Andhra Pradesh State Skill Development Corporation

Skill AP
APSSDC

eSkill AP
Learn Anytime Anywhere

Andhra Pradesh State Skill Development Corporation

INTERNET OF THINGS (IoT)

MIT APP INVENTOR 2

MIT App Inventor 2

What is mit

MIT-Massachusetts Institute of Technology is a private research university in Cambridge, Massachusetts. The institute is a land-grant, sea-grant, and space-grant university, with an urban campus that extends more than a mile alongside the Charles River.

MIT App Inventor is an intuitive, visual programming environment that allows everyone – even children – to build fully functional apps for smartphones and tablets. Those new to MIT App Inventor can have a simple first app up and running in less than 30 minutes. And what's more, our blocks-based tool facilitates the creation of complex, high-impact apps in significantly less time than traditional programming environments.

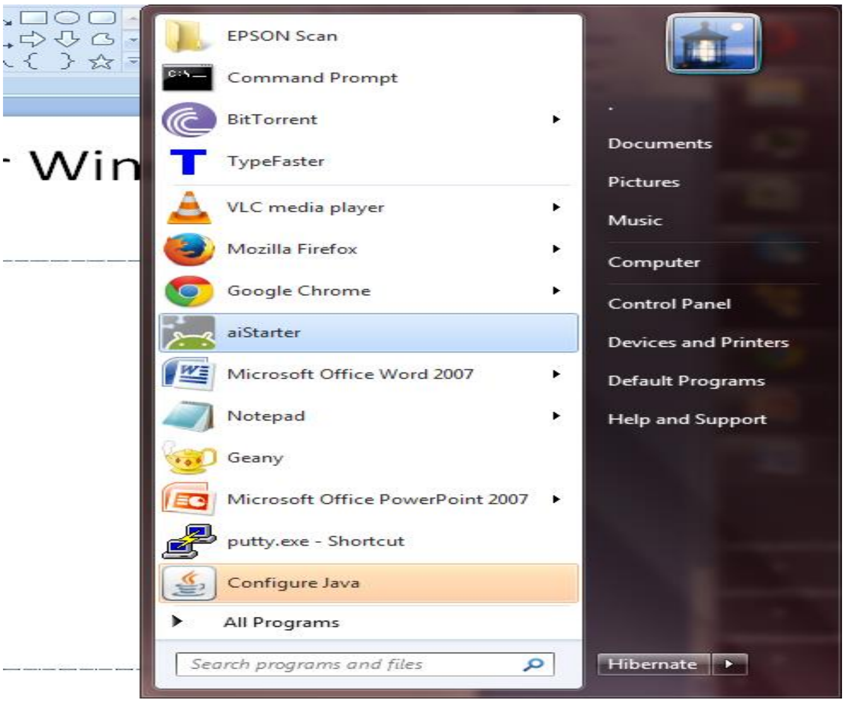
Install Chrome or Firefox

- The App Inventor works only on Chrome and/or Firefox. It does not work on Internet Explorer. Please install any of these web browsers
    - The browsers software is available in the software/browsers folder

Install AppInventorSetup Installer

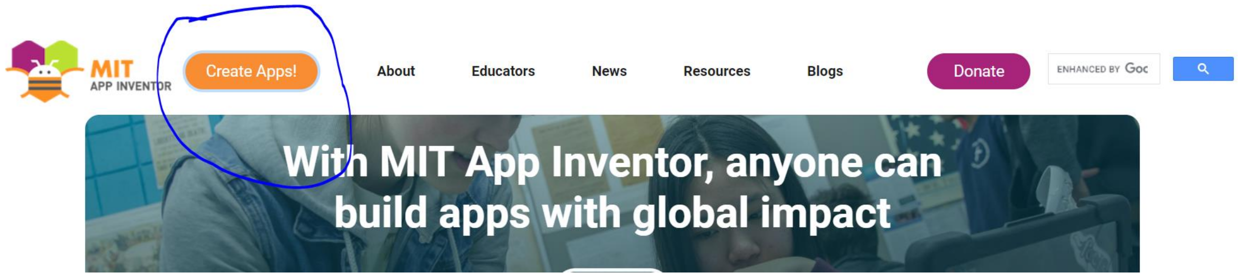
- From the software folder, install the AppInventor_Setup_Installer_v_2_2.exe

