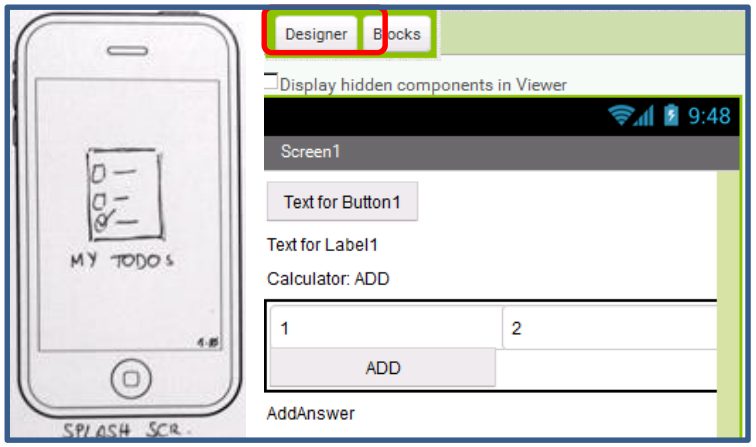


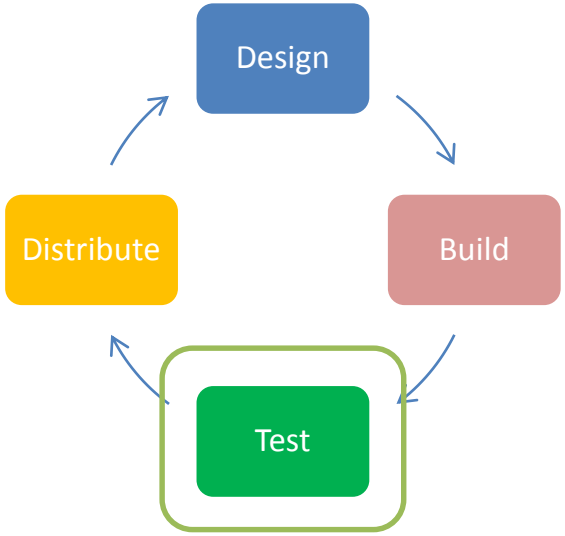


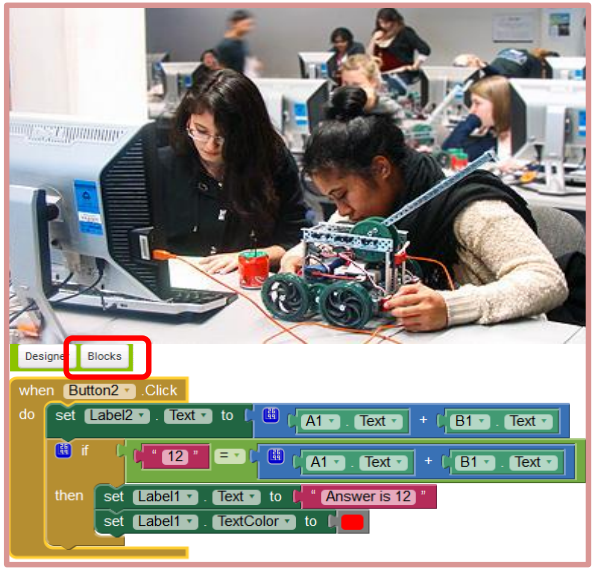


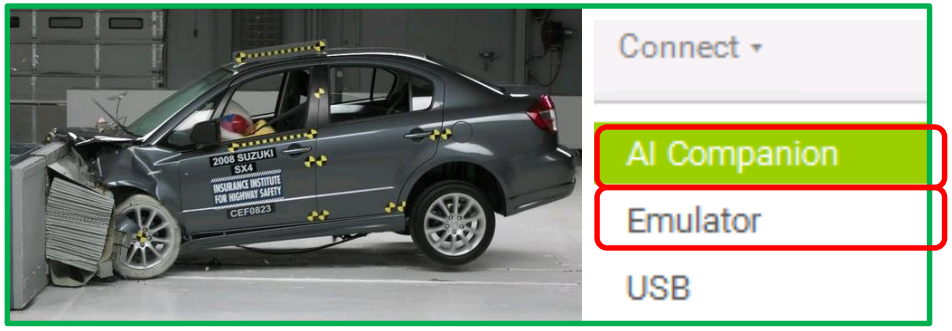




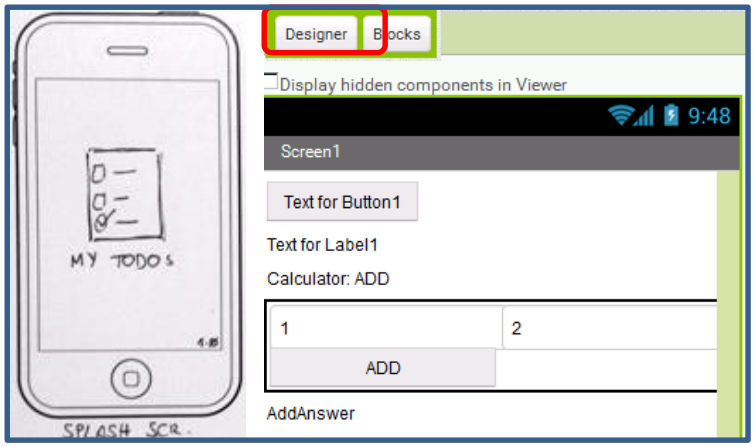


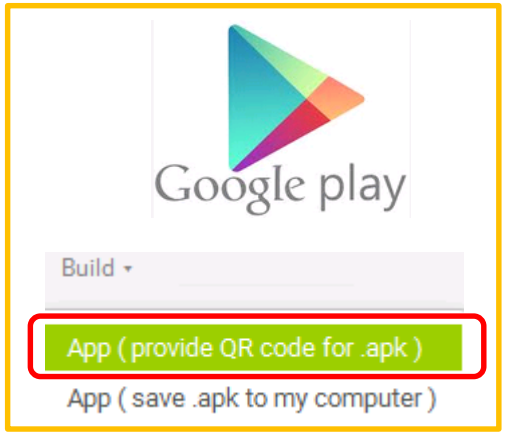


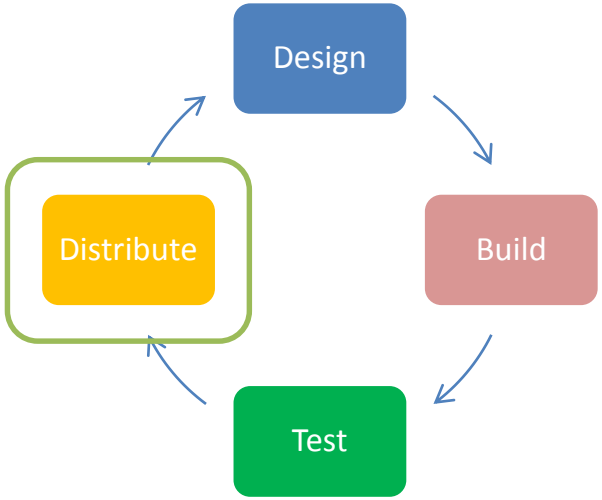


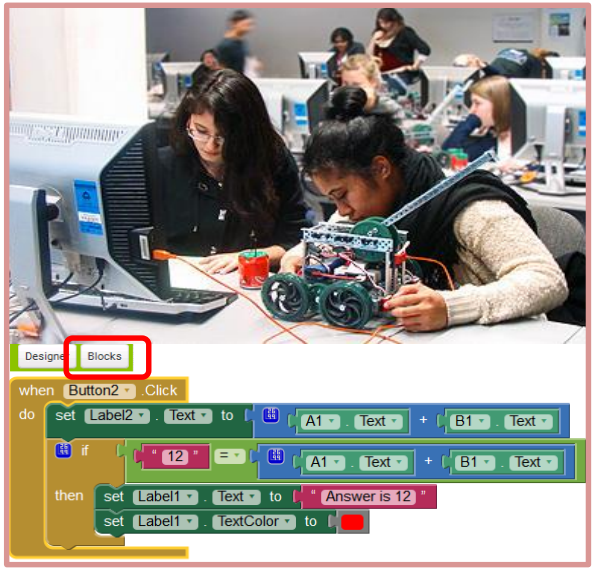


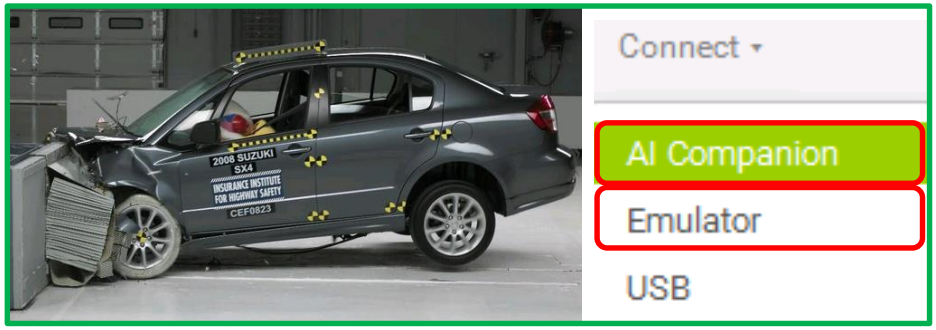
http://media.mediatemple.netdna-cdn.com/wp-content/uploads/2013/05/uiflow_large_mini.jpg
http://www.engineering.auckland.ac.nz/en/for/currentstudents/women-in-engineering/wie-engineer-her-future.html
http://indianautosblog.com/wp-content/uploads/2008/12/suzuki-sx4-crash-test-iihs.jpg











Setting Up Your Phones

Getting Started

What you need:

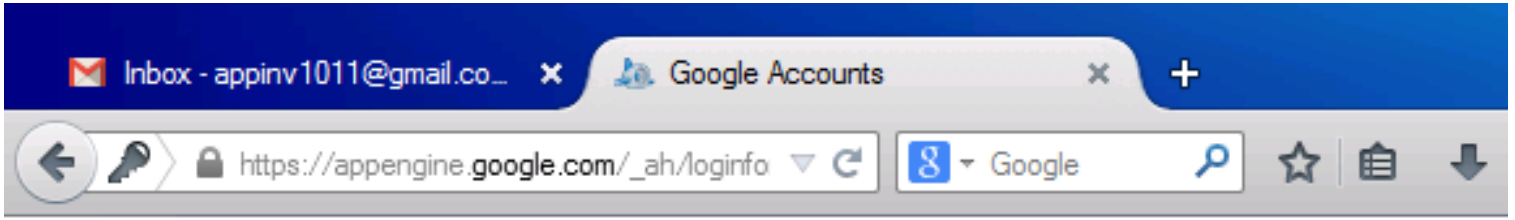
- Internet access
- Laptop (Windows or Mac)
- Gmail account
- Computer with Firefox 3.6/Chrome 4.0/ Safari 5.0 web browser
- Android phone or tablet with OS 2.3 or higher

Log in to Gmail

- Go to Google and click on Sign In
- If you have an account sign in now
- Otherwise, click on New Account and create one

Go to App Inventor

ai2.appinventor.mit.edu



Google accounts

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

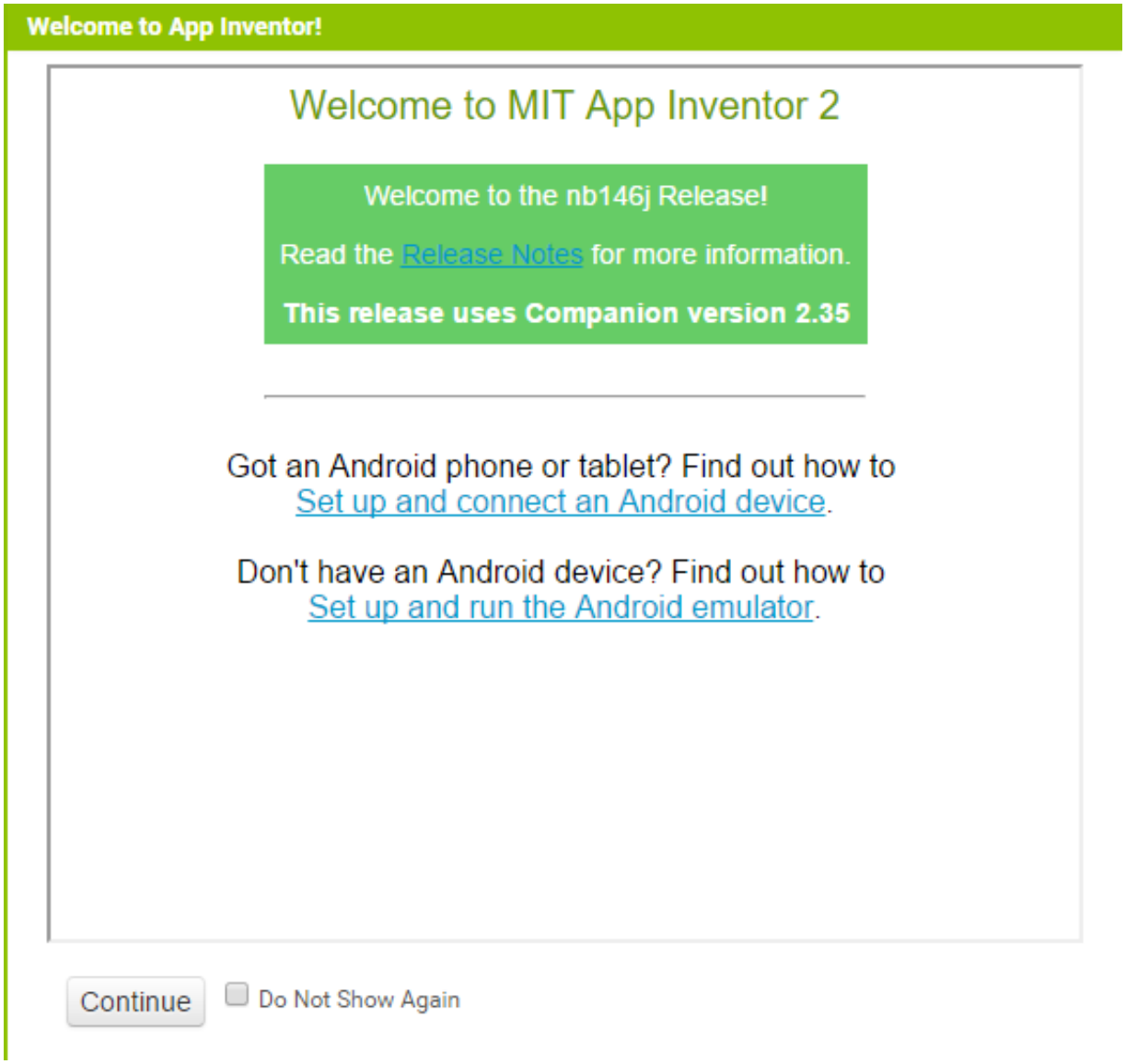
Please select an account that you would like to use.

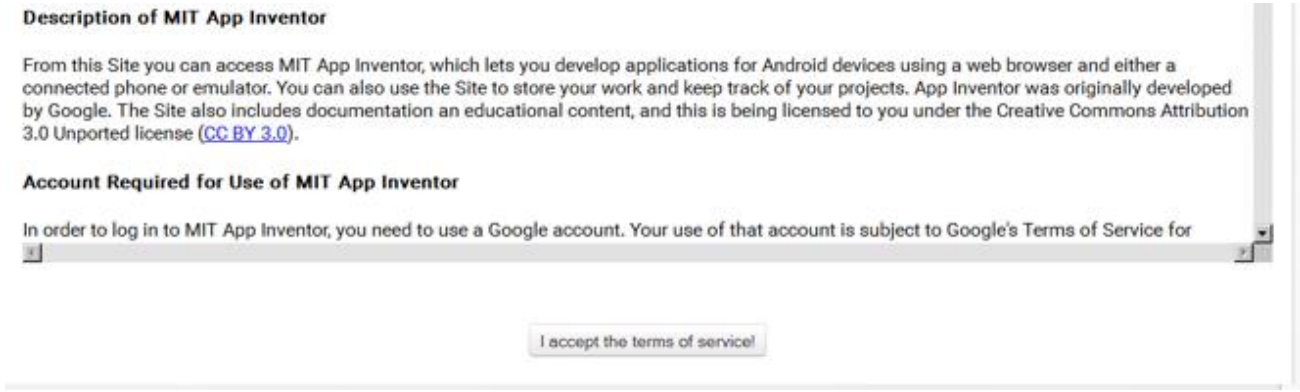
- appinv1011@gmail.com

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 AppInventor Version 2** but not your password or any other personal information.

Allow | No thanks | Sign in to another account

☑ Remember this approval for the next 30 days


Welcome to App Inventor!
Welcome to MIT App Inventor 2
Welcome to the nb146j Release!
Read the Release Notes for more information.
This release uses Companion version 2.35
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.
Continue
Do Not Show Again

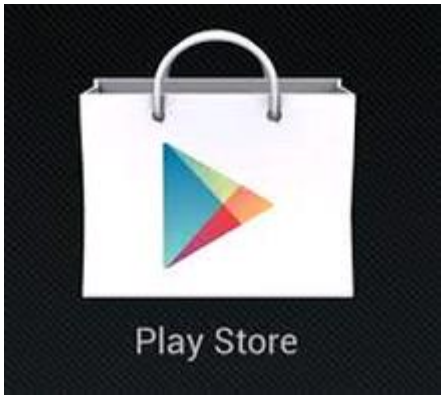

Description of MIT App Inventor
From this Site you can access MIT App Inventor, which lets you develop applications for Android devices using a web browser and either a connected phone or emulator. You can also use the Site to store your work and keep track of your projects. App Inventor was originally developed by Google. The Site also includes documentation an educational content, and this is being licensed to you under the Creative Commons Attribution 3.0 Unported license (CC BY 3.0).
Account Required for Use of MIT App Inventor
In order to log in to MIT App Inventor, you need to use a Google account. Your use of that account is subject to Google's Terms of Service for
I accept the terms of service!

AI Companion
on your phone or tablet

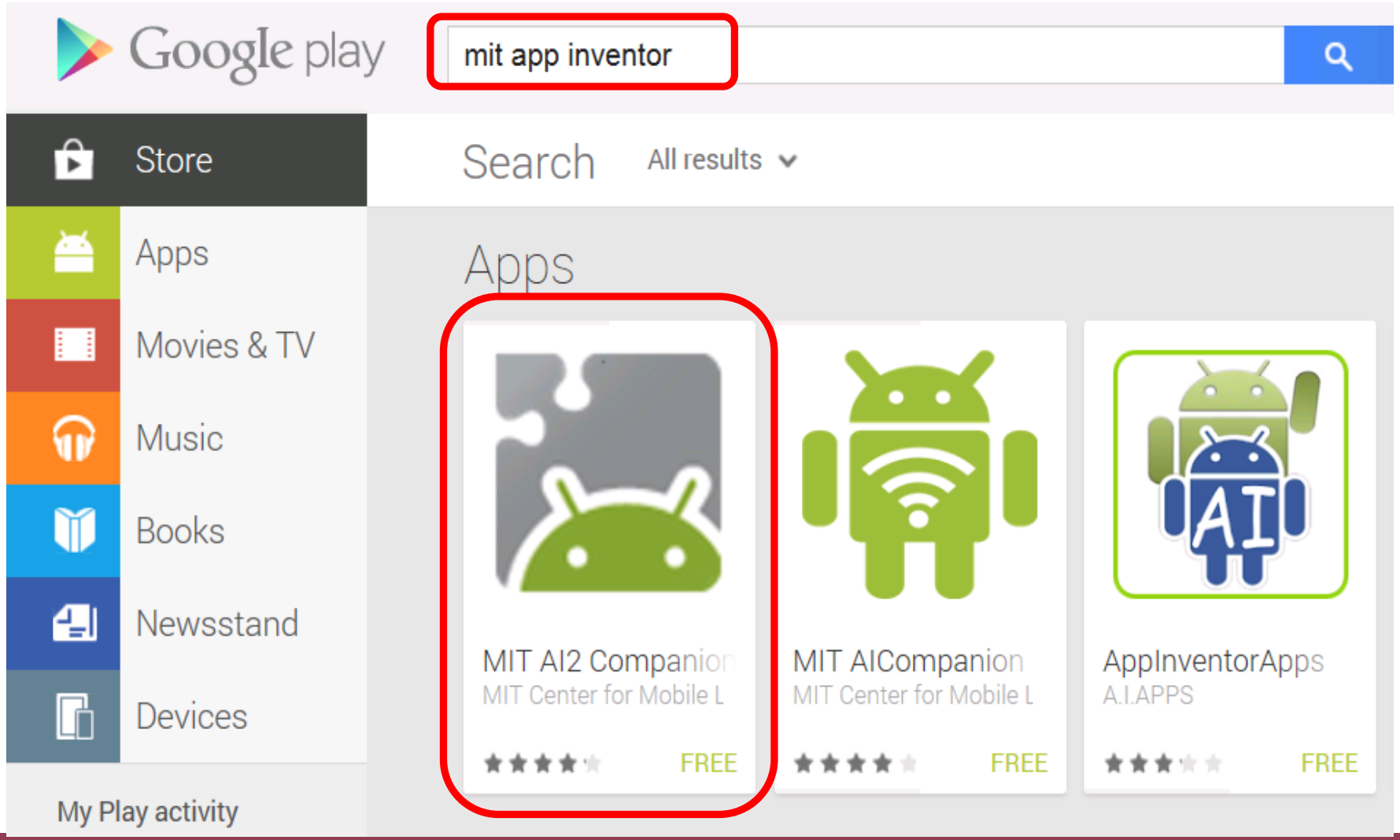
- Can see changes made in real-time
- Operate and test your app with your actual phone
- GPS, Camera, and accelerometer might work
- Application is only temporarily running on the phone. Application is not stored in the phone

