# APP Design (Day 1)

- Introduction to App Inventor
- Training Plan
- Interface and Control
- Block programming
- Mini-Project



## Introduction to App Inventor

- Created by MIT Computer Science & Artificial
   Intelligence Laboratory
- Visual programming environment for Android phones and tablets
- Programming editor based on Blockly
- Other Blockly based editors:
  - Blockly Games
  - Code.org
  - MakeCode (used for micro:bit)

## Introduction to App Inventor

- Primarily used for teaching coding, but many have created practical and useful apps using it
  - App to help blind navigate around the school in Texas
  - Crowd-sourcing app to locate safe drinking water in Moldova
  - App to help parents track students on buses

# Training Plan

- Day 1: Intro and Basics
- Day 2: Sprites and Graphics
- Day 3: GPS, Map, inter-app communications
- Day 4: Camera, File, Physics
- Day 5: Project Work
- Day 6: Project Presentation

# Setting Up (Part 1 of 2)

- Visit http://appinventor.mit.edu or do a search for "app inventor"
- Click "Create apps!"
- Login with your Google Account (create a new account if you don't have one)

HourOfCode	Scree	Add Screen Remove Screen		Designer
Palette	Vi	ewer	Components	Properties
User Interface		Display hidden components in Viewer	Screen1	Screen1
Button	۲	Check to see Preview on Tablet size.		AboutScreen
CheckBox	۲	Screen1		
DatePicker				AccentColor Default
Mage Image				AlignHorizontal
A Label				Left:1 *
ListPicker				AlignVertical
ListView				AppName
A Notifier				HourOlCode
PasswordTextBox				BackgroundColor
Slider	۲			Backgroundimage
Spinner	۲			None
TextBox	T			CloseScreenAnimation
TimePicker	۲			beaut *
WebViewer				icon.prg
Lavout			Rename Delete	OpenScreenAnimation
Modia			Media	Default *
Denvice and Animation			codi.jpg	PrimaryColor Default
Drawing and Animation			icon.png	PrimaryColorDark
Maps			Bee-Sound mp3	Default
Sensors			oprodu r ne	ScreenOrientation Unspecified *
Social				Scrollable
Storage				0
Connectivity				ShowListsAsJson
LEGO® MINDSTORMS®				ShowStatusBar
Experimental				8
Extension				Sizing

#### **Palette**

Where you select and drag components into the viewer

lette	Vi	ewer	Components	Properties
ser Interface		Display hidden components in Viewer	Screen1	Screen1
Button	۲	Check to see Preview on Tablet size.	8	AboutScreen
CheckBox	T	Screen1		
] DatePicker	T			AccentColor Default
Image				AlignHorizontal
Label	T			Left:1 *
ListPicker				AlignVertical
ListView	•			AppName
Notifier	۲			HourOfCode
PasswordTextBox				BackgroundColor
Slider				Backgroundimage
Spinner	۲			None
] TextBox	T			CloseScreenAnimation
TimePicker	•			Default *
WebViewer				icon.prg
			Rename Delete	OpenScreenAnimation
ayout		1 C T	Media	Default •
ledia			codi.jpg	PrimaryColor  Default
rawing and Animation			icon.png	PrimaryColorDark
laps			Bee-Sound.mp3	Default
ensors			Upload File	ScreenOrientation
ocial				Scrollable
torage				0
onnectivity				ShowListsAsJson
EGO® MINDSTORMS®				ShowStatusBar
xperimental				8

**Viewer** Represents your phone. You can drag components in here.

HourOfCode		Screen1 • Add Screen	Remove Screen		Designer
Palette		Viewer		Components	Properties
User Interface		Display	hidden components in Viewer	Screen1	Screen1
Button	۲	Check to	o see Preview on Tablet size.	18	AboutScreen
CheckBox	۲	Screen1			
DatePicker	۲				AccentColor Default
image	۲				AlignHorizontal
A Label					Left:1 *
ListPicker	T				AlignVertical
ListView	۲				AppName
A Notifier	۲				HourOlCode
PasswordTex	tBox ()				BackgroundColor
Slider	۲				Backgroundimage
Spinner	۲				None
TextBox	۲				CloseScreenAnimation
TimePicker	۲				Default *
WebViewer	(7)				icon.prg
				Rename Delete	OpenScreenAnimation
Layout				Madia	Default •
Media		-		codi ina	PrimaryColor
Drawing and Anin	nation			icon.png	PrimaryColorDark
Maps				Bee-Sound mp3	Default
Sensors				Upicad File	ScreenOrientation Unspecified *
Storage					Scrollable
Connectivity					ShowListsAsJson
LEGO® MINDSTO	ORMS®				0
Experimental					ShowStatusBar
Extension					Sizing

**Components** 

List all added components. Screen1 is already added by default, and you can add multiple screens if needed.