Start aiStarter on your Windows System



HOW TO START MIT APP

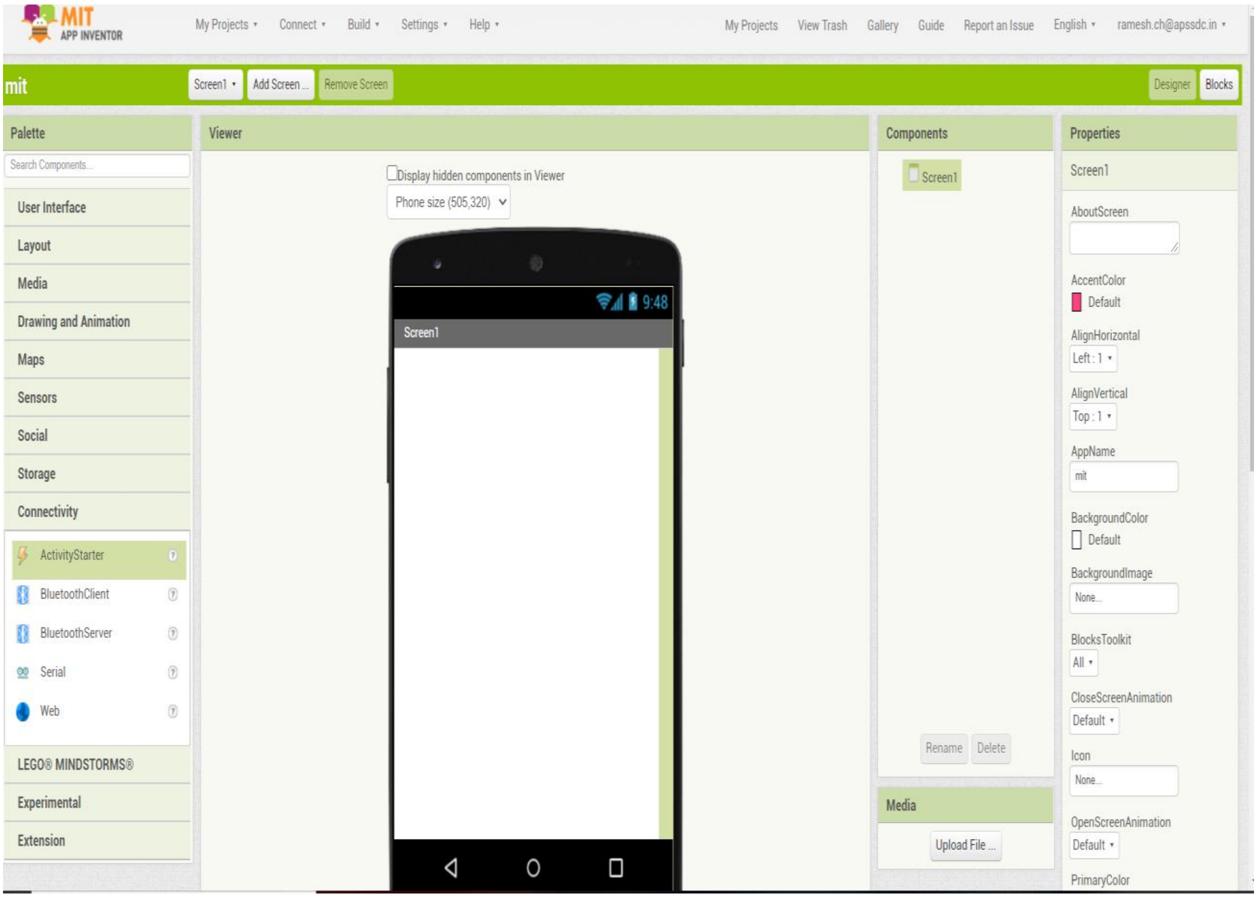
- OPEN chrome type mit app inventor(https://appinventor.mit.edu/)   click create apps



With MIT App Inventor, anyone can build apps with global impact

How start mit in chrome

- Select gmail to access mit app inventor and mit app inventor is open select new project
- In mit you can see 5 types of modules 1.palette 2. viewer 3. components 4. properties 5.media
- In palette different types components
- In Viewer display hidden component in viewer
- In components you can see which component is present in in viewer
- In Properties you can see properties of components is present in viewer
- In Media you can add file ,photos ,song, videos



□ Display hidden components in Viewer

Phone size (505,320)

AboutScreen

AccentColor: Default

AlignHorizontal: Left : 1

AlignVertical: Top : 1

AppName: mit

BackgroundColor: Default

BackgroundImage: None...

BlocksToolkit: All

CloseScreenAnimation: Default

Icon: None...

OpenScreenAnimation: Default

PrimaryColor

Palette

User Interface

- Palette in different types in components first user interface some components they are button check box date piker image etc…
- Button :Button with the ability to detect clicks. Many aspects of its appearance can be changed, as well as whether it is clickable (Enabled). Its properties can be changed in the Designer or in the Blocks Editor.
- Checkbox :A CheckBox component raises an event when the user taps it. There are many properties affecting its appearance that can be set in the Designer or Blocks Editor.
- Date picker:A button that, when clicked on, launches a popup dialog to allow the user to select a date on the Gregorian Calendar.
- Image:displaying images and basic animations.The picture to display, and other aspects of the Image’s appearance, can be specified in the Designer or in the Blocks Editor
- Notifier:The Notifier component displays alert messages and creates Android log entries through an assortment of methods.
- Slider:A Slider is a progress bar that adds a draggable thumb. You can touch the thumb and drag left or right to set the slider thumb position.
- Spinner:A Spinner component that displays a dialog with a list of elements. These elements can be set in the Designer or Blocks Editor by setting the ElementsFromString property to a comma-separated list of values (for example, choice 1, choice 2, choice 3) or by setting the Elements property to a List in the Blocks editor
- TextBox:Users enter text in a text box component.The initial or user-entered text value in a text box component is in the Text property.
- TimePicker:A button that, when clicked on, opens a dialog to allow the user to select a time.
- WebViewer:Component for viewing Web pages.The HomeUrl can be specified in the Designer or in the Blocks Editor. The view can be set to follow links when they are tapped, and users can fill in Web forms.

Layout

- HorizontalArrangement:Use a horizontal arrangement component to display a group of components laid out from left to right.
- HorizontalScrollArrangement:A formatting element in which to place components that should be displayed from left to right. If you wish to have components displayed one over another, use VerticalScrollArrangement instead.
- VerticalArrangement:Use a VerticalArrangement component to display a group of components laid out from top to bottom, left-aligned.
- VerticalScrollArrangement:A formatting element in which to place components that should be displayed one below another.

Media

- Camcorder:A component to record a video using the device’s camcorder
- Camera:Use a camera component to take a picture on the phone.
- ImagePicker:A special-purpose button. When the user taps an ImagePicker, the device’s image gallery appears, and the user can choose an image.
- Player:Multimedia component that plays audio and controls phone vibration
- Sound:A multimedia component that plays sound files and optionally vibrates for the number of milliseconds (thousandths of a second) specified in the Blocks Editor
- SoundRecorder:Multimedia component that records audio.
- SpeechRecognizer:Use a SpeechRecognizer component to listen to the user speaking and convert the spoken sound into text using the device’s speech recognition feature.