Downloading AI Companion

- Go to the Play Store in your phone

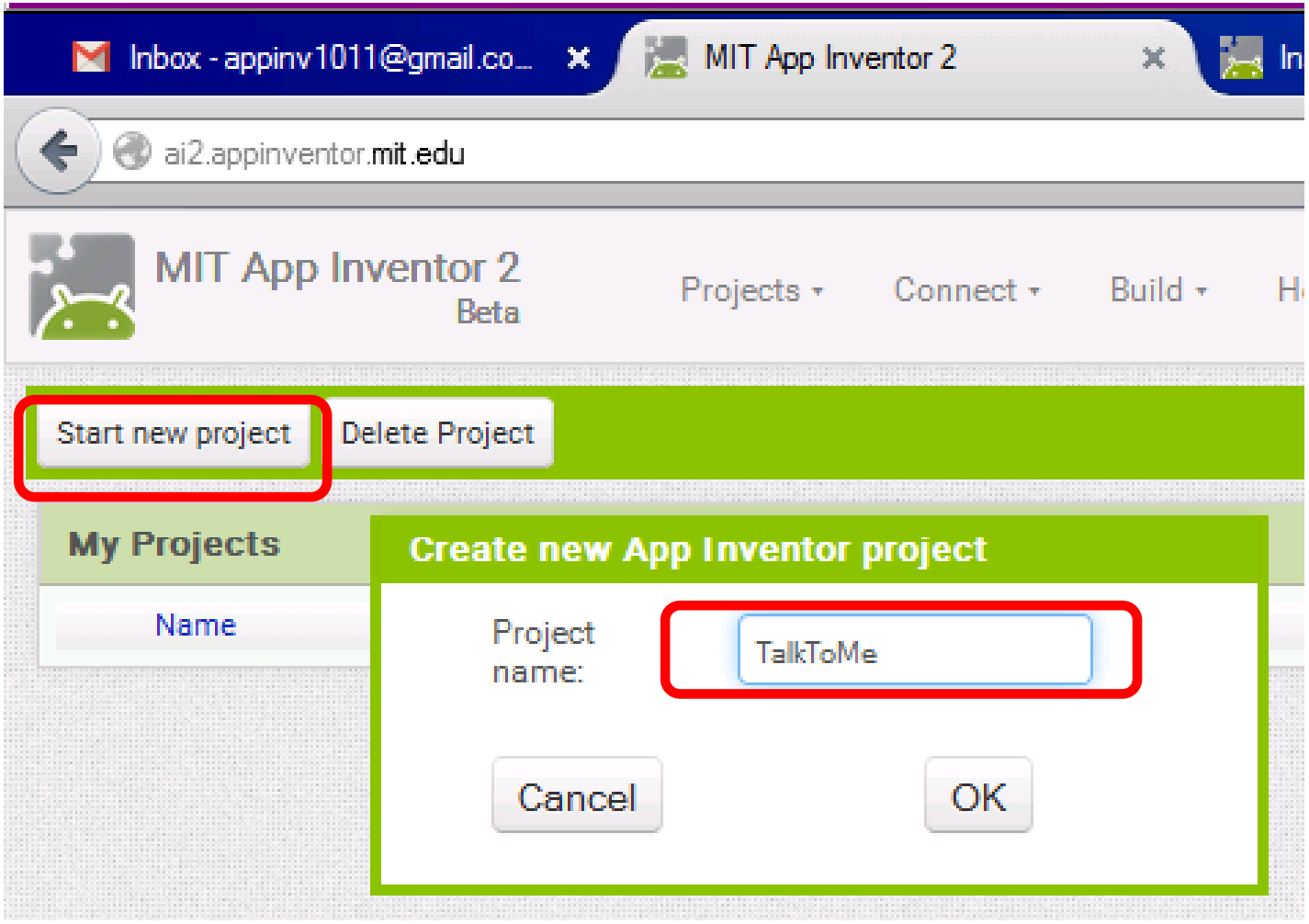


Search for “MIT app inventor”

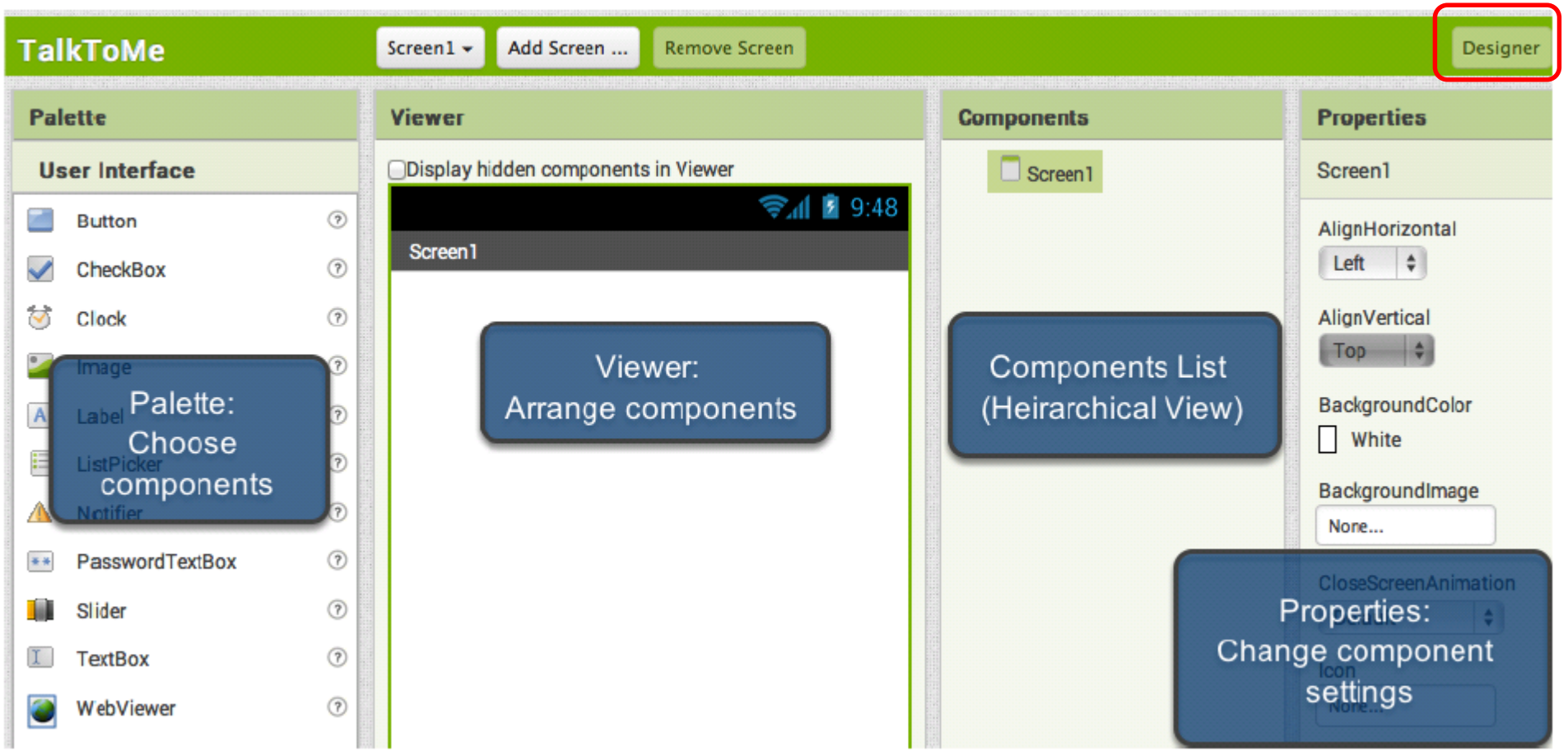


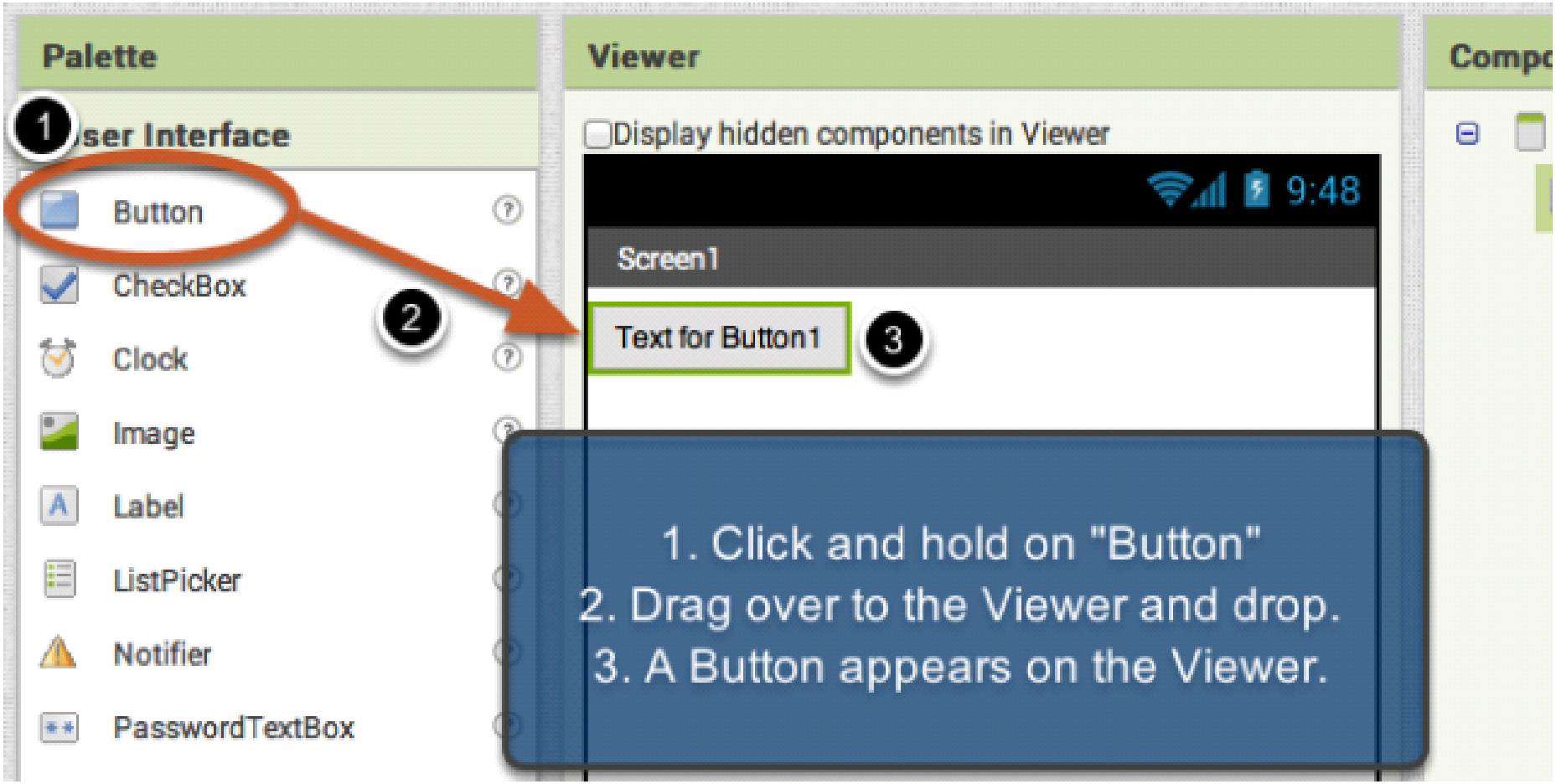
Creating your first app “Talk to Me”

