Overview:
We will be making an app that will mark different locations on the MIT campus. The app will change the screen when the phone is shaken to display a list that represents your schedule. The app will alternate between two different screens. It will look similar to this:

1. Making a new project and using App Inventor:
   b. On the upper left side you will see the word "Projects." Press on it and then choose new project as in the photo below.
   c. Choose a name for your project. (CampusMap for example)
   d. (optional) [Connect your Android phone/tablet](#) to your project to test while you build
First, Designing The User Interface:

The Map Component:

1. On the left side of the new project you should be able to see a Palette which consists of all the different component that we can use in app inventor. Scroll down to Maps and drag the map component into the screen.

2. Press on Map in the components window to your right. You will now be able to see the properties of the map component you just added.
   a. Make sure to make CenterFromString something close to 42.3590169942628, -71.081772806192. Alternatively, you can move the map view in your designer screen where you want it to be centered and then click the crosshair (middle button) in the bottom left. This updates the CenterFromString and ZoomLevel properties to reflect the map state.

b. Set the Height and Width to “Fill parent”

The Markers:
1. In the Map section of the Palette to your left you will see a marker component. Drag a couple of them and place them wherever you want on the map. I’ll be putting mine on Legal Seafood, Kendall Square, and Harvard Bridge. Add an additional marker on the Sandberg Center.

2. Each of the markers has its own properties. Try changing the FillColor of each of them and write some kind of description in the Description box. For the Description to be useful, you will also need to check the EnableInfobox property. For example, Kendall Square: The place where everyone takes photos.

3. You Should have something that look like this:

The List:

In the User Interface section of the Palette to your left drag the list view component to the screen.

a. Set the Background Color to “White”.

b. Set the height to “Fill Parent”.

c. Remove the checkmark from visible.

Second, Designing The Blocks: Here we will be programming and defining the logic of our App.

The Accelerometer Sensor: We will use the Accelerometer sensor in your phone to know if you’ve shaken the phone or not.
1. From the sensors section in the Palette to your left, drag the Accelerometer Sensor icon to the Screen.
2. Your screen should look similar to this:

The Blocks: Now that we have all the necessary components on our screen, we can start to program them using the blocks
1. Press on the Blocks button on the right upper corner.
2. Initializing the list:
   a. Press on Screen in the blocks menu to choose from the blocks that controls the screen functions. Look for “when screen1. Initialize” and drag it to the whiteboard.
b. Now go to the ListView Component in the Blocks menu and drag the “Set ListView1.Elements to” block and insert it inside the Initialize screen block. It should look like this:

![Diagram of ListView block](image)

c. From the Blocks menu choose the Lists component and drag the “make a list” block and connect it to the “set ListView1.Elements to” block. You can increase the number of elements in the list by pressing on the blue square in the block and dragging more items into it, like this:

![Diagram of make a list block](image)

d. From the Text component in the Blocks menu drag empty text blocks and connect them to the list. Fill the text blocks with your schedule.
3. Using the Accelerometer:
   a. From the AccelerometerSensor1 section in the blocks menu drag this block:

   ![Accelerometer Sensor Block]

   We will be controlling the visibility of the map and list components using this block. When the phone is shaked the visibility of them will swap. In other word, the visible component will become invisible and vice versa.

   b. From the map section in the blocks drag the “set Map1.Visible to” and “Map1.Visible” blocks.

   ![Map Visibility Blocks]

   c. We want to swap the visibility of the map so we will need a “not” block from the logic components and we will connect all of these in way that reads: set Map1.Visible to not map1.Visible. This means that if Map1.Visible is true then it will become false and if it’s false, it will become true. Your Blocks should look like this:
d. Get the same two items for the ListView1 component from its section in the blocks menu to your left and do the same thing as before with the map. This is how everything should look like in the end:

Finally Download the App Companion to your phone and from connect in the upper side of the screen select AI Companion and scan the Qr code to use your app!