Drawing and Animation

- Ball:A round ‘sprite’ that can be placed on a Canvas, where it can react to touches and drags, interact with other sprites (ImageSprites and other Balls) and the edge of the Canvas, and move according to its property values.
- Canvas:A two-dimensional touch-sensitive rectangular panel on which drawing can be done and sprites can be moved.
- ImageSprite:A ‘sprite’ that can be placed on a Canvas, where it can react to touches and drags, interact with other sprites (Balls and other ImageSprites) and the edge of the Canvas, and move according to its property values. Its appearance is that of the image specified in its Picture property (unless its Visible property is false.

Sensors

- AccelerometerSensor:Non-visible component that can detect shaking and measure acceleration approximately in three dimensions using SI units (m/s2).
- Clock:Non-visible component that provides the instant in time using the internal clock on the phone. It can fire a timer at regularly set intervals and perform time calculations, manipulations, and conversions.
- LocationSensor:Non-visible component providing location information, including Latitude, Longitude, Altitude (if supported by the device), speed (if supported by the device), and address.
- OrientationSensor:Use an orientation sensor component to determine the phone’s spatial orientation.An orientation sensor is a non-visible component
- Pedometer:This component keeps count of steps using the accelerometer.
- ProximitySensor:A sensor component that can measure the proximity of an object (in cm) relative to the view screen of a device

Social

- ContactPicker:A button that, when clicked on, displays a list of the contacts to choose among. After the user has made a selection, the following properties will be set to information about the chosen contact
- EmailPicker:An EmailPicker is a kind of text box. If the user begins entering the name or email address of a contact, the phone will show a dropdown menu of choices that complete the entry.
- PhoneCall :A non-visible component that makes a phone call to the number specified in the PhoneNumber
- PhoneNumberPicker:A button that, when clicked on, displays a list of the contacts’ phone numbers to choose among
- Sharing:Sharing is a non-visible component that enables sharing files and/or messages between your app and other apps installed on a device
- Texting:A component that will, when the SendMessage method is called, launch the device’s preferred texting app to send the text message specified in the SendMessage property to the phone number specified in the PhoneNumber property.

Storage

- CloudDB:The CloudDB component is a Non-visible component that allows you to store data on a Internet connected database server (using Redis software). This allows the users of your App to share data with each other
- File:Non-visible component for storing and retrieving files. Use this component to write or read files on the device. The default behavior is to write files to the private data directory associated with the app.
- TinyWebDB:The TinyWebDB component communicates with a Web service to store and retrieve information. Although this component is usable, it is very limited and meant primarily as a demonstration for people who would like to create their own components that talk to the Web
- TinyDB: non-visible component that stores data for an app.Apps created with App Inventor are initialized each time they run. This means that if an app sets the value of a variable and the user then quits the app, the value of that variable will not be remembered the next time the app is run