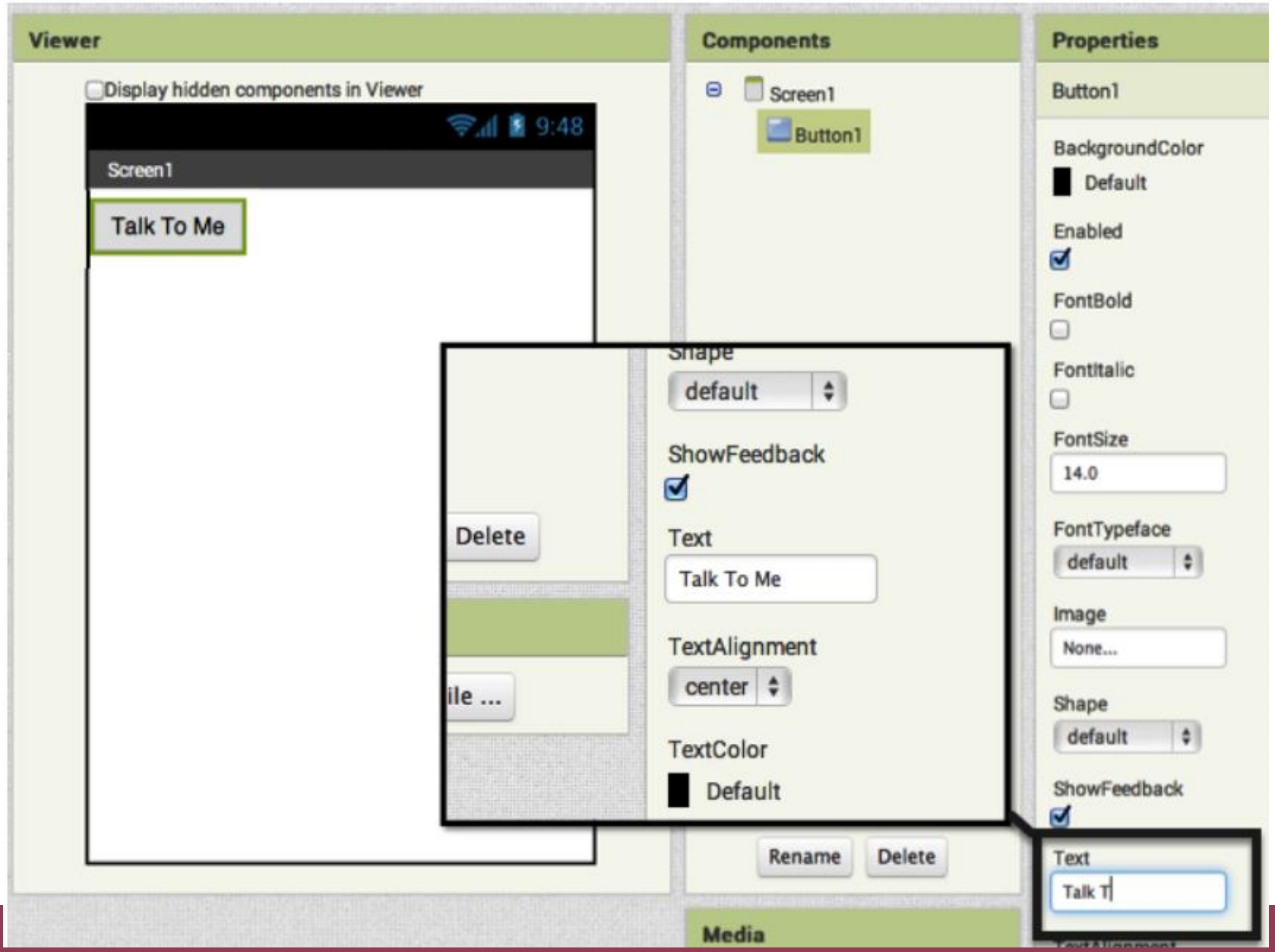
Let’s get started

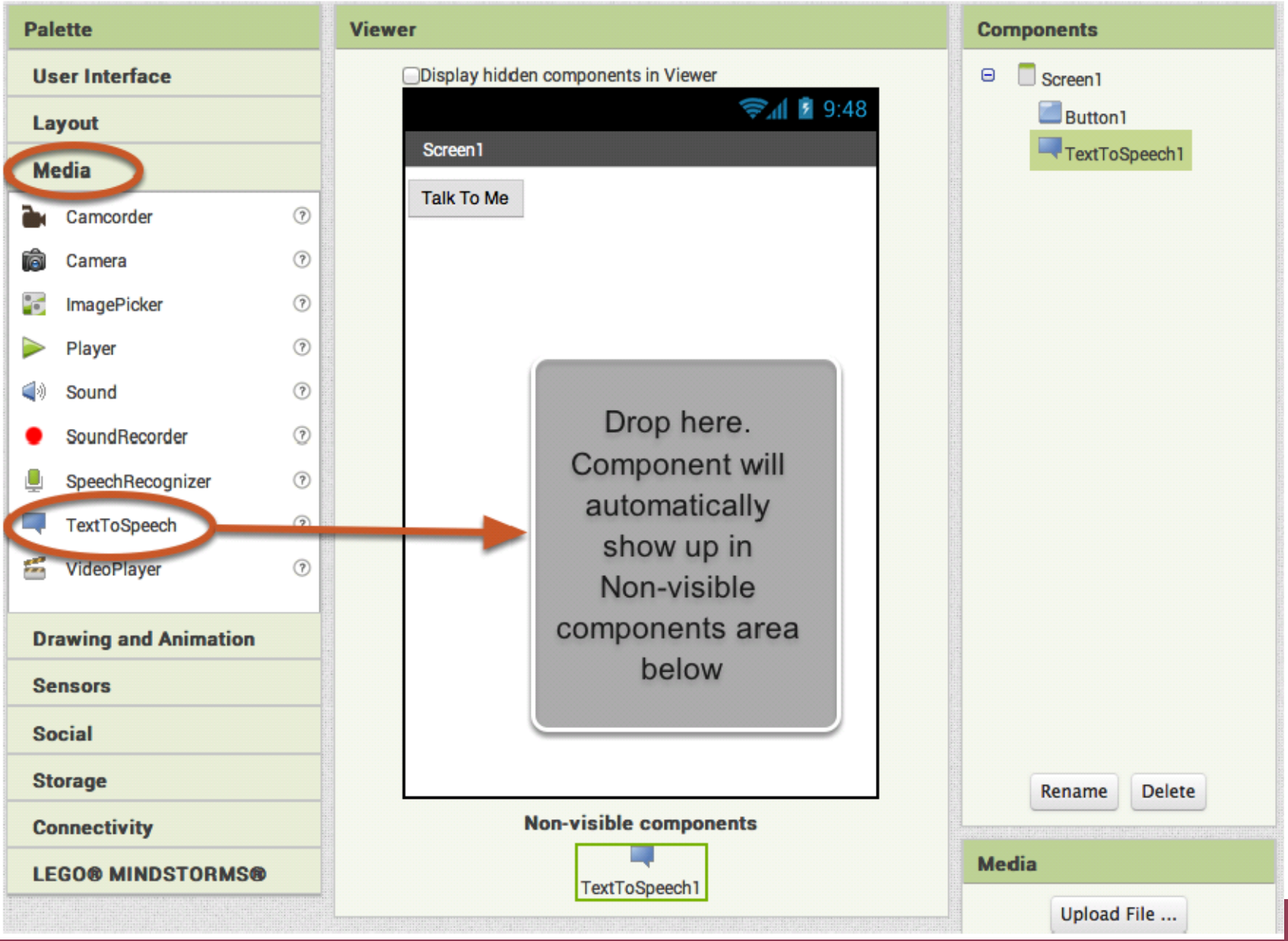


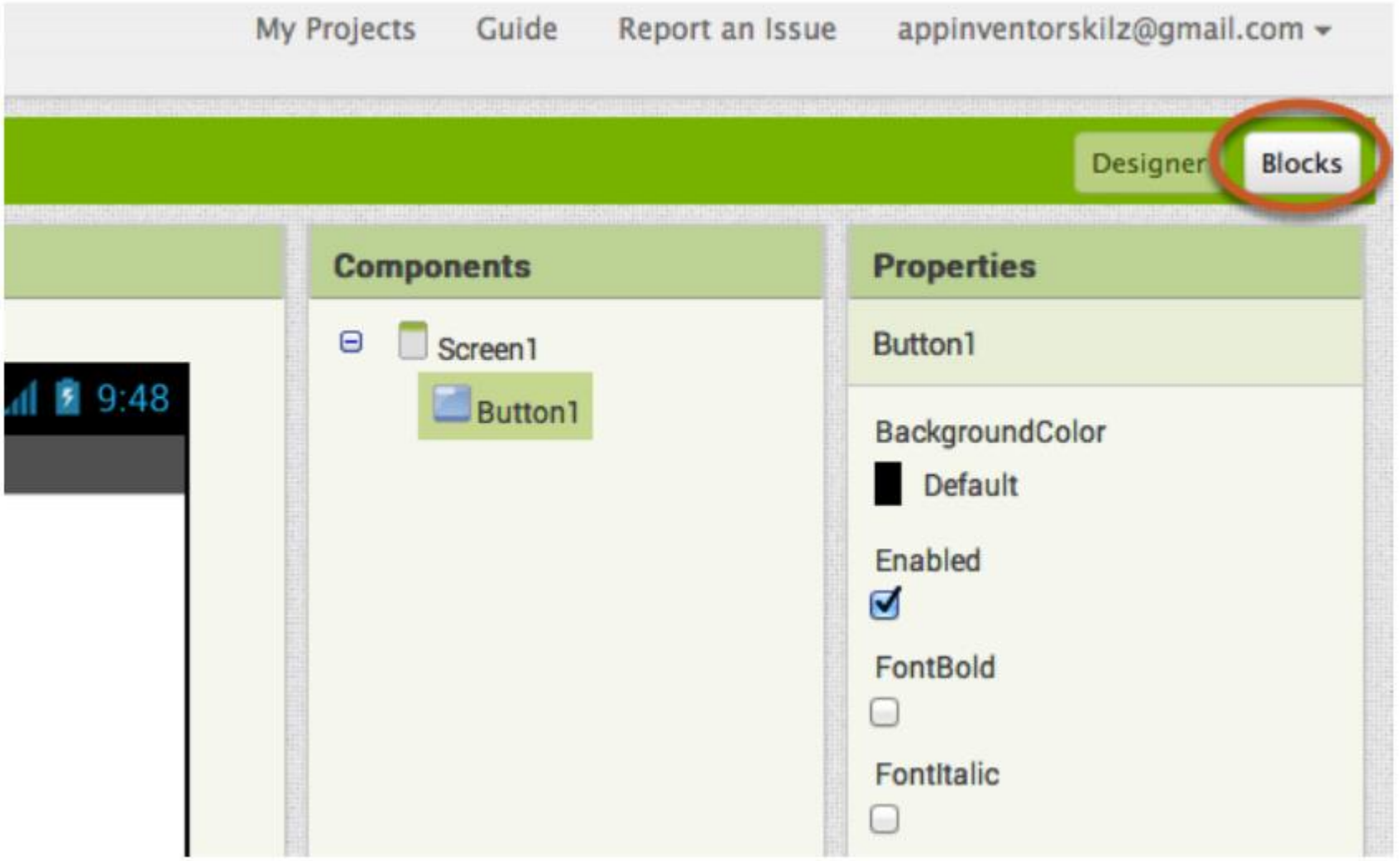
Getting Started

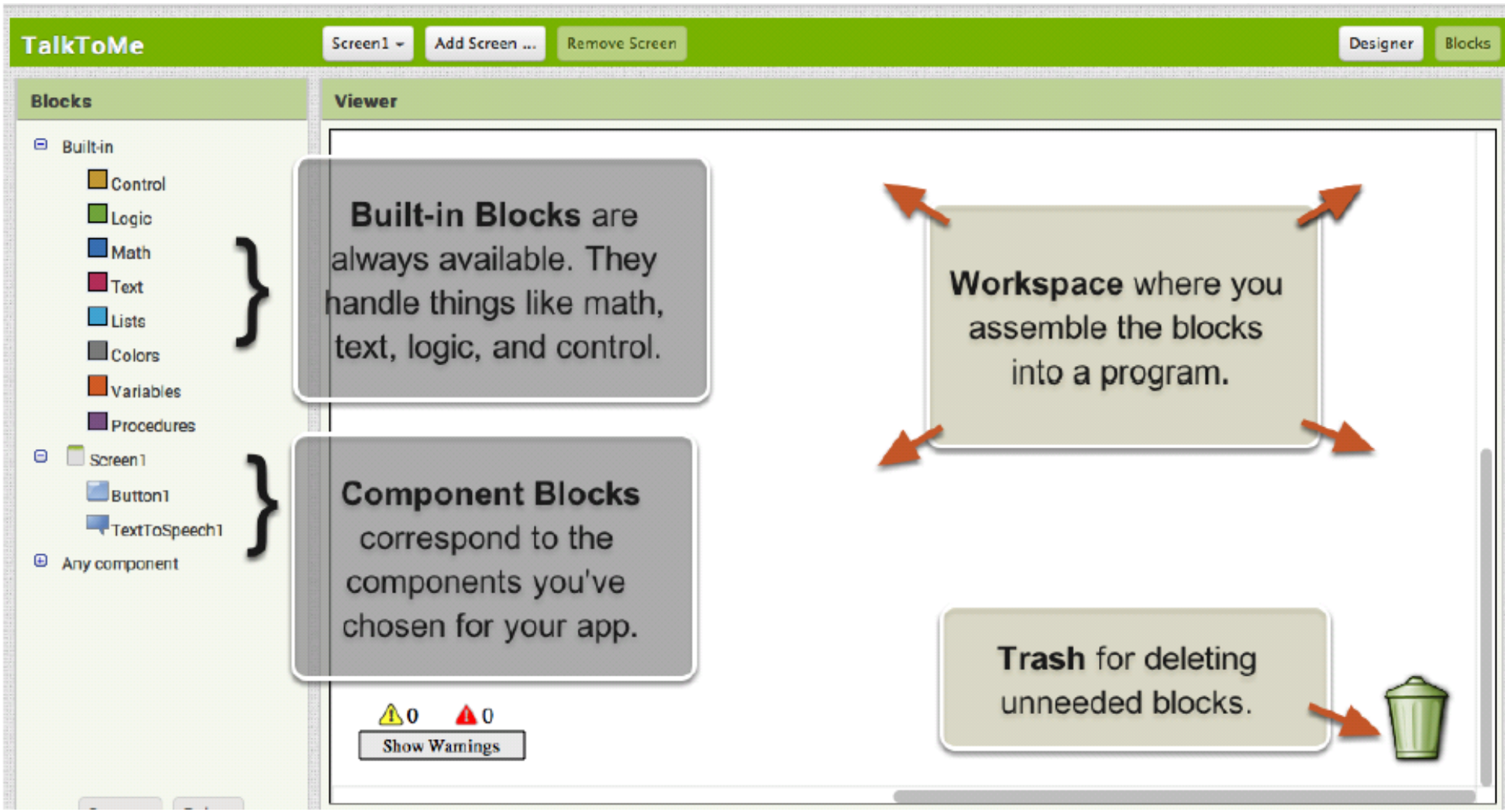


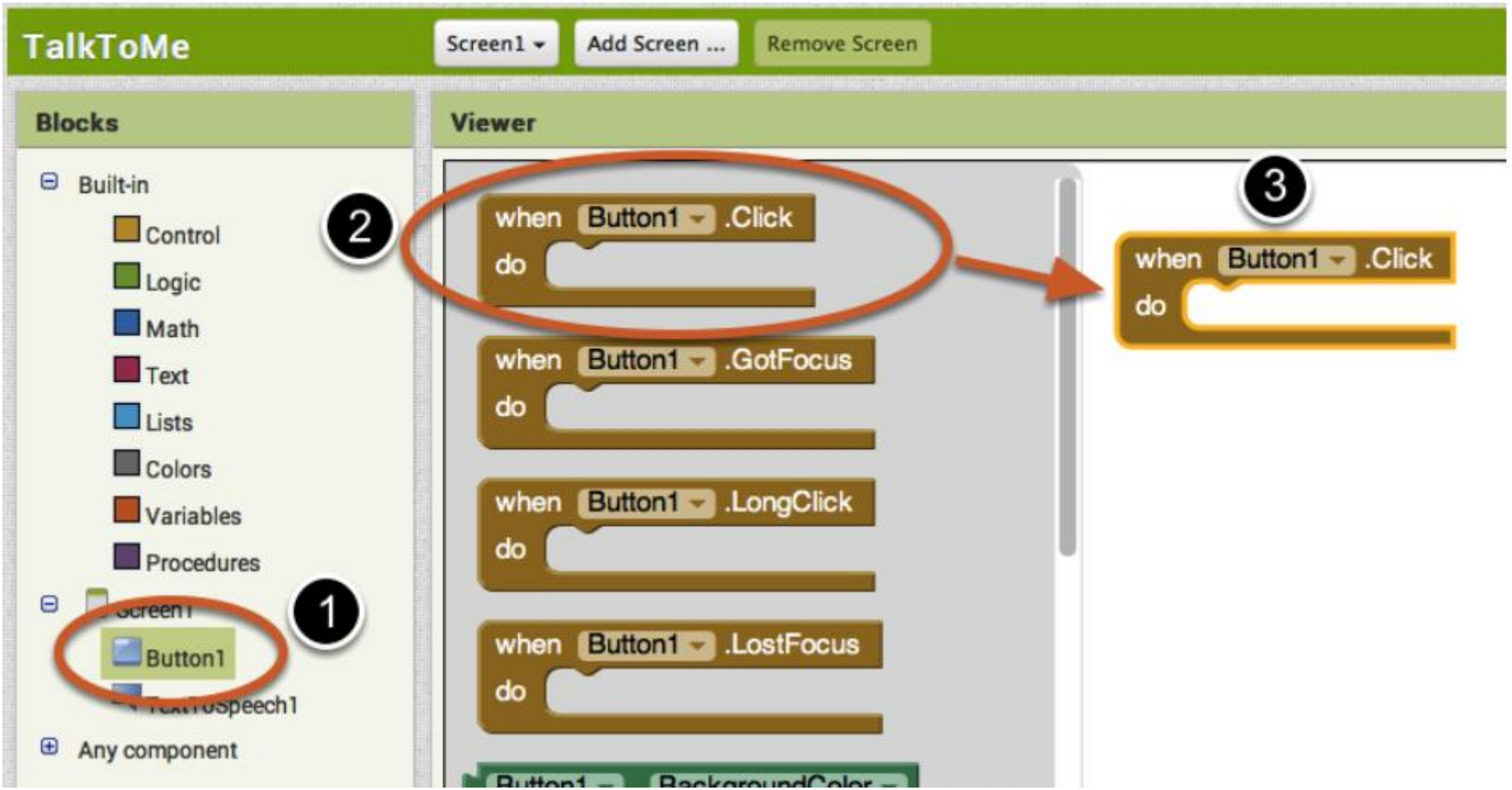

Palette
Viewer
1
ser Interface
Button
CheckBox
Clock
Image
Label
ListPicker
Notifier
PasswordTextBox
Display hidden components in Viewer
9:48
Screen1
Text for Button1
2
3
1. Click and hold on "Button"
2. Drag over to the Viewer and drop.
3. A Button appears on the Viewer.

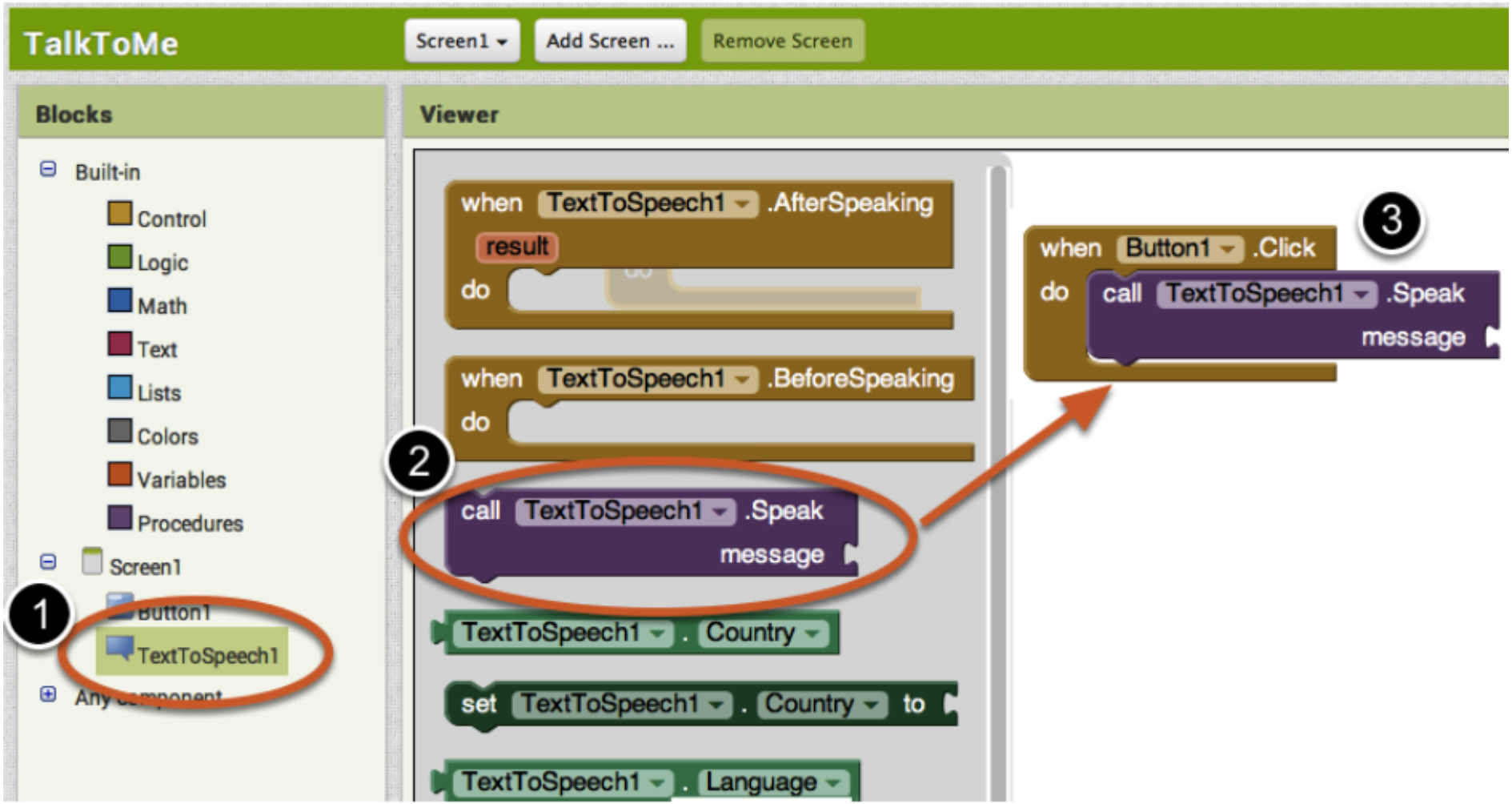

Viewer
Display hidden components in Viewer
9:48
Screen1
Talk To Me
Components
Screen1
Button1
Properties
Button1
BackgroundColor
Default
Enabled
FontBold
FontItalic
FontSize
14.0
FontTypeface
default
Image
None...
Shape
default
ShowFeedback
Text
Talk T
Delete
ile ...
Shape
default
ShowFeedback
Text
Talk To Me
TextAlignment
center
TextColor
Default
Rename
Delete
Media

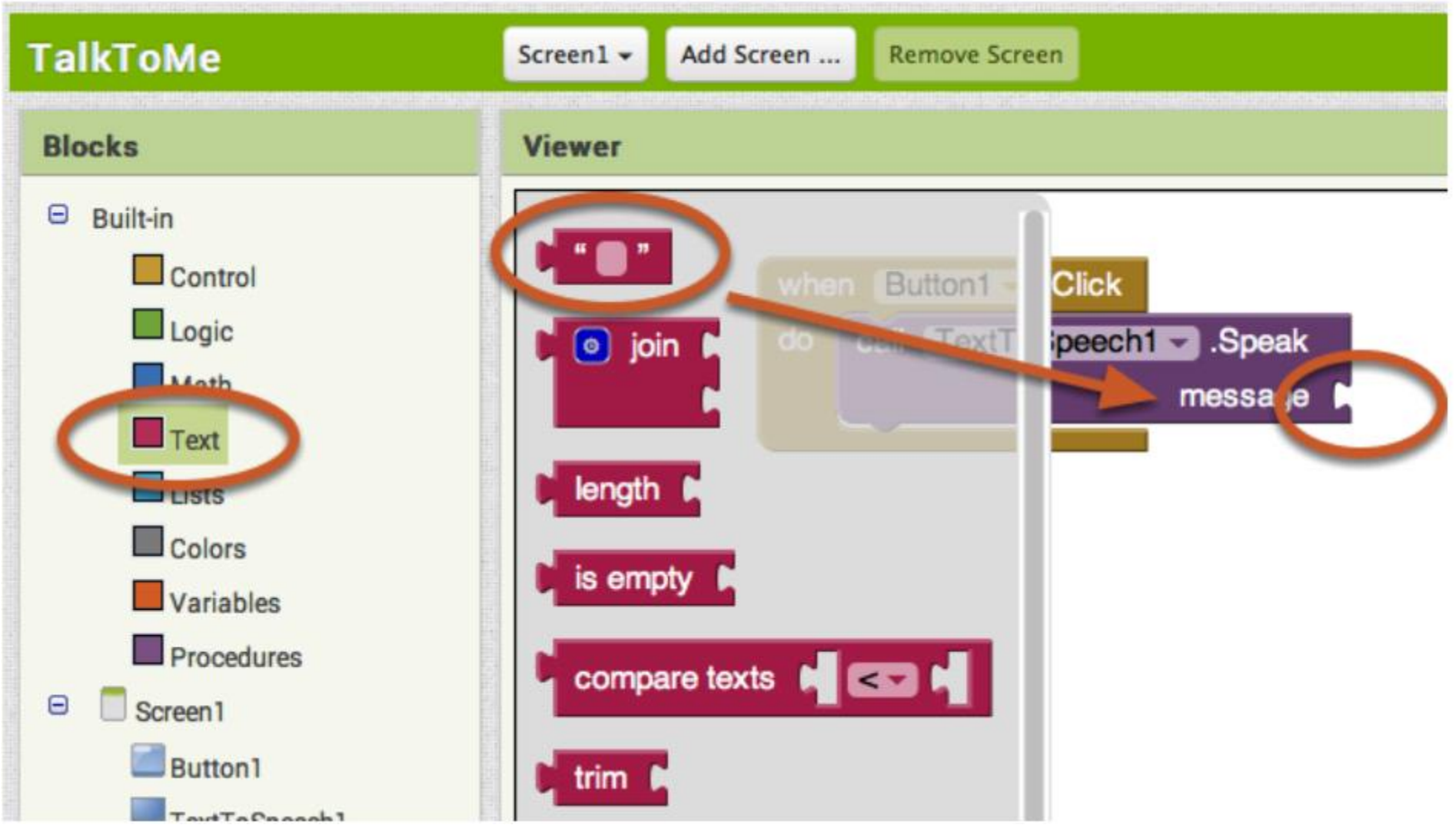

Palette
User Interface
Layout
Media
Camcorder
Camera
ImagePicker
Player
Sound
SoundRecorder
SpeechRecognizer
TextToSpeech
VideoPlayer
Drawing and Animation
Sensors
Social
Storage
Connectivity
LEGO® MINDSTORMS®
Viewer
Display hidden components in Viewer
9:48
Screen1
Talk To Me
Drop here. Component will automatically show up in Non-visible components area below
Non-visible components
TextToSpeech1
Components
Screen1
Button1
TextToSpeech1
Rename
Delete
Media
Upload File ...