HourOfCode	Screen1 • Add Screen Remove Screen		Designer Blocks
Palette	Viewer	Components	Properties
Palette       User Interface       ■       Button       ✓       CheckBox       ●       DatePicker       ✓       Image       ▲       ListPicker       ■       ListPicker       ●       Notifier       ●       PasswordTextBox       ●       Slider       ●       TextBox       ●       TimePicker       ●       WebViewer	Verver       Image: Display hidden components in Viewer       Image:	Screen1	Properties         Screen1         AboutScreen         Default         AlignHorizontal         Left:1 •         AlignVertical         Top:1 •         AppName         HourOCode         BackgroundColor         Default         BackgroundColor         Default         BackgroundColor         Default •         None         CloseScreenAnimation         Default •         Icon         Icon prg
Layout Media Drawing and Animation Maps Sensors Social Storage Connectivity LEGO® MINDSTORMS® Experimental		Rename Delete Media codi.jpg icon.png Bee-Sound mp3 Upload File	OpenScreenAnimation Default • PrimaryColor Default PrimaryColorDark Default ScreenOrientation Unspecified • Scrollable ShowListsAsJaon ShowStatusBar
Extension			Sizing Fixed •

<u>Media</u> List all uploaded media (eg. sound, images).

Palette	Vie	ewer	Components	Properties
User Interface		Display hidden components in Viewer	Screen1	Screen1
Button	۲	Check to see Preview on Tablet size.		AboutScreen
CheckBox	•	Screen1		
DatePicker	T			AccentColor
Image	T			AlignHorizontal
Label				Left:1 *
ListPicker	(7)			AlignVertical
LietView				Top:1 ·
Malifer				HourOlCode
Notifer	U.			BackgroundColor
Password textbox	œ			Default
Slider				Backgroundimage
Spinner	۲			None
TextBox	T			CloseScreenAnimation
i TimePicker	•			leon
WebViewer	T			icon.prg
			Rename Delete	OpenScreenAnimation
ayout		J D	Madia	Default •
Aedia			meura	PrimaryColor
Drawing and Animation			codi.jpg	Default
Maps			Bee-Sound.mp3	PrimaryColorDark Default
Sensors			Upload File	ScreenOrientation
Social				Unspecified •
Storage				Scrollable
Connectivity				ShowListsAsJson
EGO® MINDSTORMS®				Chana Chathan Dan
Experimental				SnowStatusBar
Extension				Sizing

#### **Properties**

Allow you to change properties for any component (eg. change background color of a button)

### **Creating Our First Interface**



Insert an "Image" and a "Spinner" into the viewer

## Creating Our First Interface

M	ledia		
	photo	o1.jpg	
	photo	o3.jpg	
		Upload File	

Upload 2 or more images through the media window

Properties	
Spinner1	
ElementsFromString photo1.jpg, photo3.jpg	
Width	
Automatic	
Prompt	
Selection	
Visible 🕑	

Select the "Spinner".

Add the image filename into "ElementsFromString" , separating each element with a comma.

# Setting Up (Part 2 of 2)

- On your Android phone, go to the Play Store and search for "mit ai2 companion"
- The app author should be "MIT Center for Mobile Learning"
- Install the app

# **Connecting App Inventor**



On your browser, click "Connect" followed by "AI Companion"



Scan the QR code with your phone or enter the 6 characters code.

Wait for the connection. The screen you have designed will appear upon connection.

### **Block Programming**



#### **Switch to Blocks Editor**

Designer: For designing the user interface

Blocks: For coding program behavior

### **Block Programming**



#### **Block Programming**

HourOfCode	Close App Inventor	Toggle Tutorial	Screen1 -	Add Screen	Remove Screen		Designer Bloc
Blocks	Viewer						
😑 Built-in							~
Control							
Logic							
Math							
Text							
Lists							
Colors							
Variables							
Procedures							
Screen1							
Button 1							
Sound1							
<ul> <li>Any component</li> </ul>							
							-0-
							Ó
Rename Delete							Ð
Media							$\overline{-}$
Bae-Sound mo3	-						_
codi.jpg	0 €	30					
icon.png	Show Warr	nings					
Upload File			_			_	

#### **Viewer**

Area where you drag your blocks to program your app



when	Spinner1 .AfterSelecting
sele	ection
do	
1	

...and drag the above block into your viewer window

#### **Event-driven programming**

Programming for graphical user interface typically uses an event-driven approach. In this approach, the program flow occurs in response to an event. The "when" block above represents an event.