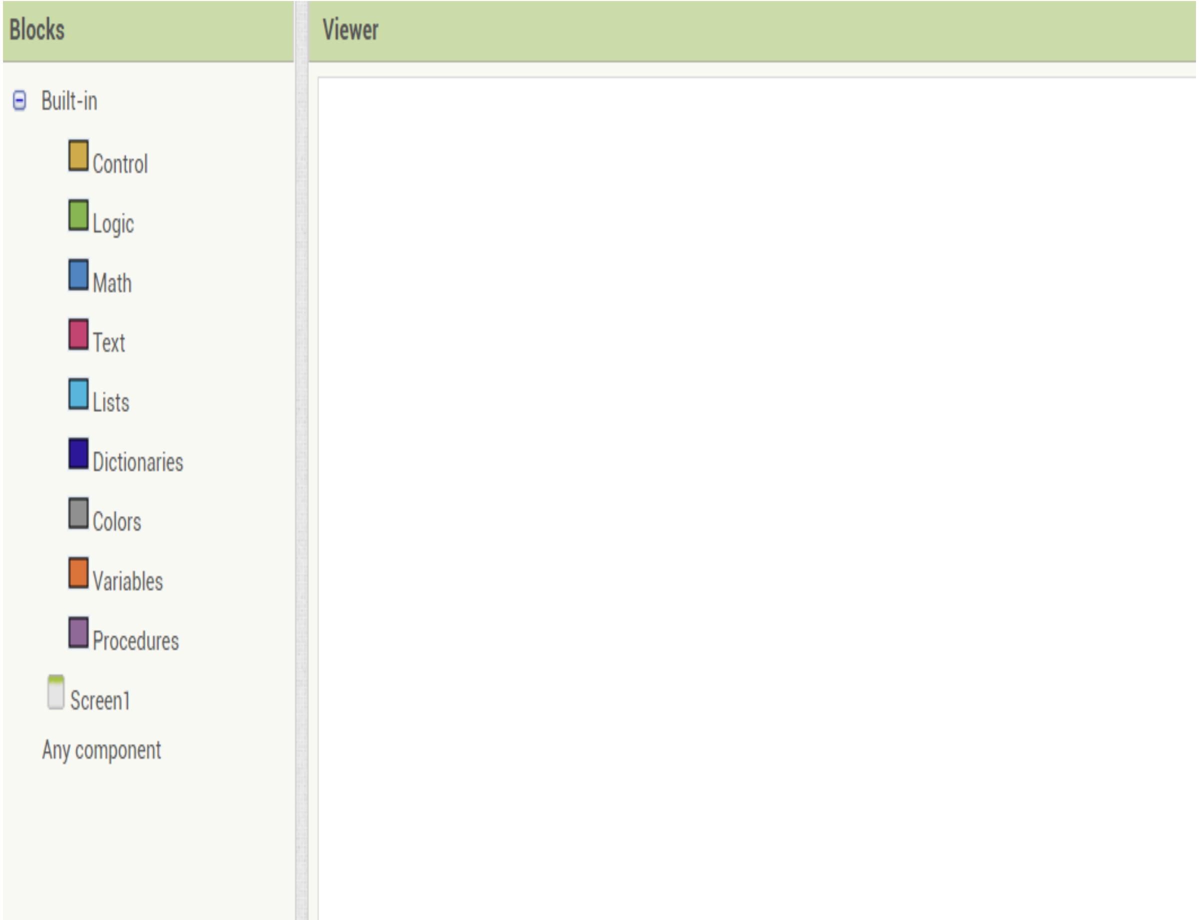
Connectivity

- ActivityStarter:A component that can launch an activity using the StartActivity method.
- Activities that can be launched include:
- Starting another App Inventor for Android app. Starting the camera application.Performing web search.

- BluetoothClient:Use BluetoothClient to connect your device to other devices using Bluetooth. This component uses the Serial Port Profile (SPP) for communication
- BluetoothServer:Use the BluetoothServer component to turn your device into a server that receive connections from other apps that use the BluetoothClient component.
- Web:Non-visible component that provides functions for HTTP GET, POST, PUT, and DELETE requests.