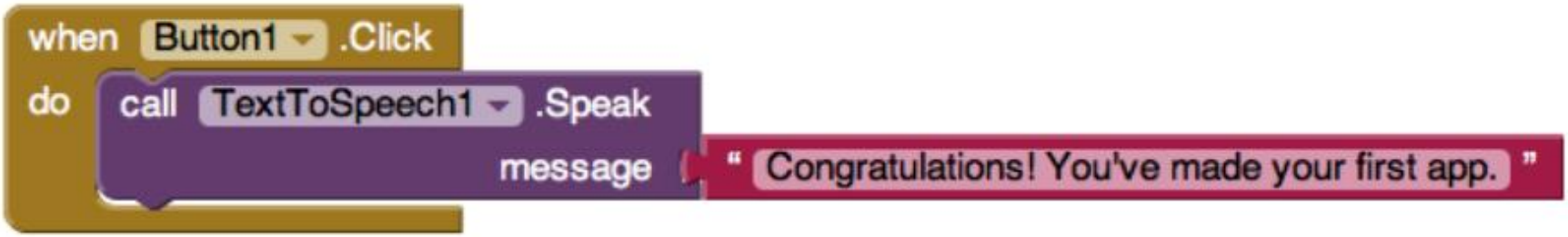

My Projects
Guide
Report an Issue
appinventorskilz@gmail.com
Designer
Blocks
Components
Screen1
Button1
Properties
Button1
BackgroundColor
Default
Enabled
FontBold
FontItalic
9:48

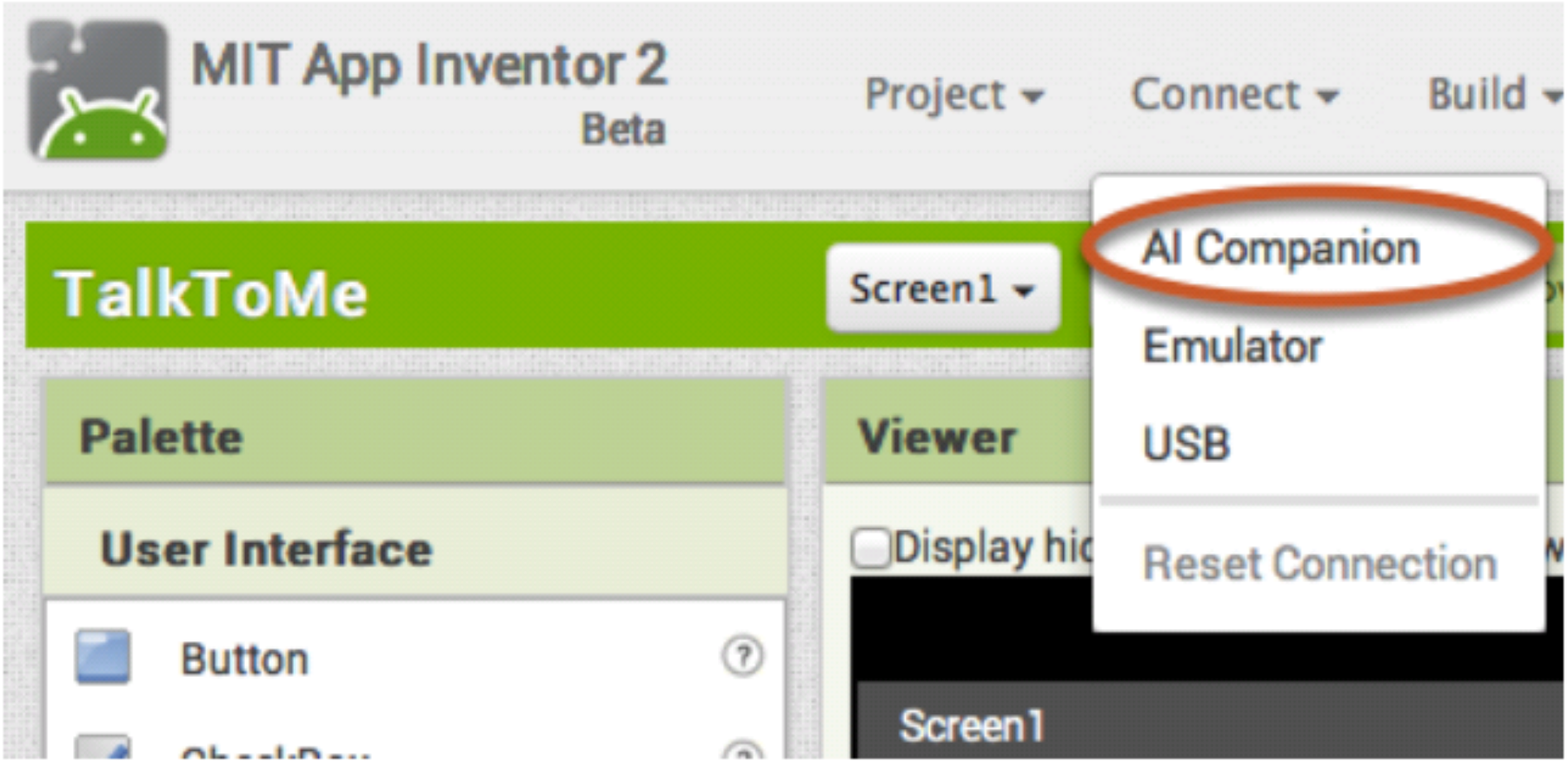

TalkToMe
Screen1
Add Screen ...
Remove Screen
Designer
Blocks
Blocks
Viewer
Built-in
Control
Logic
Math
Text
Lists
Colors
Variables
Procedures
Screen1
Button1
TextToSpeech1
Any component
Built-in Blocks are always available. They handle things like math, text, logic, and control.
Component Blocks correspond to the components you've chosen for your app.
Workspace where you assemble the blocks into a program.
Trash for deleting unneeded blocks.
0
0
Show Warnings


TalkToMe
Screen1
Add Screen ...
Remove Screen
Blocks
Viewer
Built-in
Control
Logic
Math
Text
Lists
Colors
Variables
Procedures
Button1
Any component
when Button1 .Click
do
when Button1 .GotFocus
do
when Button1 .LongClick
do
when Button1 .LostFocus
do
1
2
3

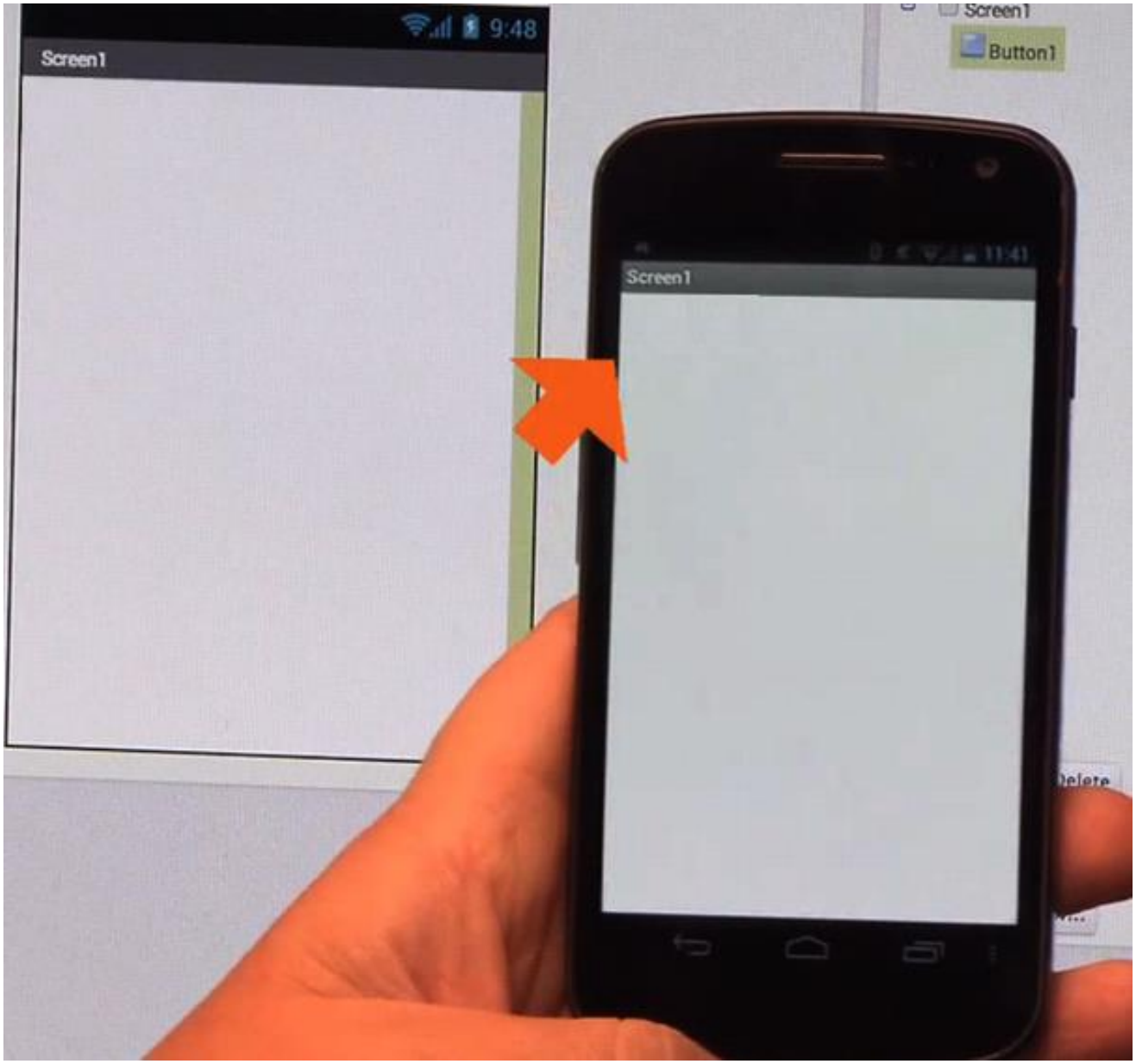

TalkToMe
Screen1
Add Screen ...
Remove Screen
Blocks
Built-in
Control
Logic
Math
Text
Lists
Colors
Variables
Procedures
Screen1
Button1
TextToSpeech1
Any component
Viewer
when TextToSpeech1 .AfterSpeaking
result
do
when TextToSpeech1 .BeforeSpeaking
do
call TextToSpeech1 .Speak
message
TextToSpeech1 . Country
set TextToSpeech1 . Country to
TextToSpeech1 . Language
when Button1 .Click
do
call TextToSpeech1 .Speak
message
1
2
3

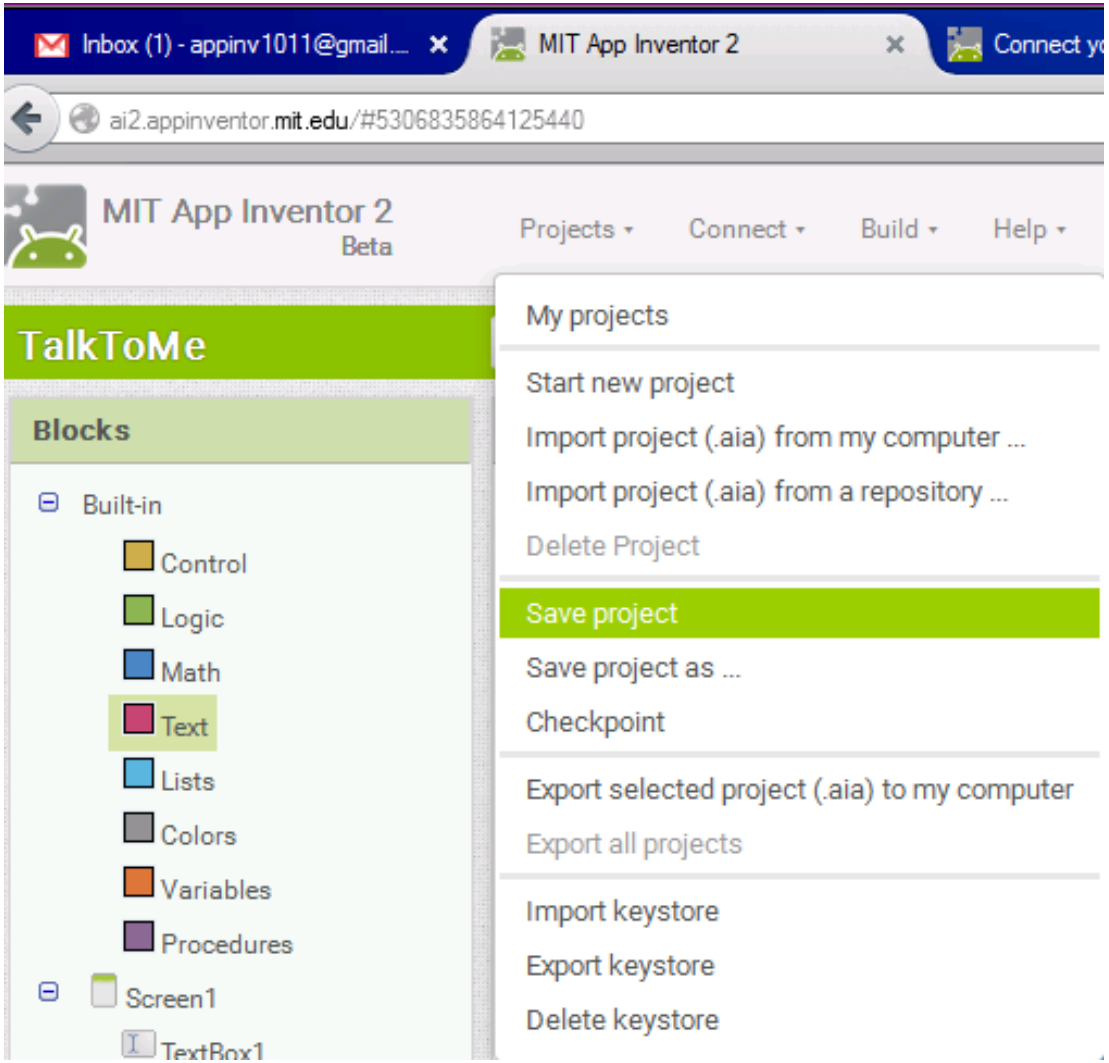

TalkToMe
Screen1
Add Screen ...
Remove Screen
Blocks
Viewer
Built-in
Control
Logic
Text
Colors
Variables
Procedures
Screen1
Button1
join
length
is empty
compare texts
trim
when Button1 Click
do
Speech1 .Speak
message


when Button1 .Click
do call TextToSpeech1 .Speak
message " Congratulations! You've made your first app. "

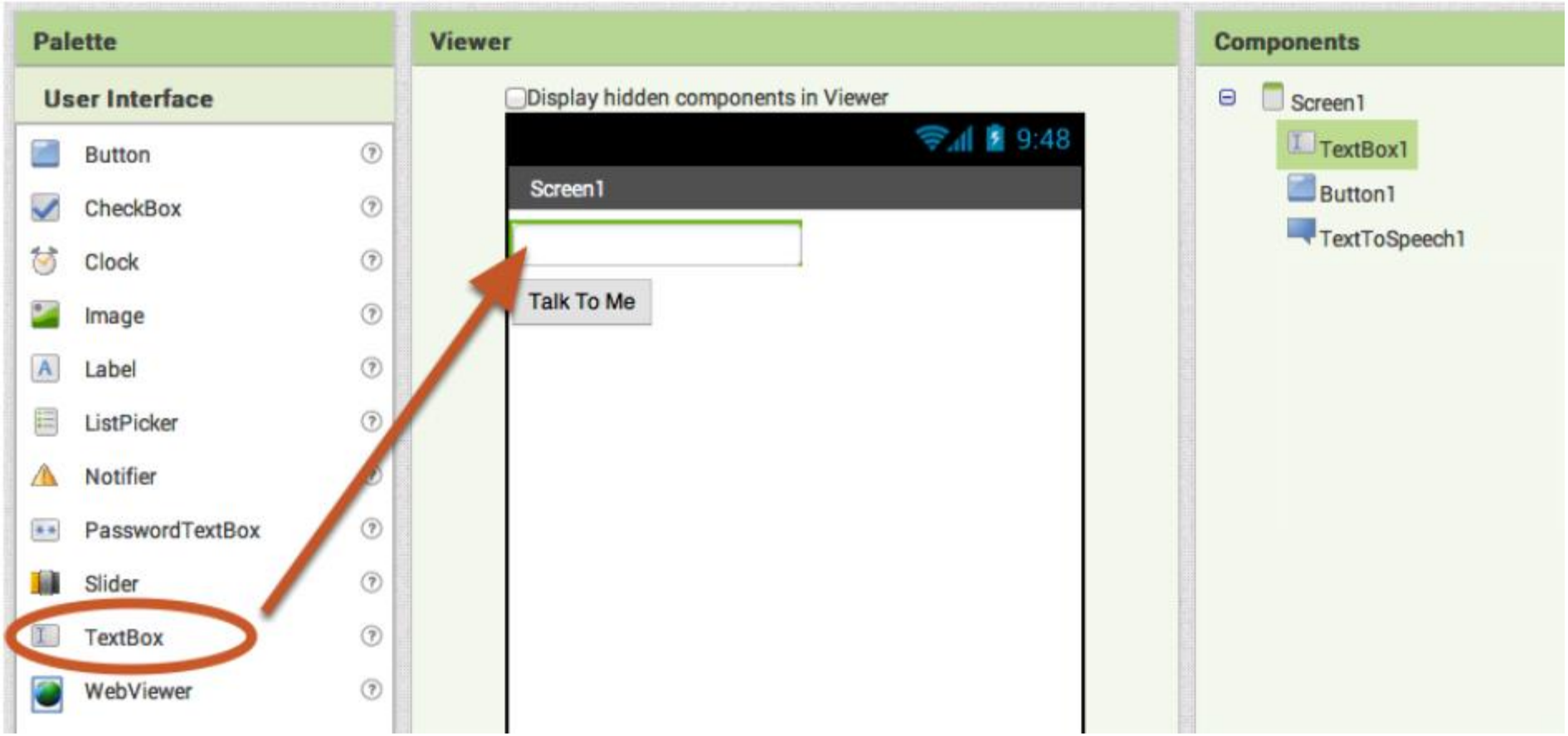

MIT App Inventor 2
Beta
Project
Connect
Build
AI Companion
Emulator
USB
Reset Connection
TalkToMe
Screen1
Palette
Viewer
User Interface
Button
Screen1

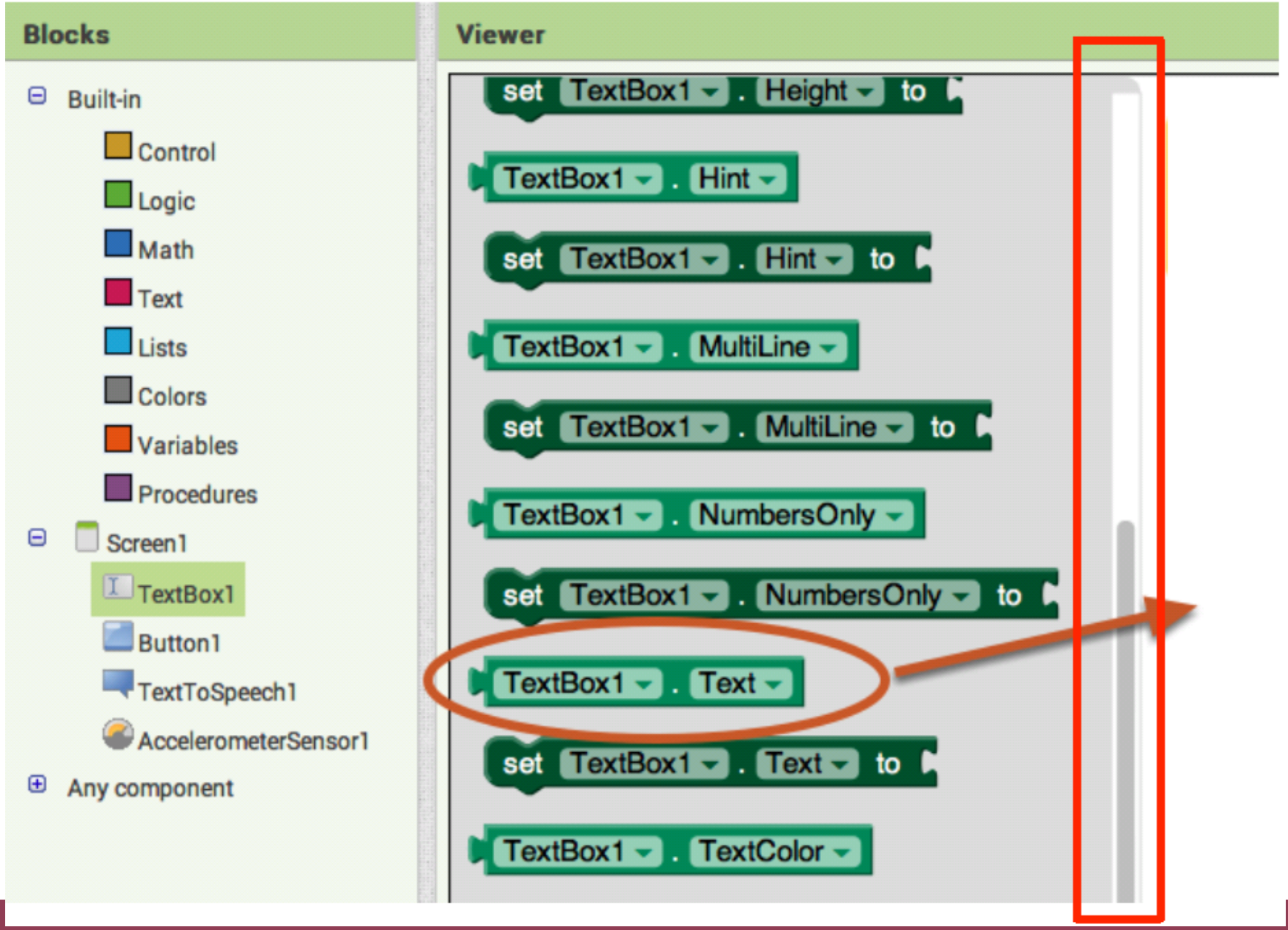

Screen1
Add Screen
Remove Screen
Viewer
Connect to Companion
Launch the MIT AI2 Companion on your device and then scan the barcode or type in the code to connect for live testing of your app.
Need help finding the Companion App?
Your code is:
dafjjs
Cancel
MIT App Inventor 2 Companion
MIT App Inventor 2
type in the 6-digit code
-or-
scan the QR code
Six Digit Code
connect with code
scan QR code