While any programming language can be used with an event-driven approach, most introductory courses in programming uses a procedure-driven approach.



Next, select your image...



...and drag the above block into your viewer window. Put it inside the "when" block like this...



#### **Warnings and Errors**

After the last step, you should see this at the bottom left of your viewer.

The symbol on the left represents a "Warning", and there is currently one warning.

The symbol on the right represents an "Error", and there is currently no errors.



Clicking on "Show Warnings" will cause the warning to be displayed on the blocks.

In this case, the warning occurs because we haven't provided the value for setting the picture (...empty socket on the right).



#### **Warnings and Errors**

#### <u>Error</u>

An unrecoverable problem. The program cannot continue.

#### **Warning**

The program encountered a problem, but can still continue. For example; in the current case, since we didn't provide a filename, the program will assume we are setting it to a blank and continue based on that.

Most programming languages have errors and warnings, and the meaning of these two terms are largely consistent across most programming languages.



Hover your mouse cursor over "selection" and the "get" and "set" blocks will appear.

Pull out the "get selection" block and attach it to the end of the "set image" block.

"selection" is a variable name that represents which spinner option was selected. You'll learn more about variables in a later lesson.

- Try out the app on your phone
- If it doesn't work, try checking...
  - Under "ElementsFromString" are the values the same as the filename in media?
  - App Inventor may change the filename upon upload. Double check that the names match!
  - Try reseting the connection between your browser and your phone

Screen1	off fobility	şu B	9:48
add items 🔻			
Speak			
Ç	$\Box$	Ū	

Add a "TextBox" and a "Button" into your viewer.

Select your button, then from the properties window, change the text to "Speak"



From the component palette, select "Media" then drag a "TextToSpeech" component into the viewer.

The "TextToSpeech" component will appear at the bottom, as it is not a visible component.



What if we want the button to be on the right of the text box? You can't just drag it into position!



Drag a "HorizontalArrangement" component into your viewer.

The "HorizontalArrangement" component doesn't do anything by itself. It is used to contain other component, and any components inside it will be given a horizontal arrangement.

Many graphical interfaces restricts your ability to place a component at an exact position. While this may seem inconvenient, it is because your app is expected to run on many different devices with different screen size and shapes. Components sizes may also change when the app is running. Setting an exact location may look fine on one phone, but not on another.

		😪 🖬 📓 9:48
Screen1		
2		
add items 🔻		
	Speak	
4	$\sim$	_

Drag the text box and the button into the horizontal arrangement box.

Optional: Set the width of the horizontal arrangement box to "Fill Parent".

Optional: Set the width of the text box to "Fill Parent".

What happens if you set "Fill Parent" on the text box, but not on the horizontal arrangement box?



Add in a checkbox and set the text to "Talk fast"

Add a label and change the text to "Press speak to make the app talk"

Labels are not interactive components. Through the block programmer, you can change their text, color, fontsize, etc, but it cannot detect clicks.





...and drag the above block into the viewer.







...drag out an "if" block and add it to the top of the button click event block.



Clicking on the gear will open up an option window. Add an "else" into the if block.



Add the following blocks into the "if" block.

"Checkbox Checked": This sets the condition for the if.

"Set SpeechRate" to 2: The "2" can be found under Math. This sets the action to be done if the condition is true.

"Set SpeechRate" to 0: This sets the action to be done if the condition is not true.

- Try out the app on your phone
- If it doesn't work, try reseting the connection between your browser and your phone

## Challenges

- We haven't used the textbox yet. Make the app speak the text in the textbox.
- Bonus Challenge: Explore the YandexTranslate component (found under media). Can you make the app translate the text to Spanish and speak it? (Tip: Language code for Spanish is "es")



# Mini-Project

- Create a fortune telling app
- The machine should produce a random fortune each time you shake it
- Fortune can be visual, spoken, or both
- Bonus:
  - Add in sound effects
  - Add in a gender selection, and change your fortune text to reflect the user's gender
  - Add in graphical effects