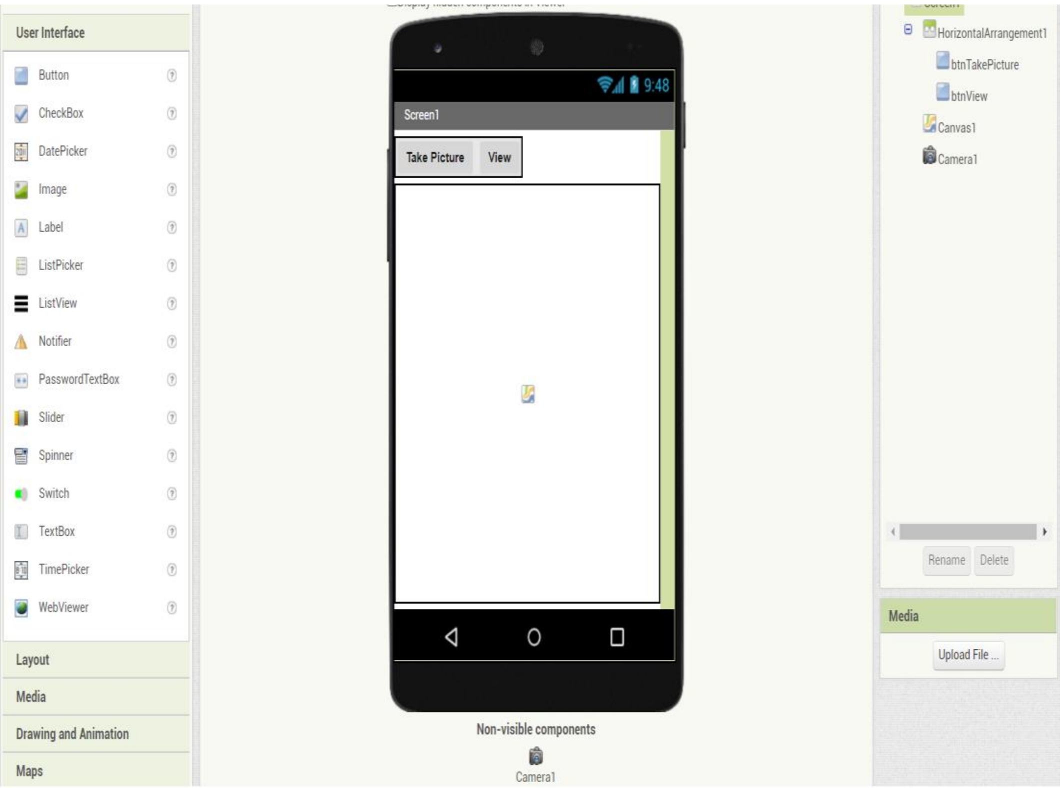
Blocks

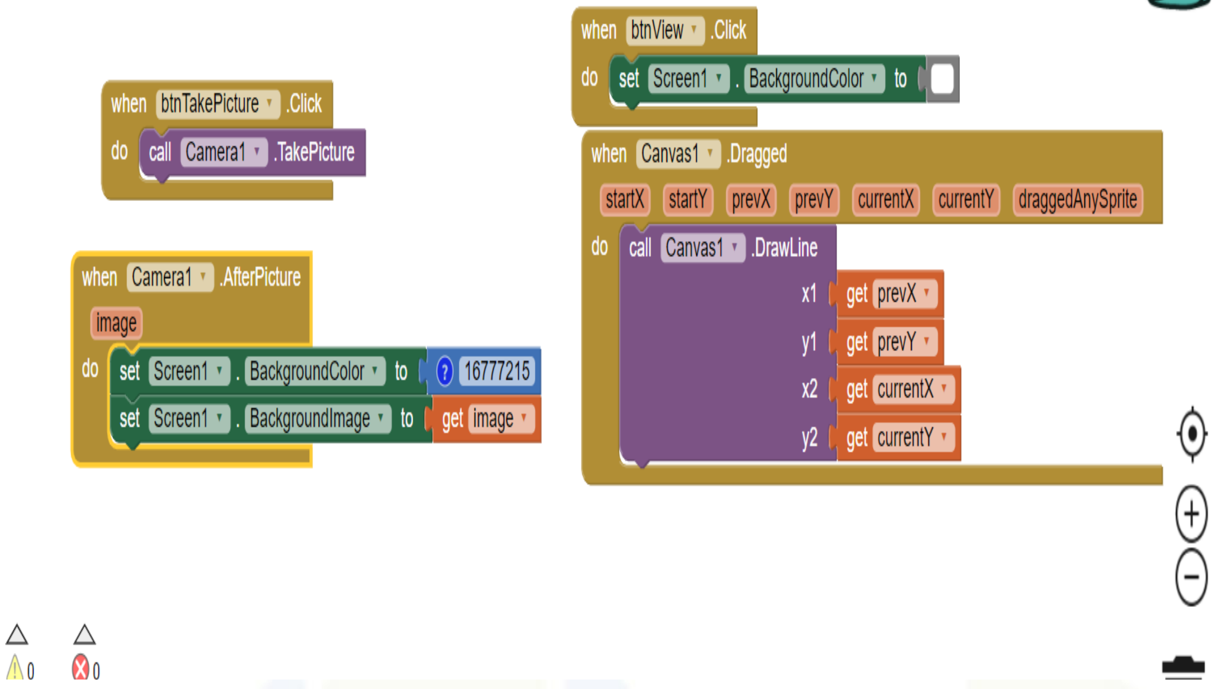
- In blocks some built in functions is present in mit
- Control functions:in control functions in shown loop functions, condition functions and next screen operations
- Login functions:login operations it means and or equal operations
- Math functions:math operations is present in function it means +,-,*
- Text functions: you can display any text in output you can write in blocks
- List functions: you add any elements in blocks also you add list function
- Colors functions:you add any colors to screen or output functions
- Variables functions: Variables are the names you give to computer memory locations which are used to store values in a computer program
- Procedures functions:you add any functions in bloks part you can used this function it means it work like function in prog part



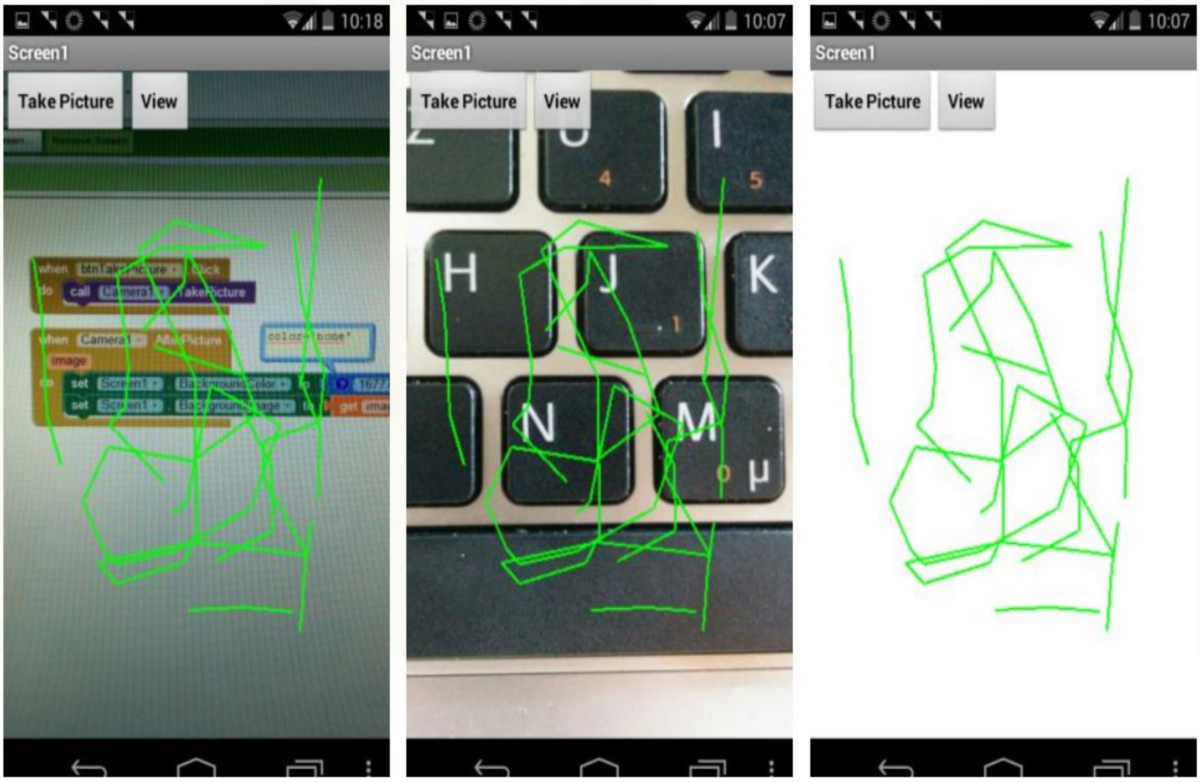
How to change background image without changing drawn lines

- I'm creating an app that will allow me to take a picture and then draw on it. The next step would be to then be able to view the drawn part without the background image.
- How would I do this?
  The only possibility I can see is to set the taken image as screen background, then draw on the canvas. I set the Canvas background color to none in the Designer window to be able to see the background image.
- The Screen background color will be set to none after taking the first image in the Camera.AfterPicture event.