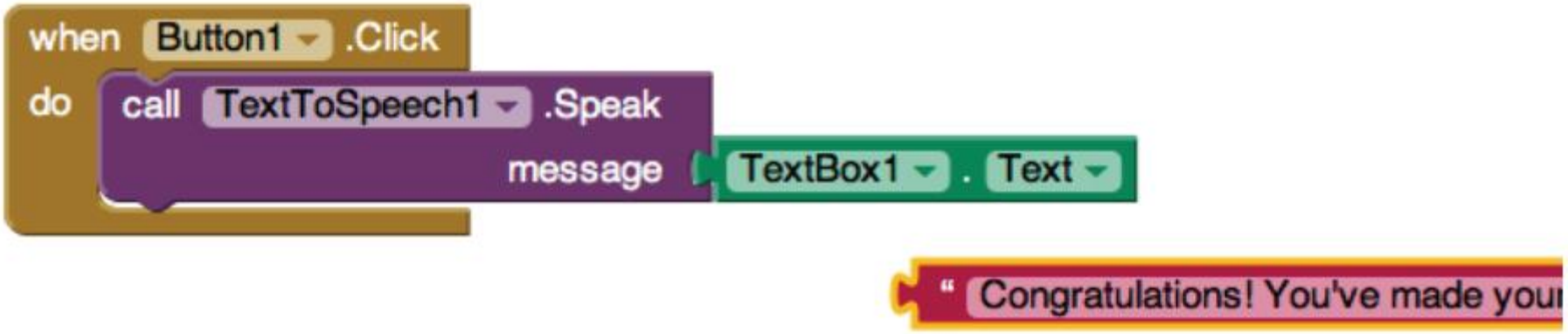

9:48
Screen1
Screen1
Button1
11:41
Screen1

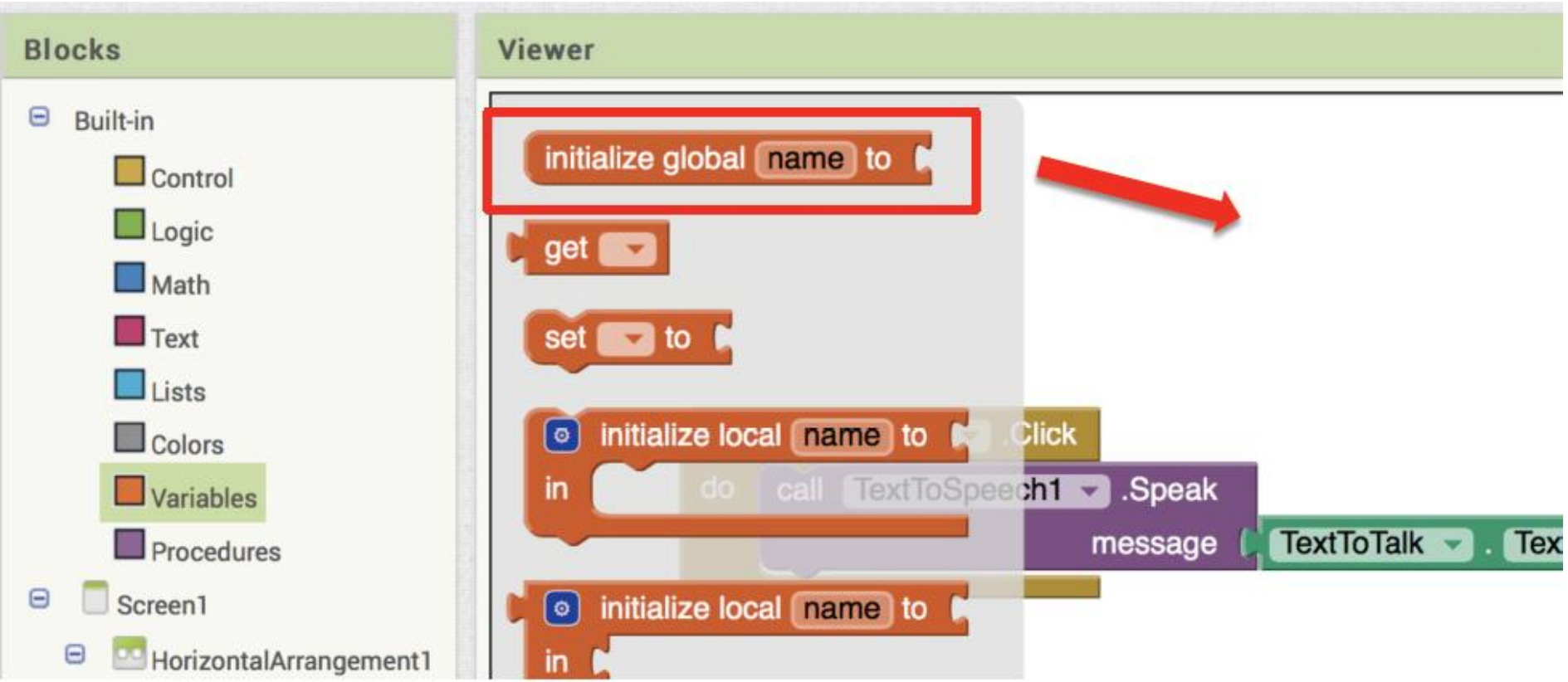

Inbox (1) - appinv1011@gmail....
MIT App Inventor 2
Connect yo
ai2.appinventor.mit.edu/#5306835864125440
MIT App Inventor 2
Beta
Projects
Connect
Build
Help
TalkToMe
Blocks
Built-in
Control
Logic
Math
Text
Lists
Colors
Variables
Procedures
Screen1
TextBox1
My projects
Start new project
Import project (.aia) from my computer ...
Import project (.aia) from a repository ...
Delete Project
Save project
Save project as ...
Checkpoint
Export selected project (.aia) to my computer
Export all projects
Import keystore
Export keystore
Delete keystore

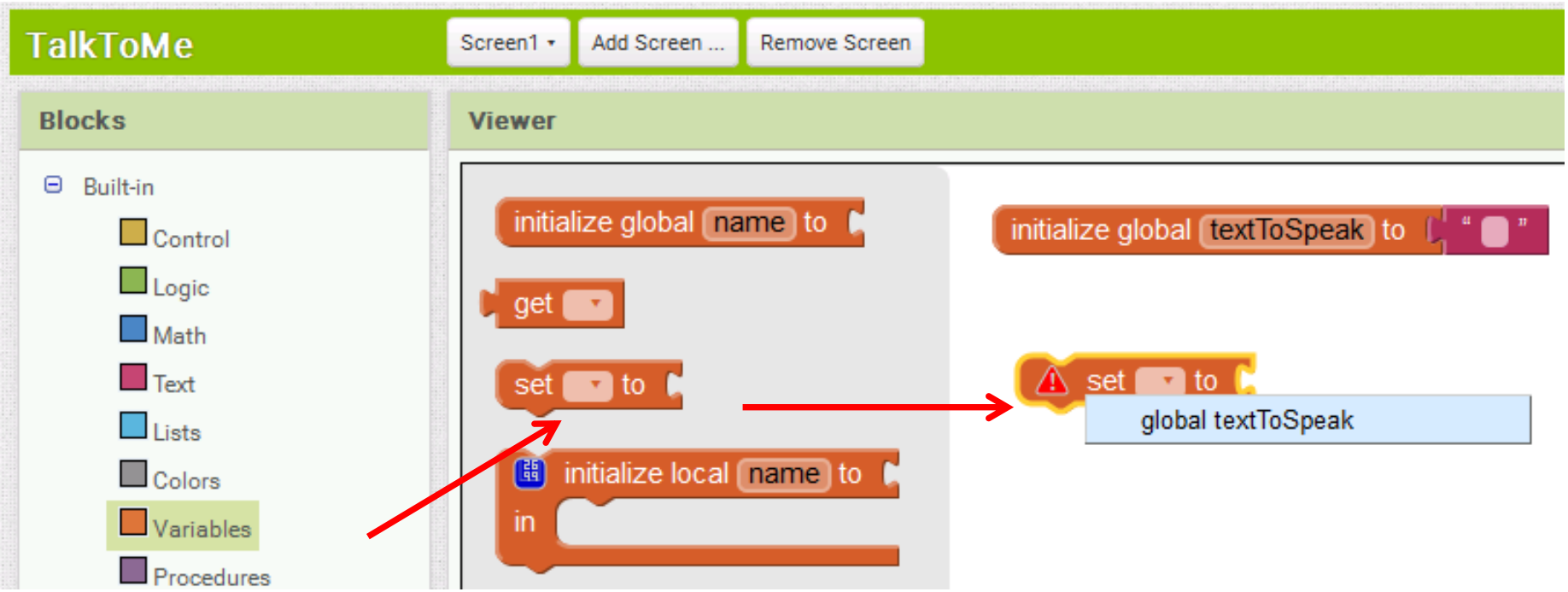
Step 2: Inputting what to say

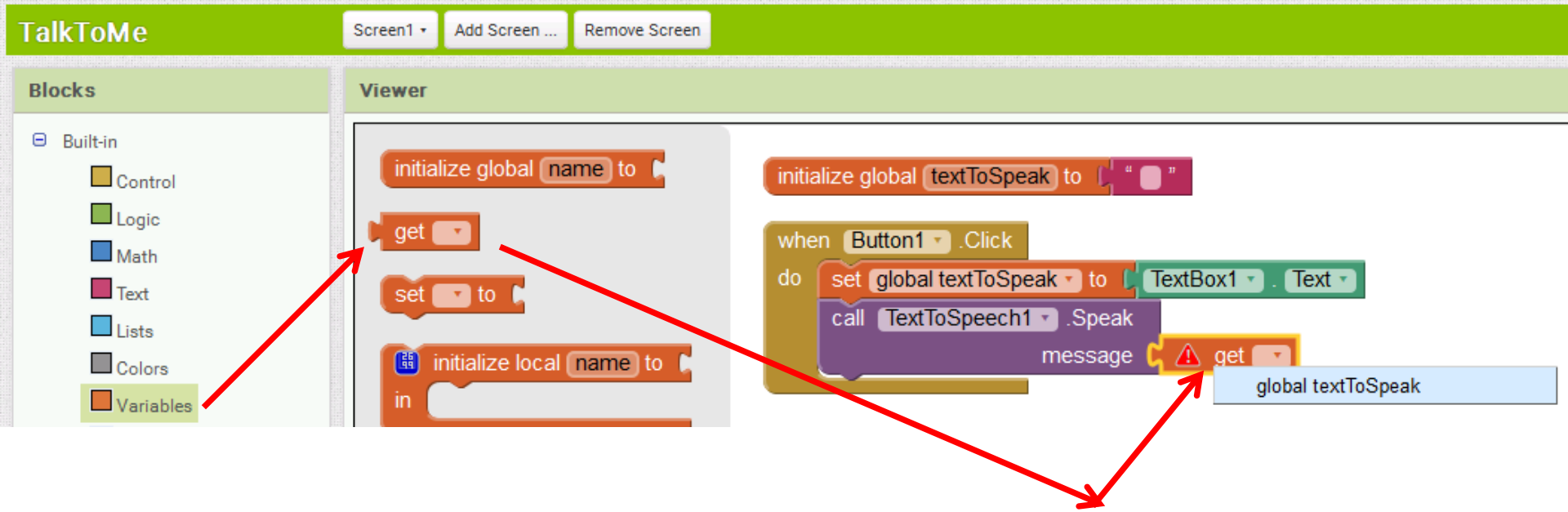


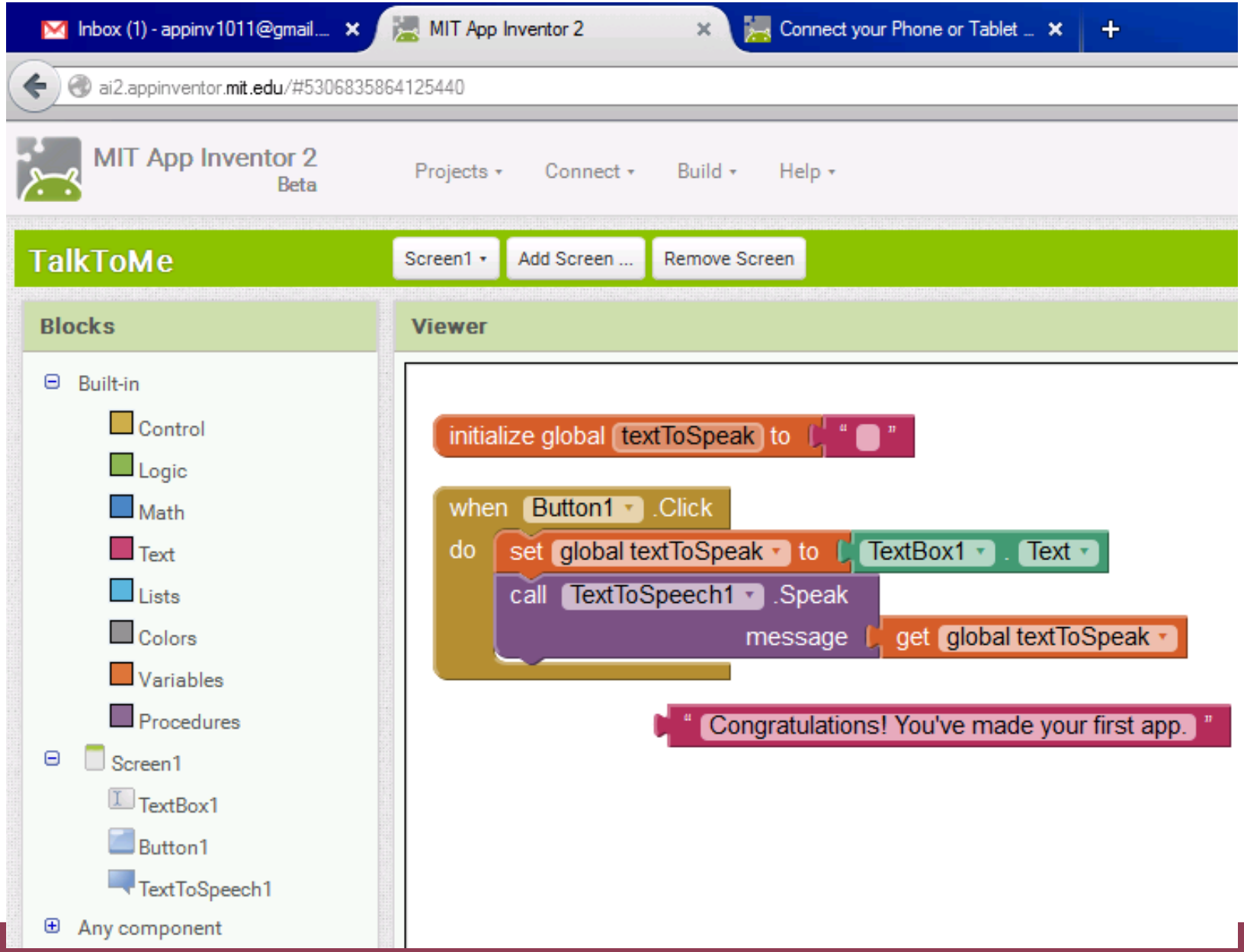

Blocks
Built-in
Control
Logic
Math
Text
Lists
Colors
Variables
Procedures
Screen1
TextBox1
Button1
TextToSpeech1
AccelerometerSensor1
Any component
Viewer
set TextBox1 . Height to
TextBox1 . Hint
set TextBox1 . Hint to
TextBox1 . MultiLine
set TextBox1 . MultiLine to
TextBox1 . NumbersOnly
set TextBox1 . NumbersOnly to
TextBox1 . Text
set TextBox1 . Text to
TextBox1 . TextColor

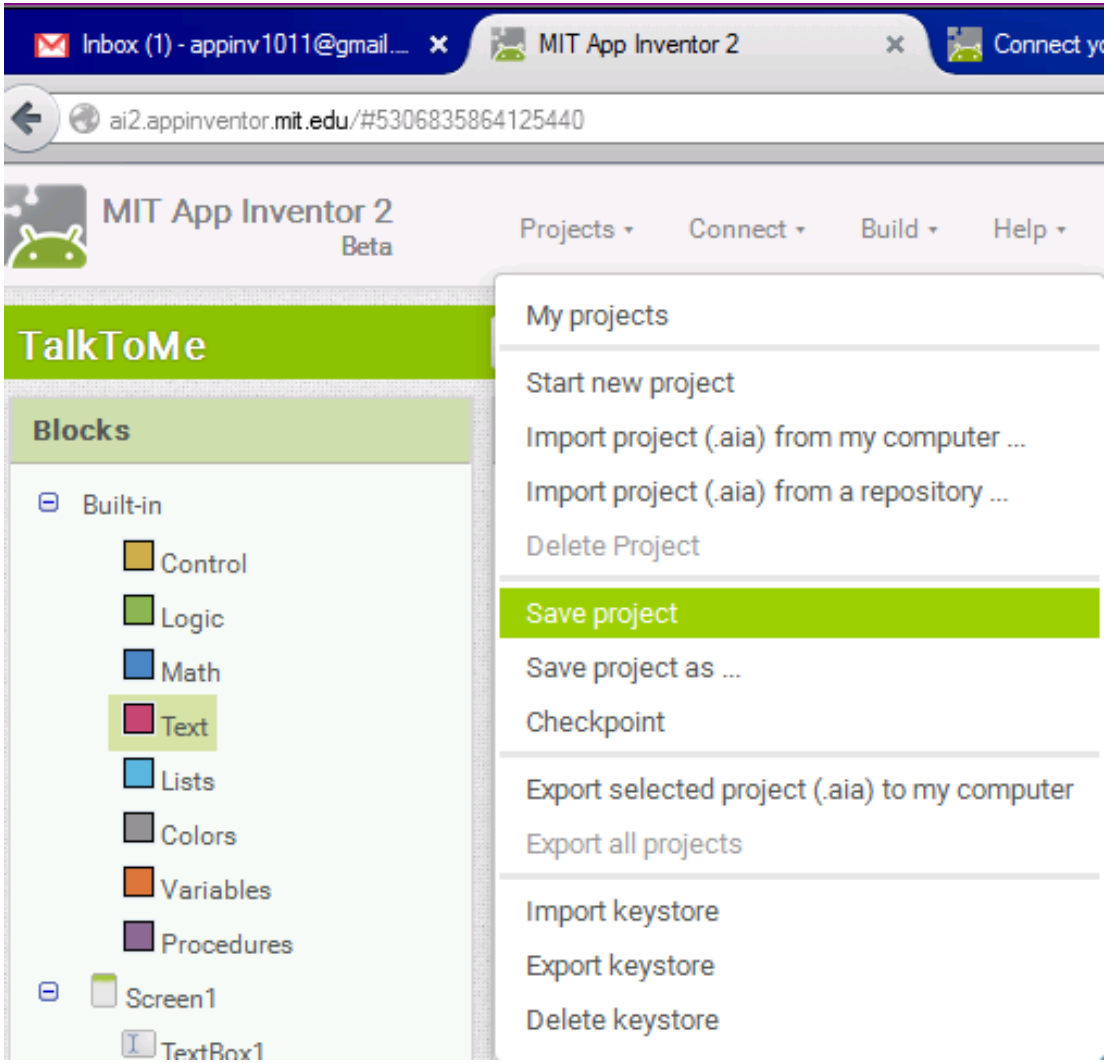

when Button1 .Click
do call TextToSpeech1 .Speak
message TextBox1 . Text
" Congratulations! You've made your


Blocks
Built-in
Control
Logic
Math
Text
Lists
Colors
Variables
Procedures
Screen1
HorizontalArrangement1
Viewer
initialize global name to
get
set to
initialize local name to
in
do
call TextToSpeech1 .Speak
message
TextToTalk
initialize local name to
in


TalkToMe
Screen1
Add Screen ...
Remove Screen
Blocks
Viewer
Built-in
Control
Logic
Math
Text
Lists
Colors
Variables
Procedures
initialize global name to
get
set to
initialize local name to
in
initialize global textToSpeak to
set to
global textToSpeak


TalkToMe
Screen1
Add Screen ...
Remove Screen
Blocks
Viewer
Built-in
Control
Logic
Math
Text
Lists
Colors
Variables
initialize global name to
get
set to
initialize local name to
in
initialize global textToSpeak to
when Button1 .Click
do
set global textToSpeak to
TextBox1 . Text
call TextToSpeech1 .Speak
message
get
global textToSpeak


Inbox (1) - appinv1011@gmail....
MIT App Inventor 2
Connect your Phone or Tablet ...
ai2.appinventor.mit.edu/#5306835864125440
MIT App Inventor 2
Beta
Projects
Connect
Build
Help
TalkToMe
Screen1
Add Screen ...
Remove Screen
Blocks
Built-in
Control
Logic
Math
Text
Lists
Colors
Variables
Procedures
Screen1
TextBox1
Button1
TextToSpeech1
Any component
Viewer
initialize global textToSpeak to " "
when Button1 .Click
do set global textToSpeak to TextBox1 . Text
call TextToSpeech1 .Speak
message get global textToSpeak
" Congratulations! You've made your first app. "


Inbox (1) - appinv1011@gmail....
MIT App Inventor 2
Connect yo
ai2.appinventor.mit.edu/#5306835864125440
MIT App Inventor 2
Beta
Projects
Connect
Build
Help
TalkToMe
Blocks
Built-in
Control
Logic
Math
Text
Lists
Colors
Variables
Procedures
Screen1
TextBox1
My projects
Start new project
Import project (.aia) from my computer ...
Import project (.aia) from a repository ...
Delete Project
Save project
Save project as ...
Checkpoint
Export selected project (.aia) to my computer
Export all projects
Import keystore
Export keystore
Delete keystore

Lunch

Lunch Instructions

- You will be given a ticket to eat at the University Residence Cafeteria
- You are expected to get back to this room by 1:00pm
- Stick together – we will go over in groups

Have a good lunch

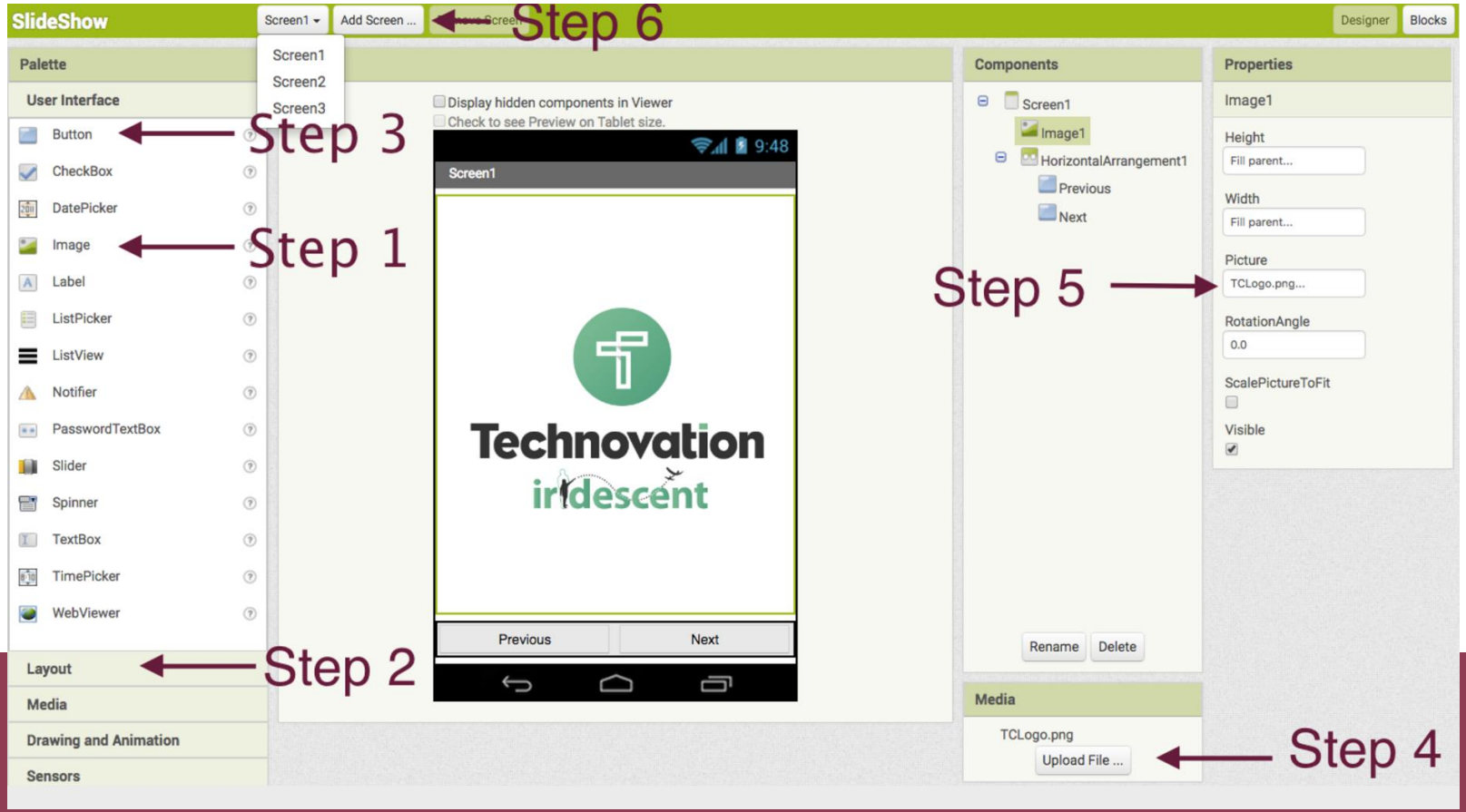
Making a slideshow

Designing your screen

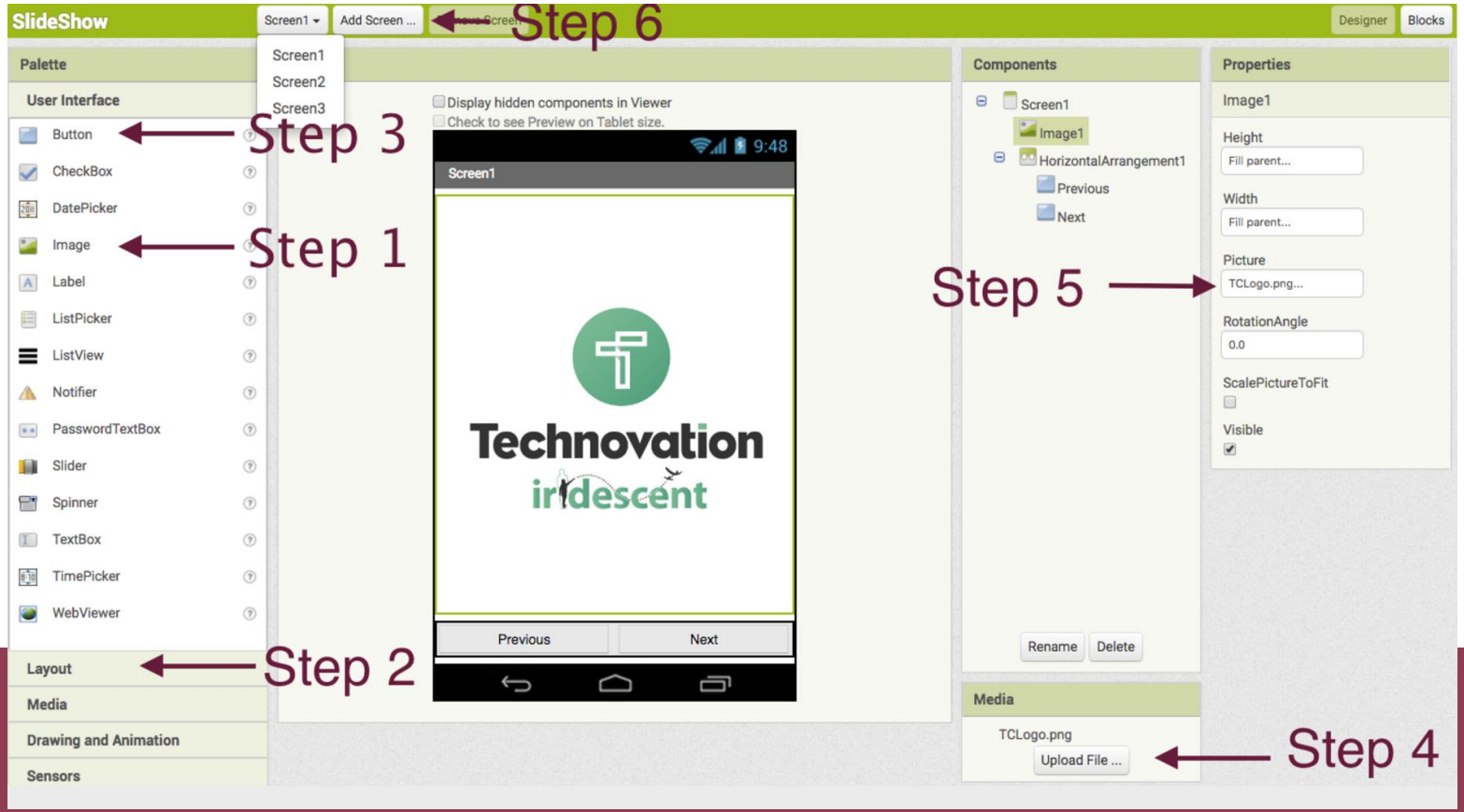
- For this app you want to create a slideshow by allowing your user to be able to navigate through images
- When you design your screen, you will need to put in an image
- You also need ‘previous’ and ‘next’ buttons for your user to click. This part will be done in the designer


SlideShow
Screen1
Add Screen ...
Step 6
Designer
Blocks
Palette
User Interface
Button
CheckBox
DatePicker
Image
Label
ListPicker
ListView
Notifier
PasswordTextBox
Slider
Spinner
TextBox
TimePicker
WebViewer
Layout
Media
Drawing and Animation
Sensors
Step 3
Step 1
Step 2
Screen1
Screen2
Screen3
Display hidden components in Viewer
Check to see Preview on Tablet size.
9:48
Screen1
Technovation
irdescent
Previous
Next
Components
Screen1
Image1
HorizontalArrangement1
Previous
Next
Rename
Delete
Media
TCLogo.png
Upload File ...
Step 4
Step 5
Properties
Image1
Height
Fill parent...
Width
Fill parent...
Picture
TCLogo.png...
RotationAngle
0.0
ScalePictureToFit
Visible

1. Add an image to your screen.
   - Make it fit your screen by changing the width and the height!
   - We choose to make our image “fill parent”. When you select “fill parent” you are telling the component to fill the space of whatever it is in.
2. Add a horizontal layout to your screen below the image.
3. Add two buttons into the horizontal layout.
   - Click on the buttons in the component menu
     - Edit the text to make the buttons say ‘previous’ and ‘next’.



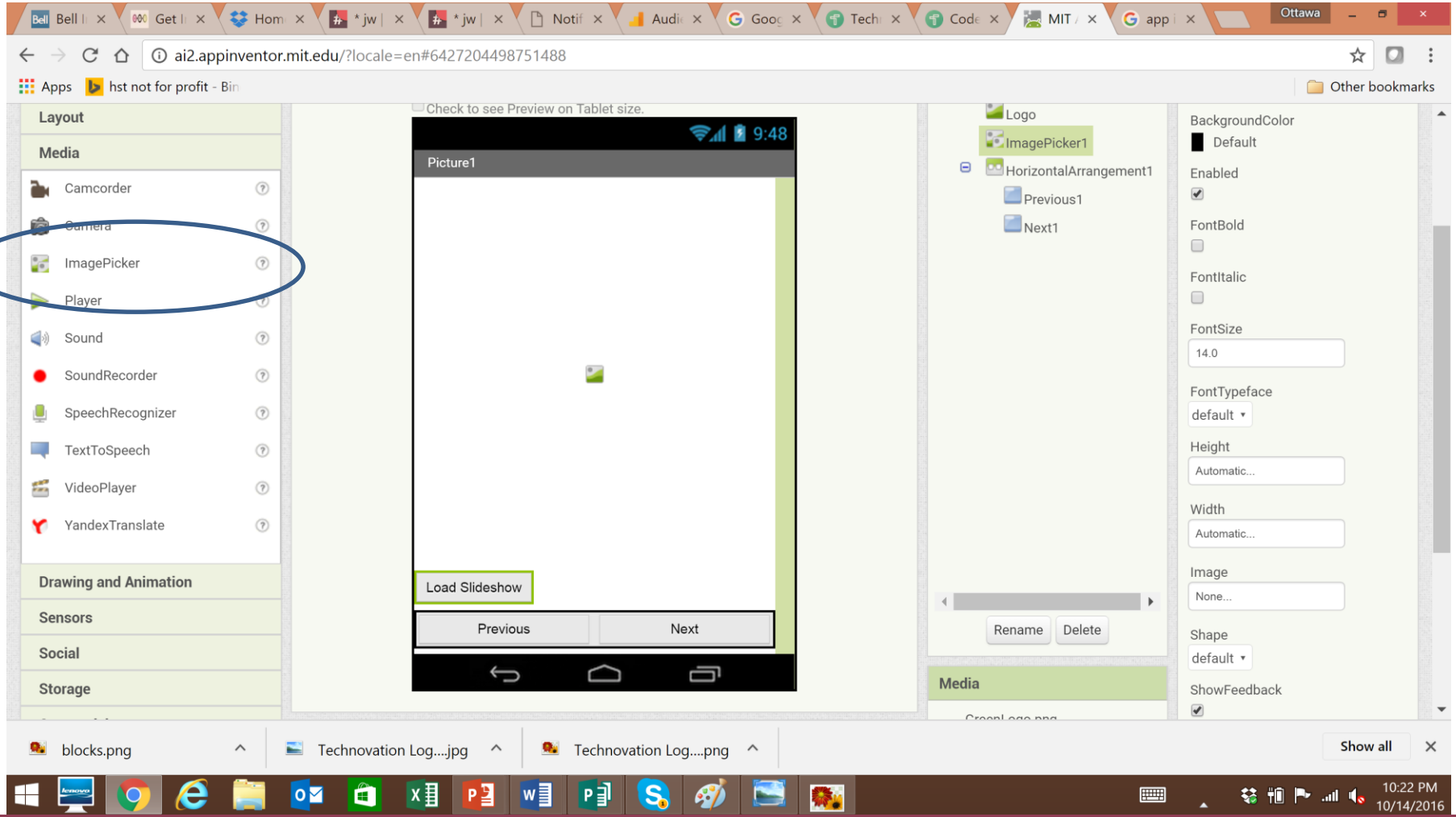
4. Upload a picture you want to use in your slide show!
5. Select the picture you want to use on the first screen of your slide show.
   - Click on “Image1” in the “Components” menu and then “Picture” in the properties menu. You can then select a picture you uploaded.
6. Add two more screens.
   - For now you do not need to do anything with those screens, but after you code the blocks you will add buttons and images to them just like you did for this screen



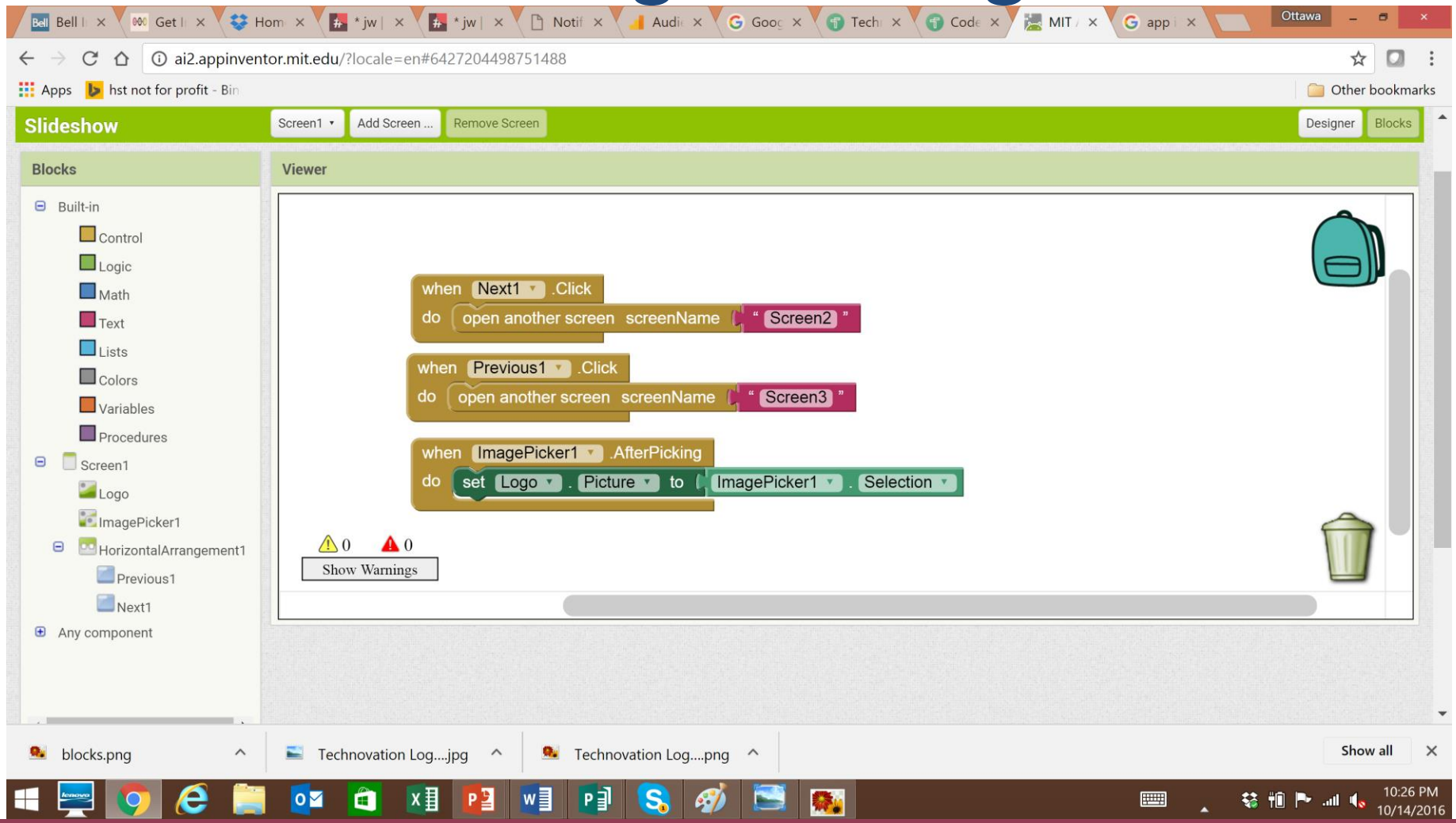
Click on “Blocks” Button

7. Click on button you named ‘previous’ to see all the event handlers you can use. Grab the ‘when button.click’ event handler and drag it to your workspace.
8. Click on ‘control’ and find the ‘open another screen screenName’ block.
9. Get an empty text box and click it into the “open another screen screenName” block. Type the name of the last screen that will be in your slide show. For us, it was Screen3.
10. Click on your next button and grab the “when button.click” event handler again
11. Get another “open another screen screenName” block from control.
12. Get an empty text block and type in the next screen that will appear in your slide show. (For us, that was Screen2.)

Using an image on your phone



Setting the image



Types of Data

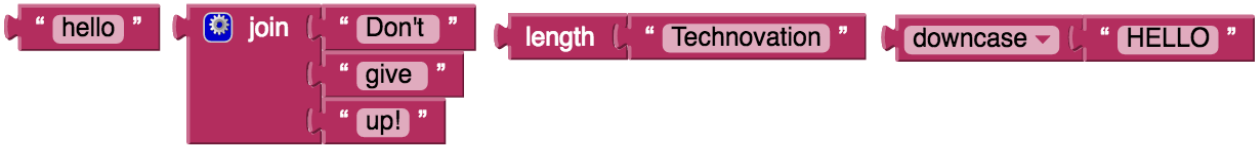
Types of Data?

	WhatsApp	Angry Birds	Slideshow
About	Message and call your friends	Game: shoot birds at pigs	Cycle through images (you made this)
Types of data	• Your username • Your friend's usernames • The message you want to send • What time it is • Your location	• Your score • levels you've completed	• Your favorite images

Types of Data

- Numbers

- Strings



- Booleans

Variables

- Variable: Data that can change in value
  - Your Age (string, number, or boolean?)
  - Your Address (string, number, or boolean?)
  - Student[yes or no] (string, number, or boolean?)

Variables

- Variable: Data that can change in value
  - Your Age (string, **number**, or boolean?) 
  - Your Address (string, number, or boolean?)
  - Student[yes or no] (string, number, or boolean?)

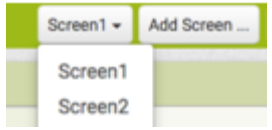
Variables

- Variable: Data that can change in value
  - Your Age (string, **number**, or boolean?) 
  - Your Address (**string**, number, or boolean?) 
  - Student[yes or no] (string, number, or boolean?)

Variables

- Variable: Data that can change in value
  - Your Age (string, **number**, or boolean?) 
  - Your Address (**string**, number, or boolean?) 
  - Student[yes or no] (string, number, or **boolean**?) 