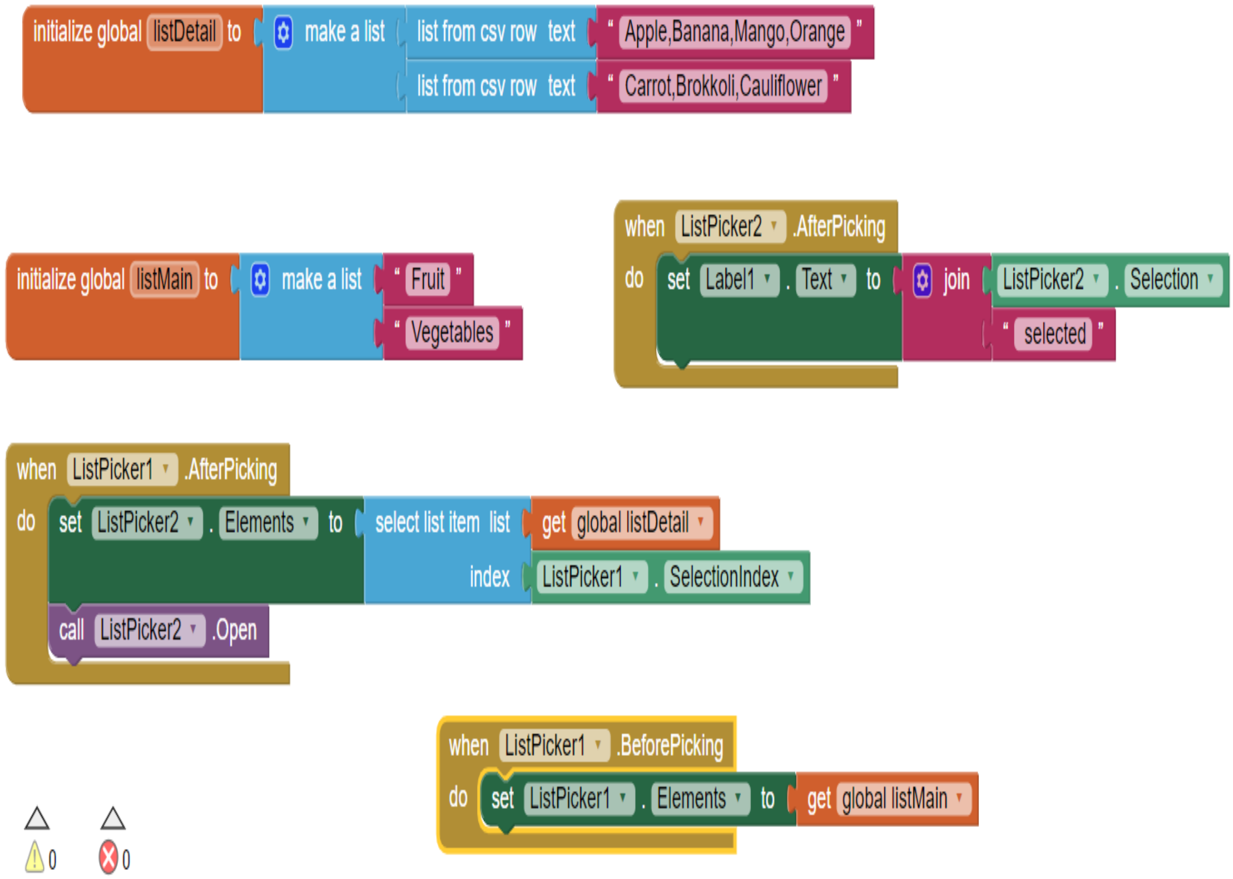


```
when btnTakePicture .Click
do   call Camera1 .TakePicture

when Camera1 .AfterPicture
  image
do   set Screen1 . BackgroundColor to 16777215
     set Screen1 . BackgroundImage to get image

when btnView .Click
do   set Screen1 . BackgroundColor to

when Canvas1 .Dragged
  startX  startY  prevX  prevY  currentX  currentY  draggedAnySprite
do   call Canvas1 .DrawLine
                   x1  get prevX
                   y1  get prevY
                   x2  get currentX
                   y2  get currentY
```



How to create a nested listpicker

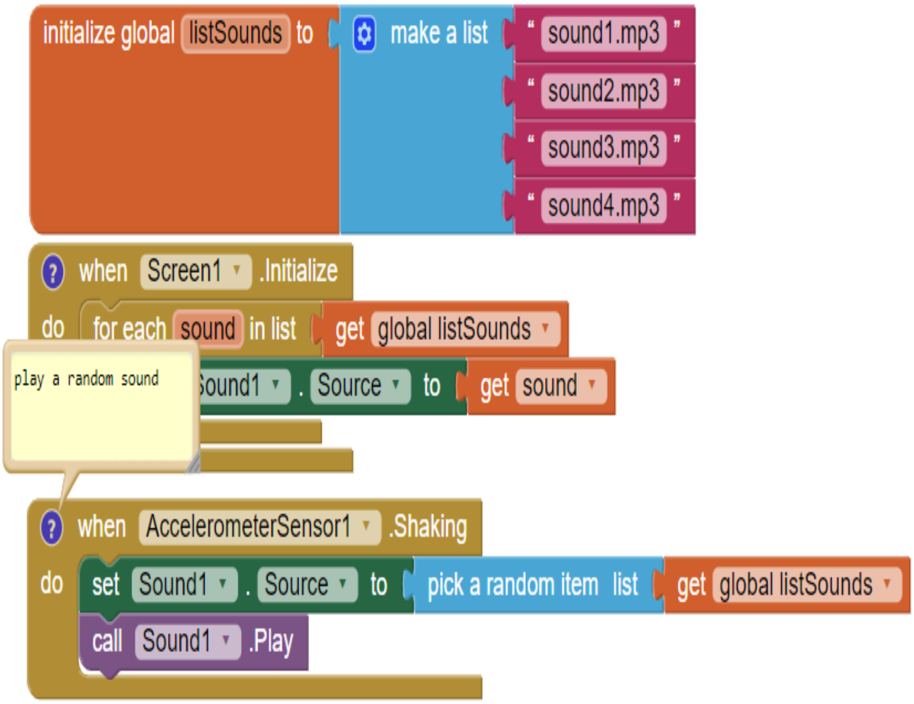
- I need a main listpicker with different sublists of items below them depending which main list item is selected.
- For the solution presented here, I'm using 2 listpicker, the second listpicker for the sublists is hidden and will be opened with the listpicker.open block.



```
initialize global listDetail to make a list
    list from csv row text "Apple,Banana,Mango,Orange"
    list from csv row text "Carrot,Brokkoli,Cauliflower"

initialize global listMain to make a list
    "Fruit"
    "Vegetables"

when ListPicker2 .AfterPicking
do  set Label1 . Text to join
        ListPicker2 . Selection
        " selected"

when ListPicker1 .AfterPicking
do  set ListPicker2 . Elements to select list item list get global listDetail
                                               index ListPicker1 . SelectionIndex
    call ListPicker2 .Open

when ListPicker1 .BeforePicking
do  set ListPicker1 . Elements to get global listMain
```