Variables

- Variable: Data that can change in value
  - Your Age (string, **number**, or boolean?) 
  - Your Address (**string**, number, or boolean?) 
  - Student[yes or no] (string, number, or **boolean**?) 
- Local variables work only on one screen 

[Screen1]VeribA=3; [Screen2]VeribB=4; [Screen3] VeribA+VeribB=Error

- Global variables can be shared on all screens

[Screen1]Global VeribA=3; [Screen2]Global VeribB=4; [Screen3]VeribA+VeribB=7

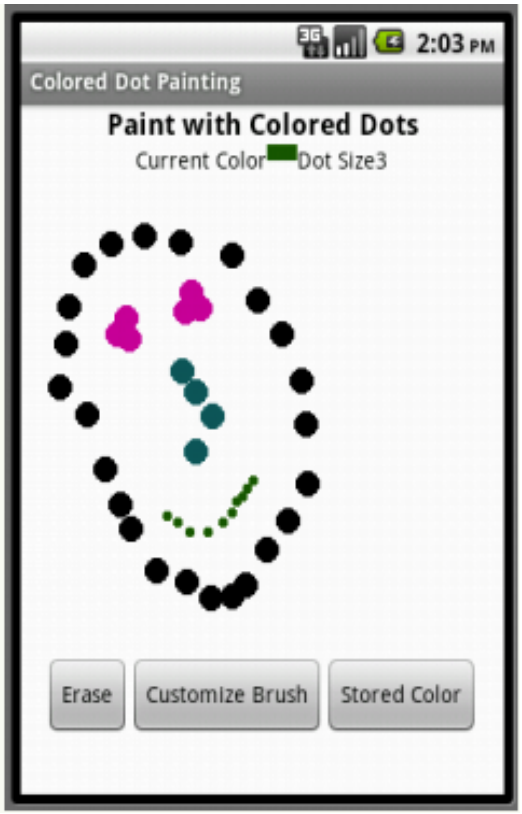
App Inventor Colored Dots

AI: Colored Dots
(Create multiple screens)

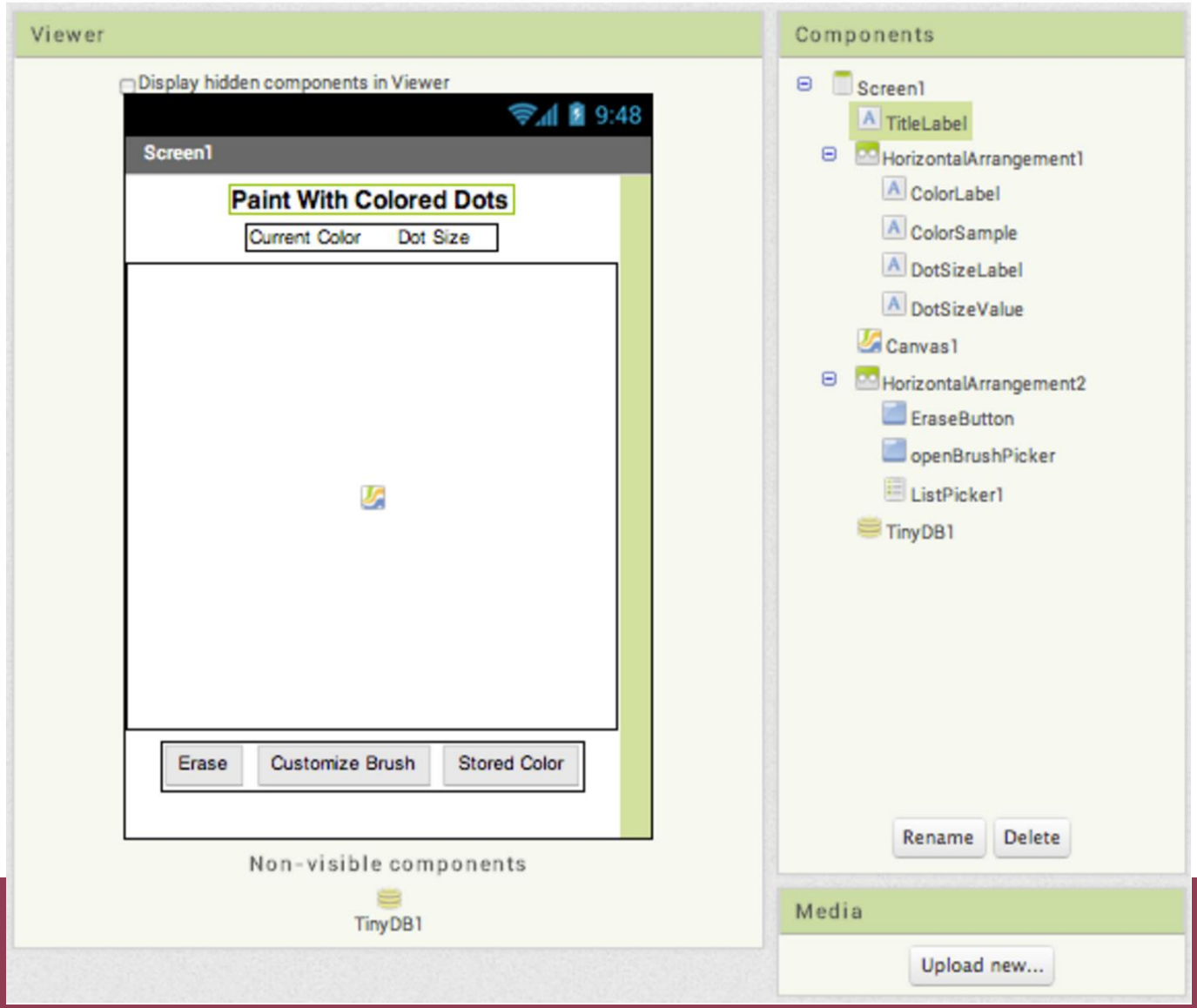
http://appinventor.mit.edu/explore/ai2/colored-dots.html

The Colored Dot tutorial teaches you how to create apps that have multiple screens. You'll learn how to:

- make an app with multiple screens
- pass values from one screen to another using TinyDB
- how to fill and use the ListPicker element

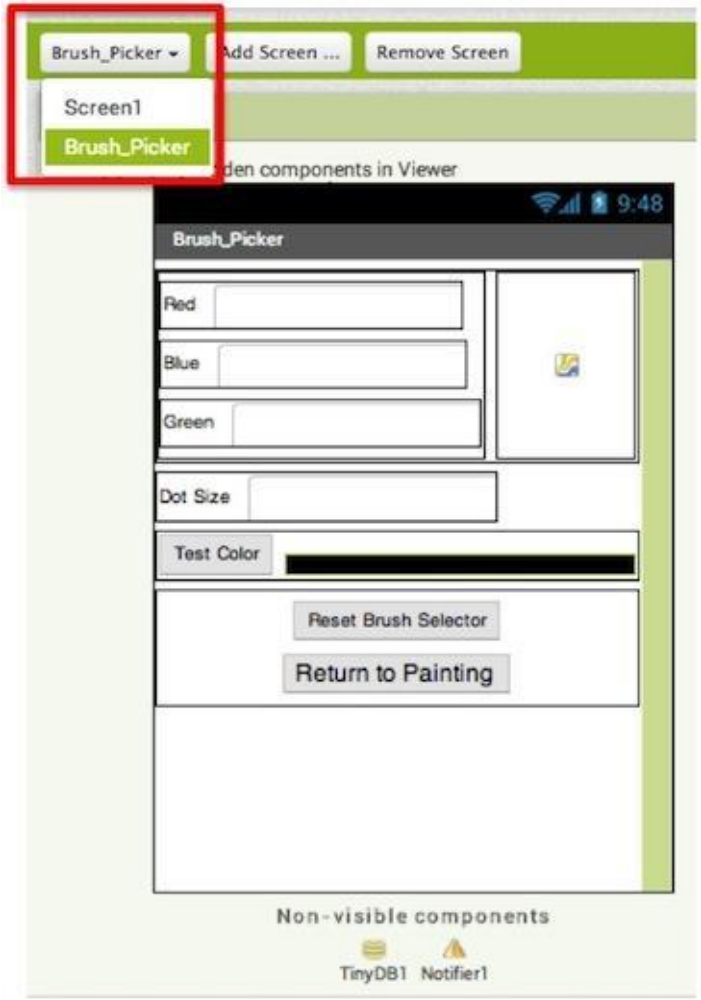


Starting Screen

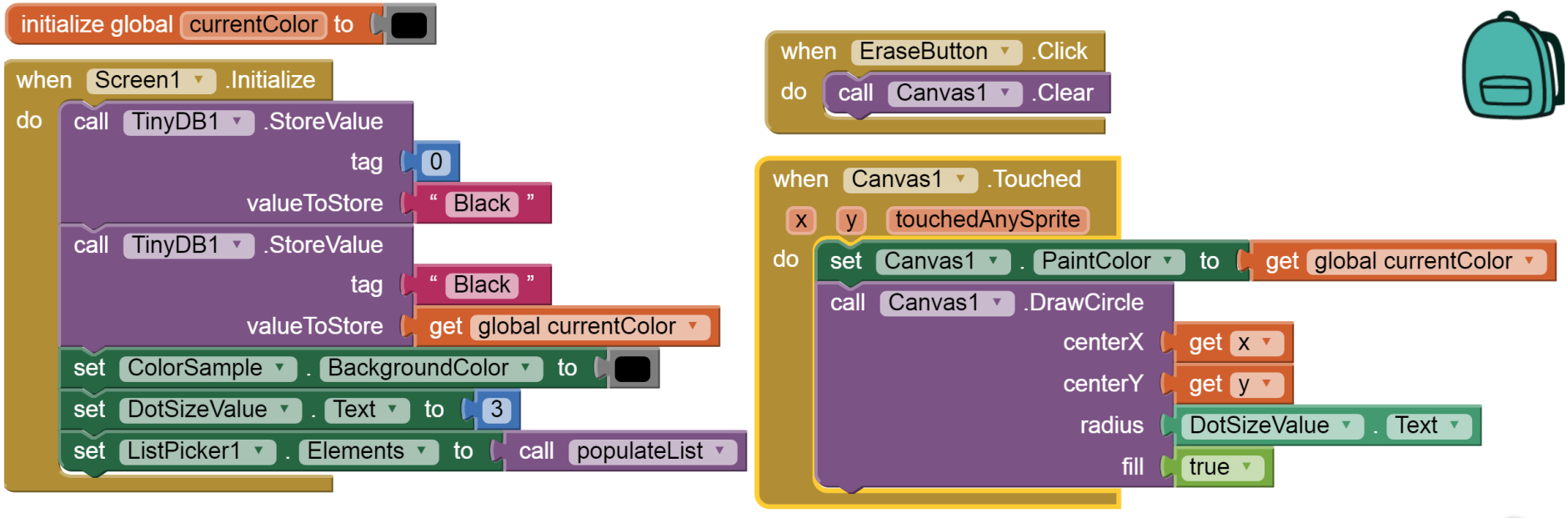


Multiple Screens

- You can add screens in the designer and use the screen transitions in blocks editor to decide which screen to go to next
  - For Example: pushing the menu button go to the menu screen
- Screen 1 will always be the screen the app starts on – it's probably best to make it a welcome screen



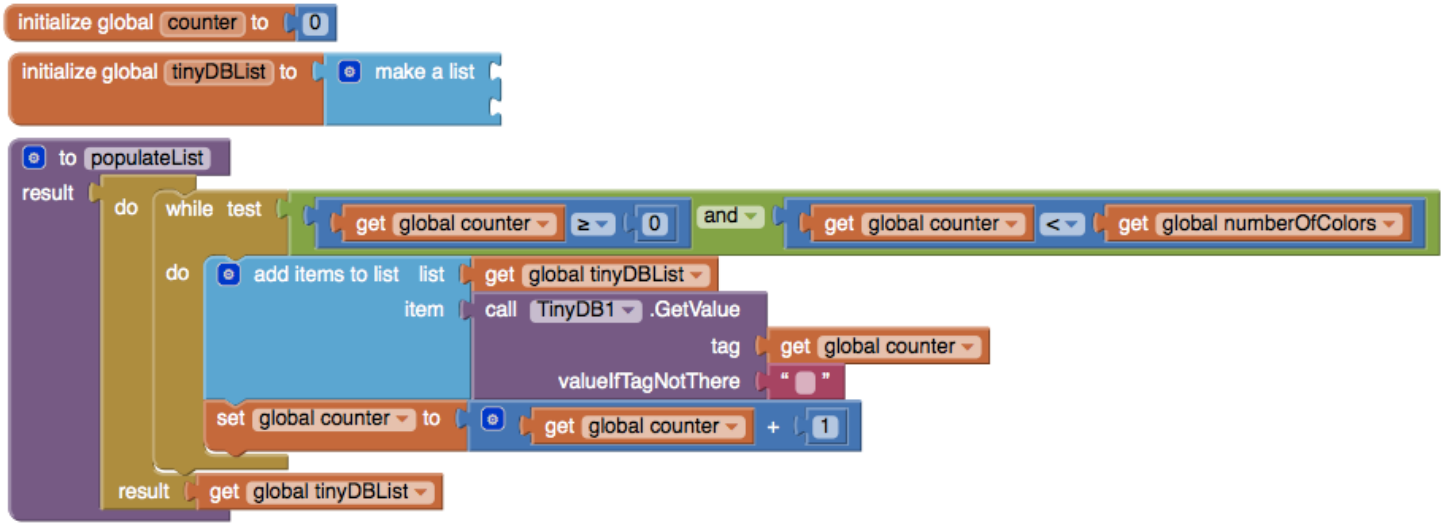
Starting with one colour



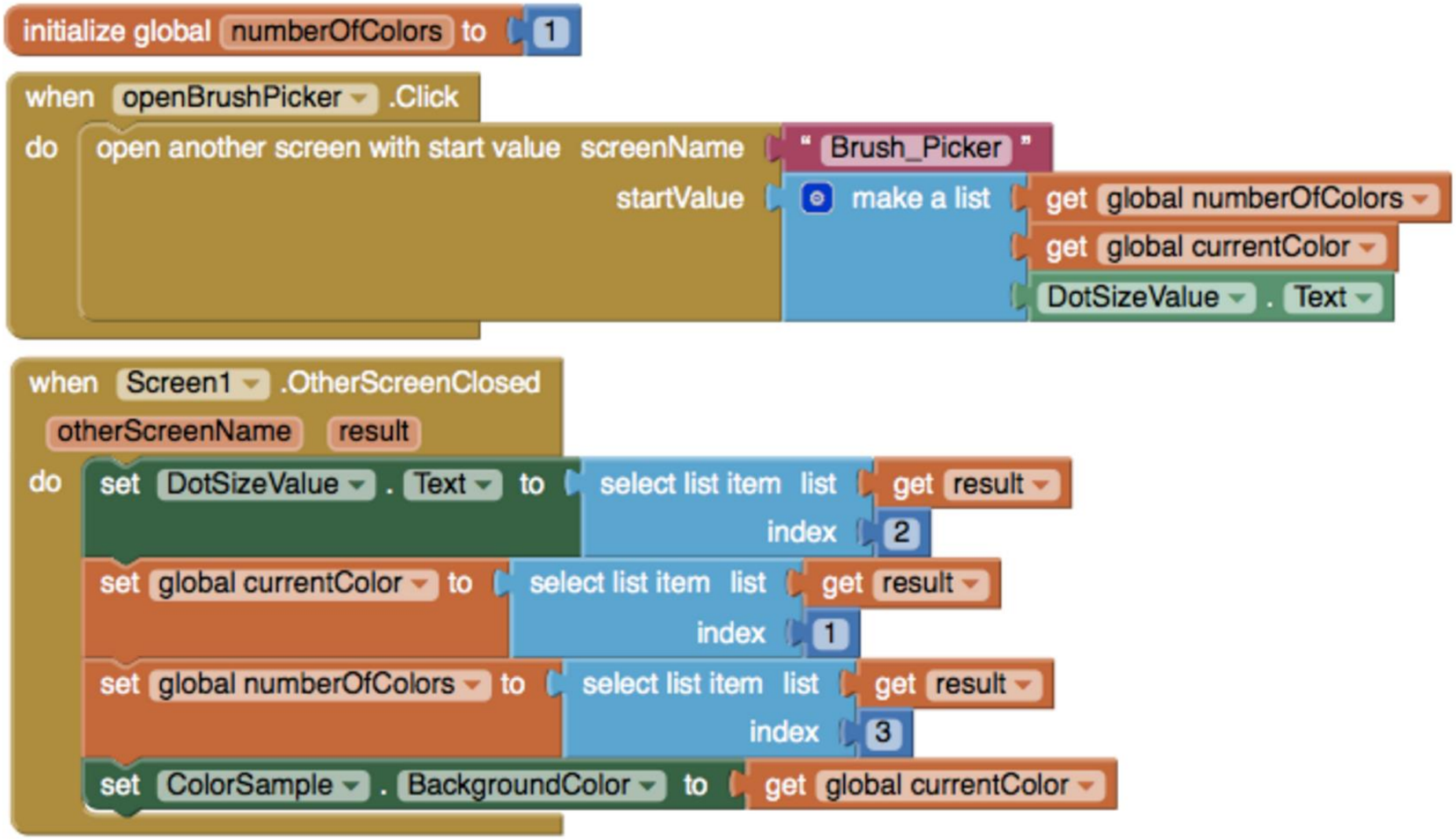
- On Screen 1 set up your starting colour
- Set up what happens when the screen is touched

TinyDB

- Besides opening screens and returning values, the different screens in a multiple screen app can communicate through TinyDB. To do this, give every screen its individual TinyDB component.
- ColoredDots uses TinyDB to let you name the colors you create and save them for later use. The saving and naming will be done in Brush_Picker

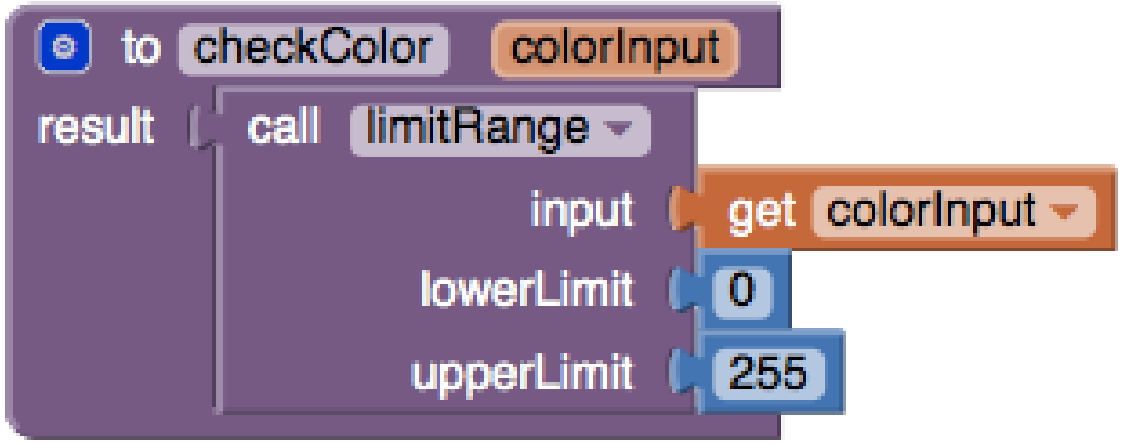


Getting ready to add colours

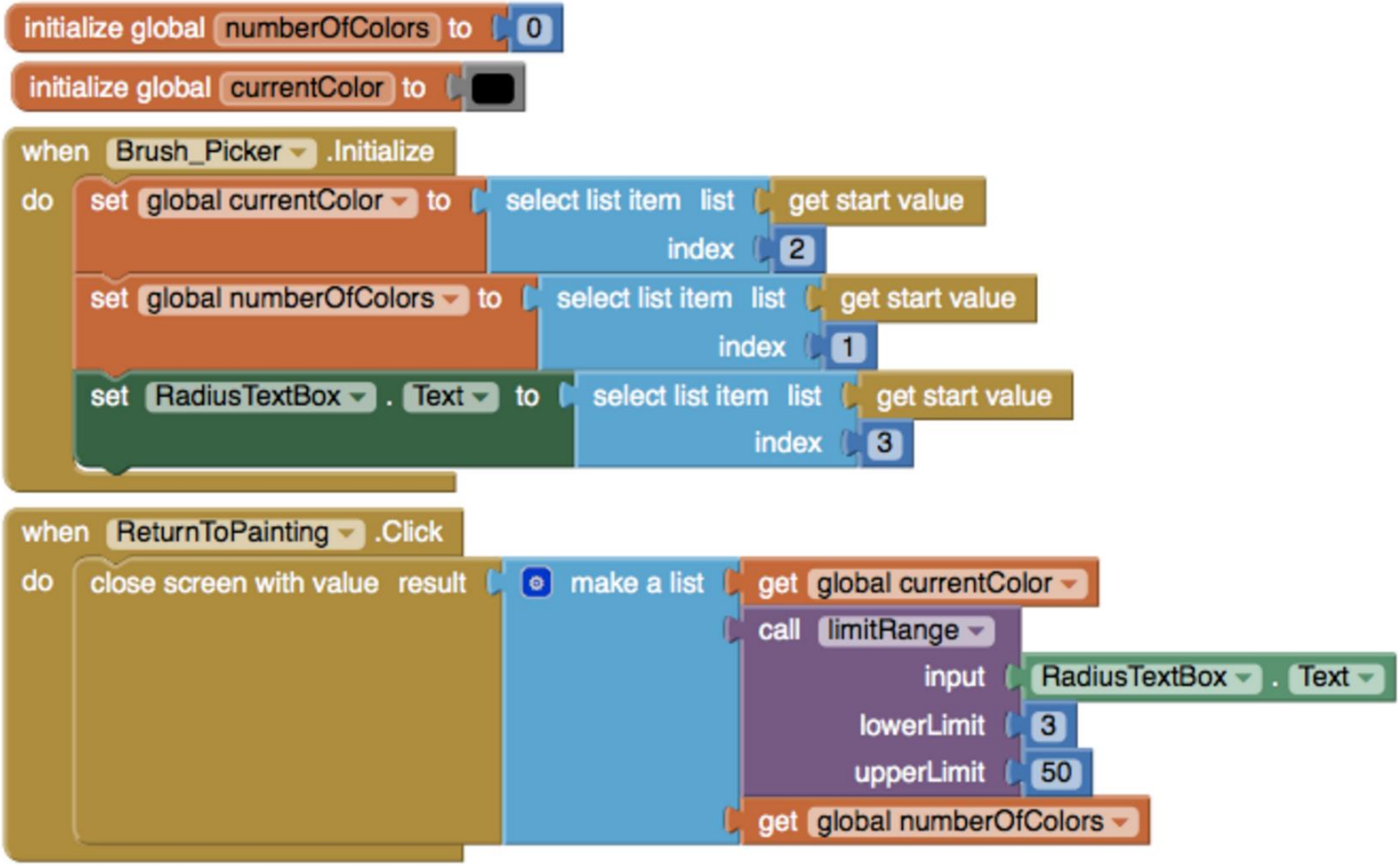


Brush Picker

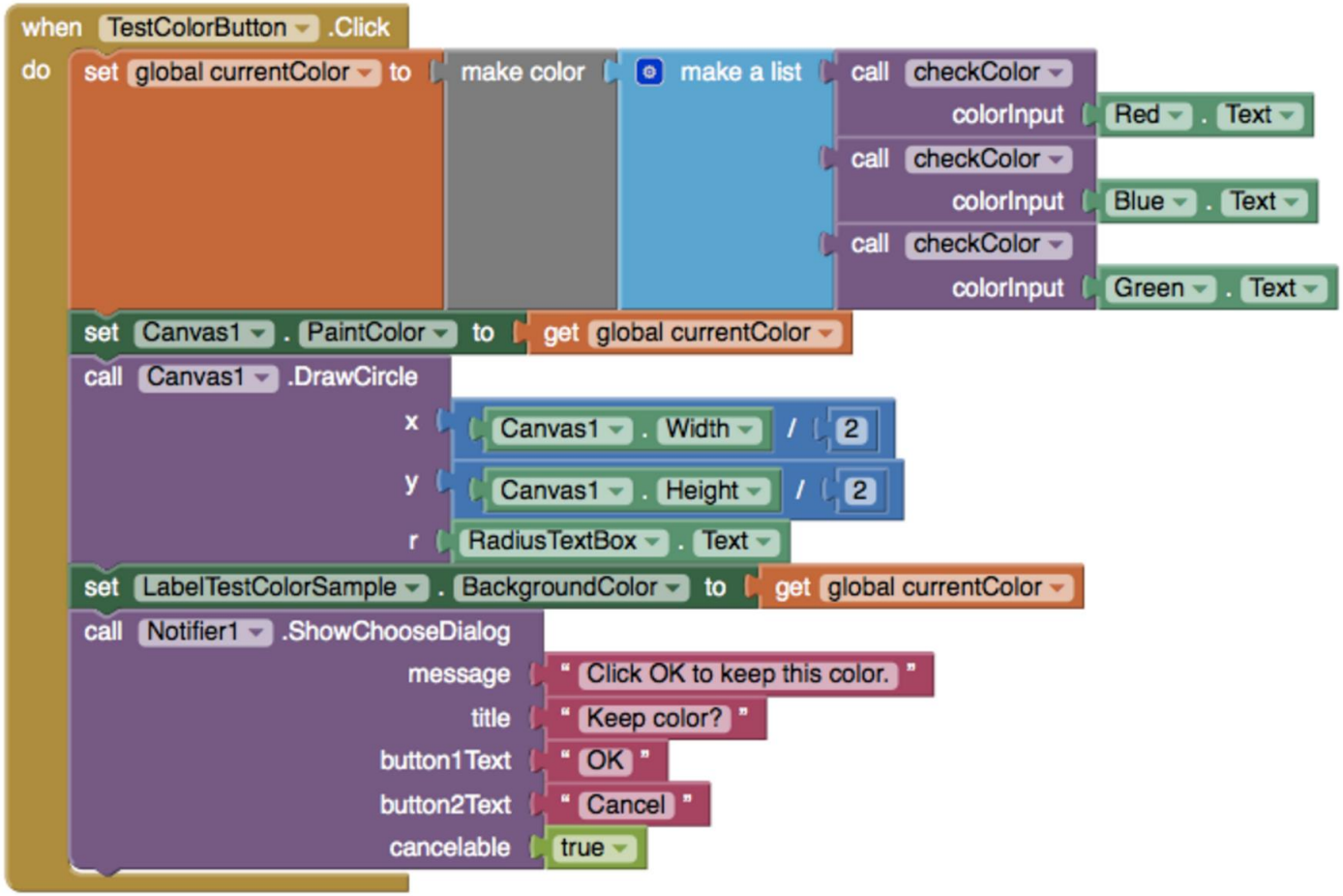
- The main job of Brush_Picker is to create a color from the red-green-blue values entered in the text boxes and provide that color to Screen1.
- One thing Brush_Picker needs to check is that it's using good values for colors and dot size. Each of the red, green, blue values should be a number between 0 and 255.



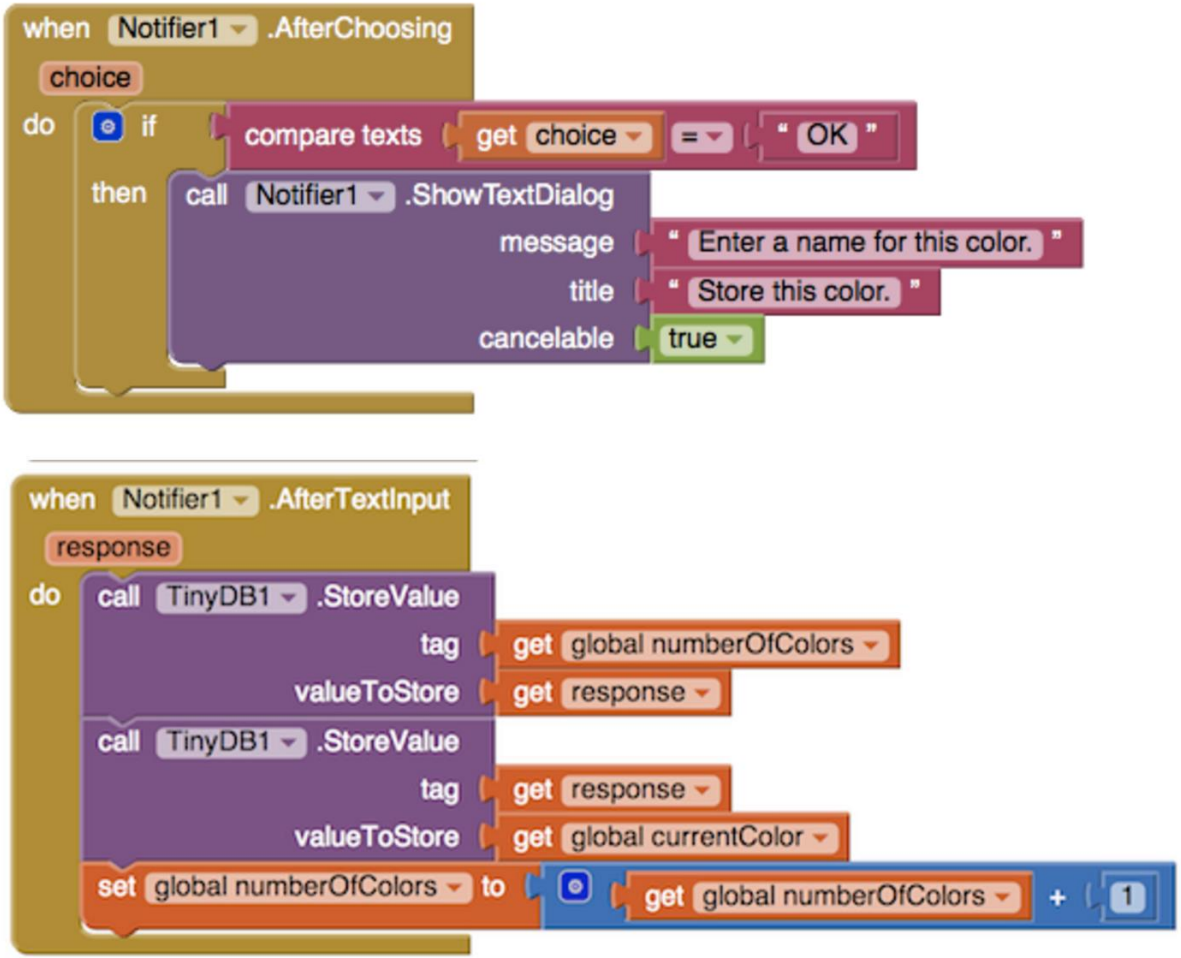
Setting up the brush picker



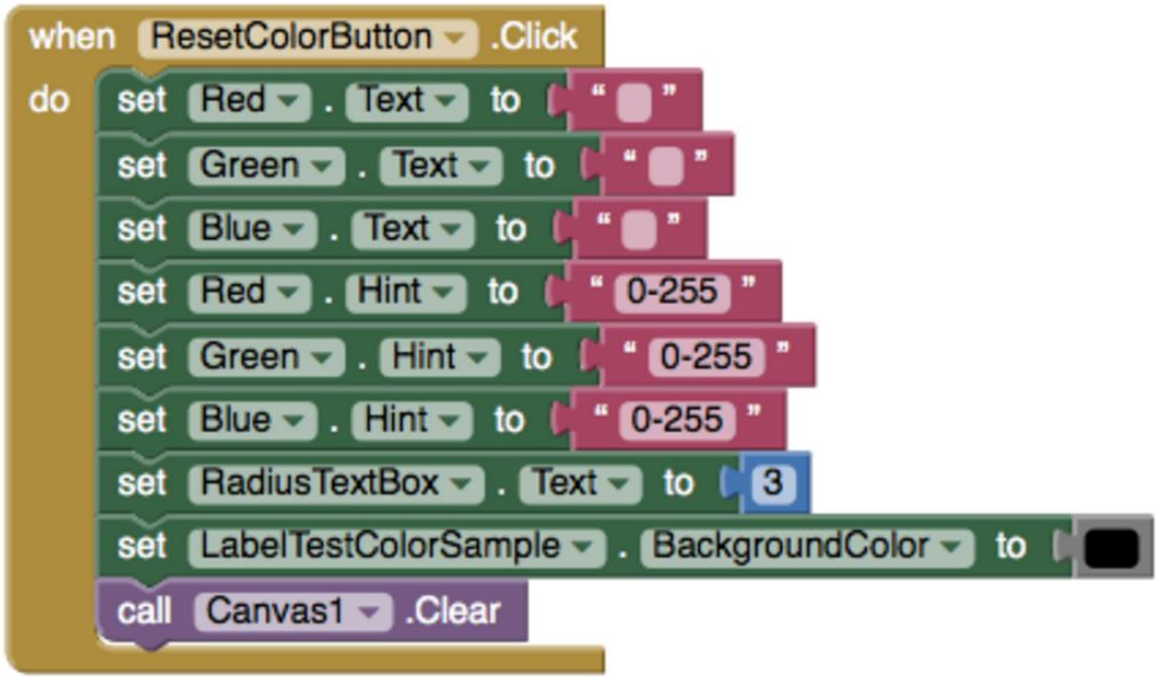
Testing the colour



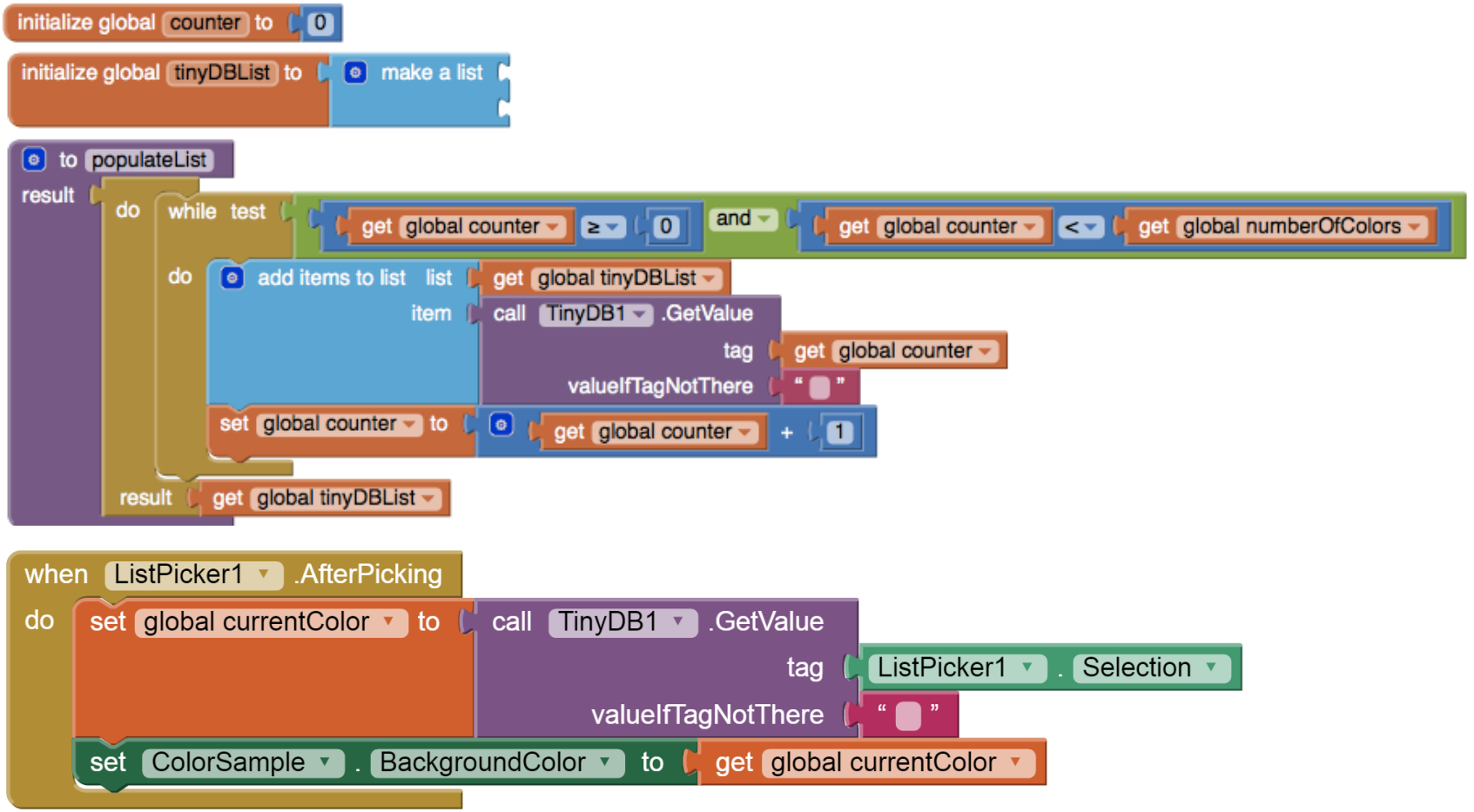
Saving the colour



Resetting colours



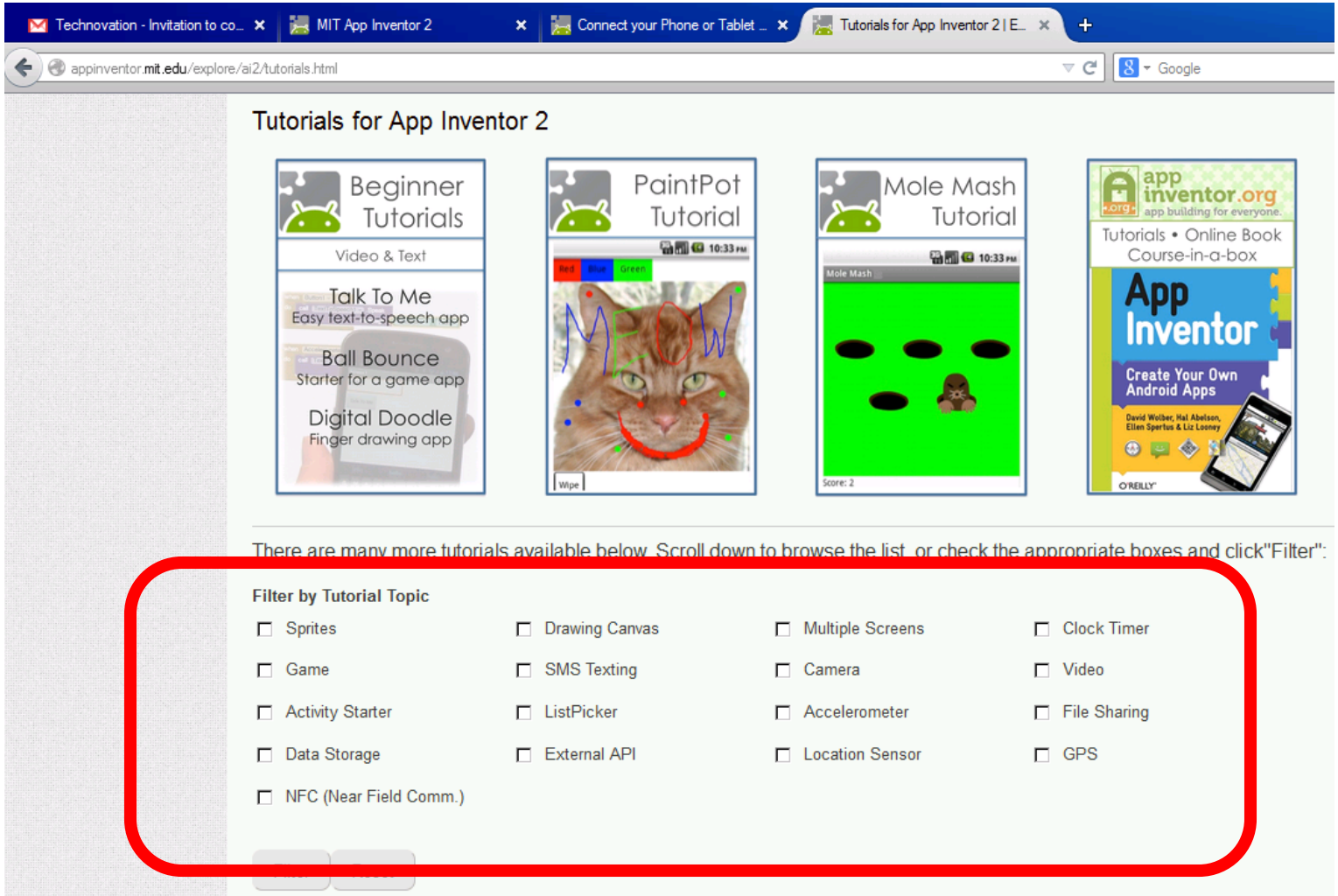
Selecting a stored color



Continuing your learning

(Tutorials)

appinventor.mit.edu/explore/ai2/tutorials

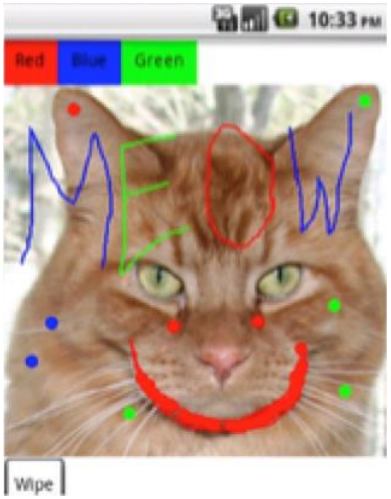


Tips for App Development

- Attention to detail is important
- Time will be spent troubleshooting & testing
  - Don’t let this discourage you
- Enjoy the challenge, impress your friends, and never give up on troubleshooting the application
- Save often

appinventor.mit.edu/explore/ai2/tutorials

Have Fun!



Technovation: Next Steps

Start brainstorming ideas

- Think about issues within your community
  - School, neighbourhood, church, other communities …
- Pull out any and all ideas
- Draw from personal experience

Themes

- Poverty - Eradicating extreme poverty, implementing social protection systems for all, and ensuring that all men and women have equal access to economic resources.
- Environment - Improving education and awareness about climate change and strengthening resilience to climate-change hazards in all countries.
- Peace - Significantly reducing violence, ending abuse of children, reducing corruption and bribery, ensuring equal access to justice for all, and ensuring public access to information.
- Equality - Ending discrimination against girls and women, enhancing the use of enabling technology to promote the empowerment of women, ensuring universal access to reproductive rights, and ensuring women's full and effective participation and opportunities for leadership.
- Education – Ensure healthy lives and promote well-being for all at all ages
- Health – Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.

Previous Technovation finalists

Apps to fight hunger	Food Rescue by AurGrrls This app allows users to donate leftover food from movie shoots in Los Angeles to local foodbanks.
	InDaFridge by IDF This app stops people from wasting food by providng customized recipes based on what is in their refrigerator.
Apps to stop violence	Neo Safety This app allows users to see if there has been recent crimes in the area they are currently in.
	Women Fight Back by Girls for Change From Dharavi This app allows women to report and ask for help when they are harassed or attacked.
Apps to go green	Discardious by Team Charis This app helps users clean up their trash in their local communities by providing carts to pick up the trash.
	Loc8Don8 by California Coders This app helps people find places to donate things they don't need anymore.

Housekeeping

- Weekly meetings Wednesday's 7-9pm
  - IBM, 3755 Riverside Dr Feb 1st
- Join the facebook group Technovation Ottawa
- Register on Technovationchallenge.org
- Read your emails!!!!

Math App Challenge

(create a calculator)

For each member of your group work on a separate math problem

- Member 1: (Add) TextBox1 +2= Label1
- Member 2: (Multiply) TextBox1 x 2= Label1
- Member 3: (Divide) TextBox1 / 2 = Label1
- Member 4: (Subtract) TextBox1 - 2= Label1
- Member 5: (Exponent) TextBox1 ^ 2 = Label1
- Member 6: (Multiply) TextBox1 x 4= Label1
- Member 7: (Subtract) TextBox1 – 4= Label1
- Member 8: (Multiply) TextBox1 x 5= Label1

Example: (Multiply)
[User Input is "3"] x 2 = [Answer is 6]