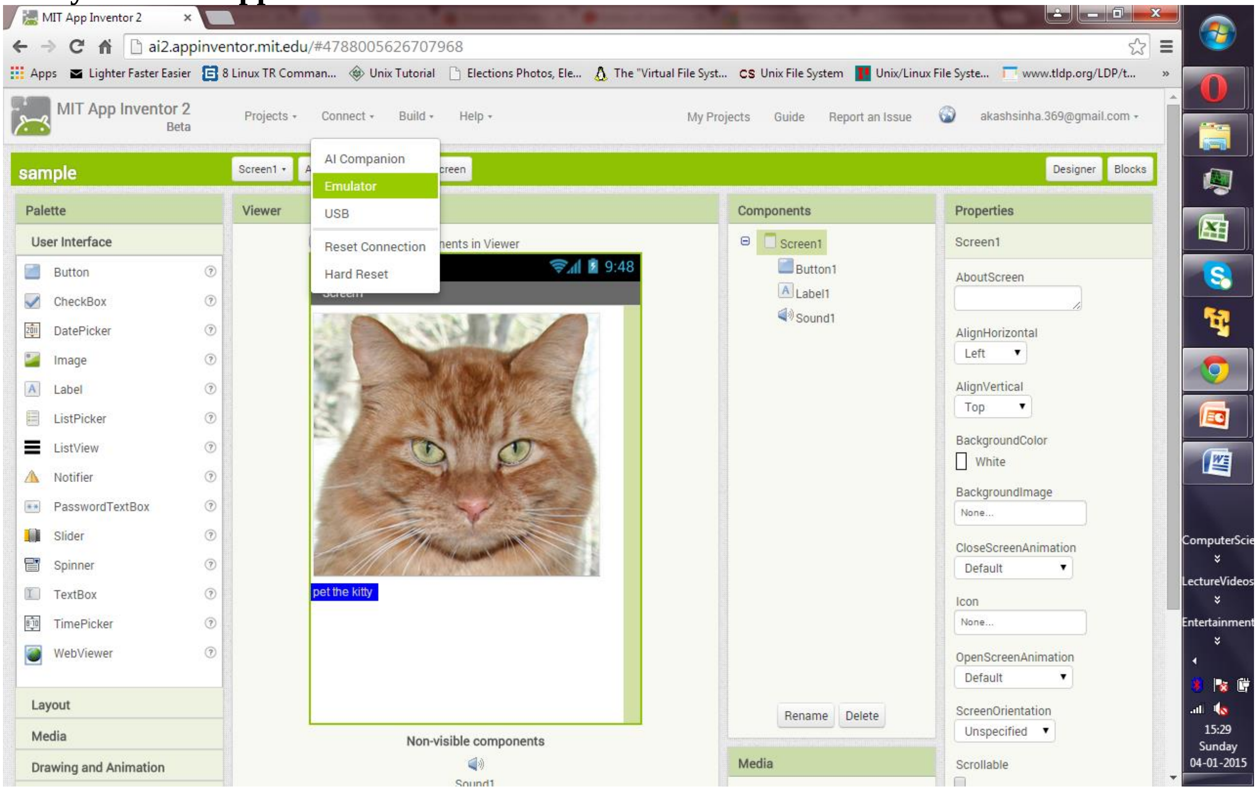
How to play a random sound when the device is shaken

- In mit if mobile is shaken some sound is paly randomDesigner
- first we can take 3 0r 4 types of sounds in chrome
- You can select sound , ActivityStarter in Designer part



```
initialize global listSounds to  make a list  " sound1.mp3 "
                                              " sound2.mp3 "
                                              " sound3.mp3 "
                                              " sound4.mp3 "

when Screen1 .Initialize
do  for each sound in list  get global listSounds
        ound1 . Source to  get sound

play a random sound

when AccelerometerSensor1 .Shaking
do  set Sound1 . Source to  pick a random item list  get global listSounds
    call Sound1 .Play
```

Test your First App in the Emulator



MIT App Inventor 2

ai2.appinventor.mit.edu/#4788005626707968

Properties

Screen1

AboutScreen

AlignHorizontal: Left

AlignVertical: Top

BackgroundColor: □ White

BackgroundImage: None...

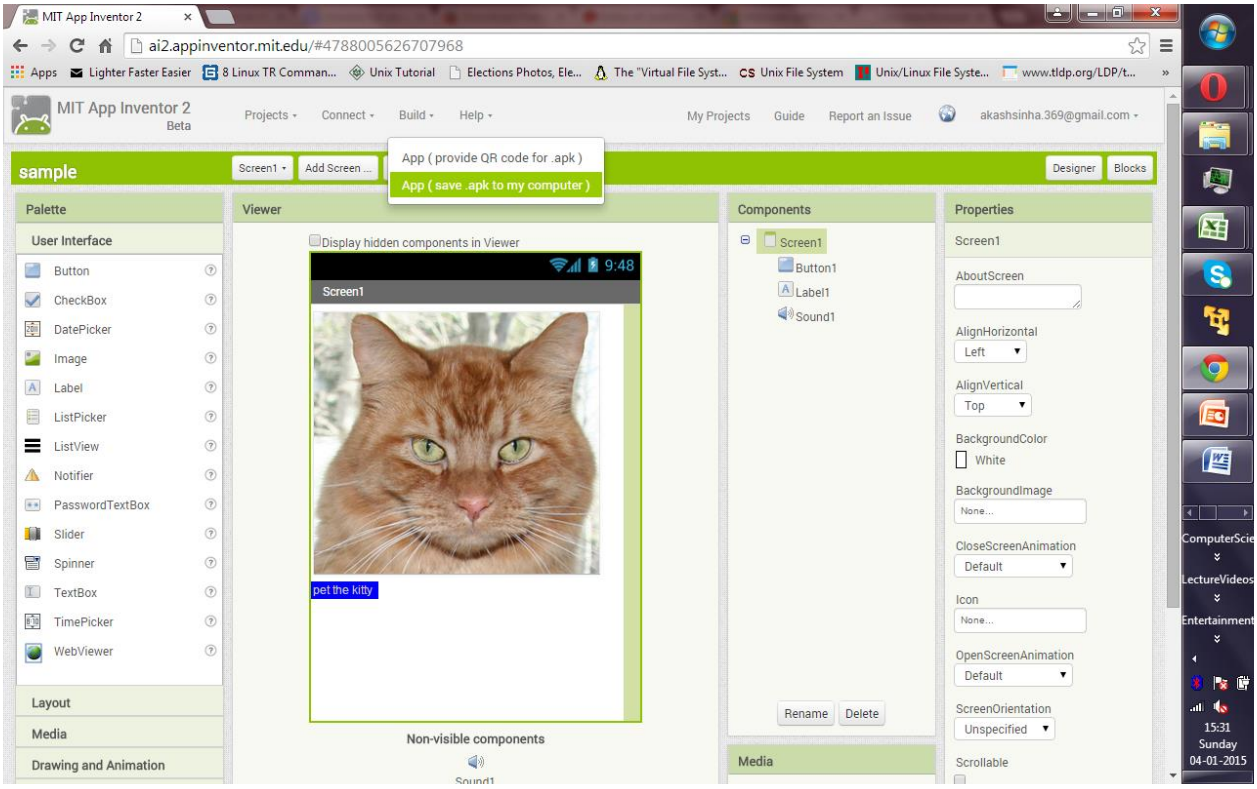
CloseScreenAnimation: Default

Icon: None...

OpenScreenAnimation: Default

ScreenOrientation: Unspecified

Scrollable: □

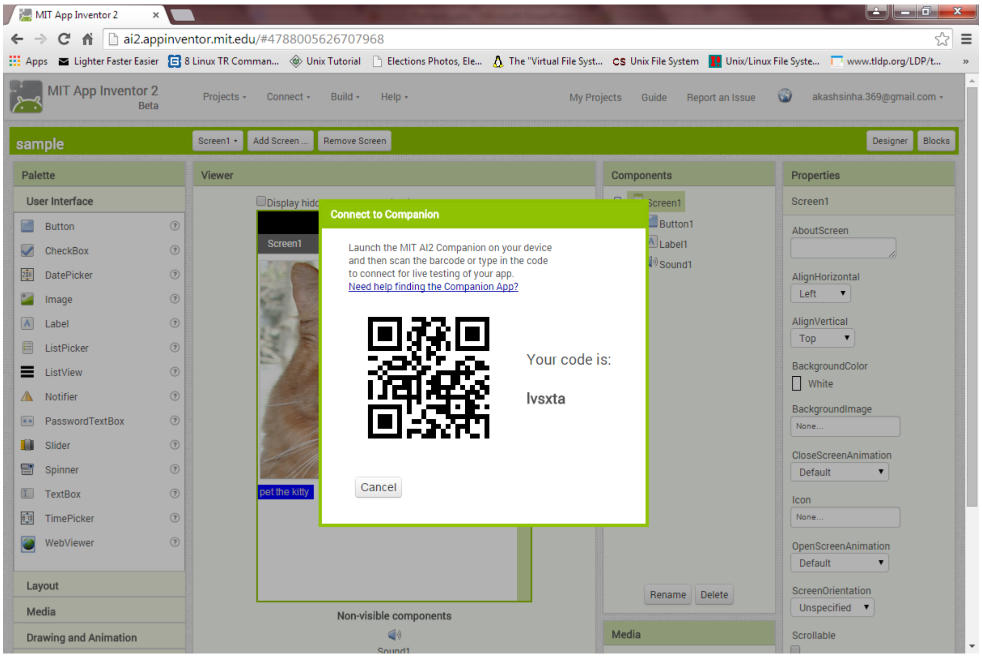


Install .apk file on your android mobile

- Copy the .apk file into your phone / SD card memory
- Use MyFiles app on phone and access the .apk file
- Double tap on the .apk file to install
- Access your installed application

You may use AI2Companion App to test your app

- Install MIT AI2Companion App on your android device from Google Playstore
- Obtain the 6 character code for your app
- Type in that code in AI2Companion to access your app



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:

lvsxta

AlignHorizontal: Left
AlignVertical: Top
BackgroundColor: White
BackgroundImage: None...
CloseScreenAnimation: Default
Icon: None...
OpenScreenAnimation: Default
ScreenOrientation: Unspecified