

LOCATION SENSOR APPLICATION

Introduction

This application will give the GPS coordinates and (some of the times) address of where you are located in the world.

There will be one button and three labels that you will be able to see. And the two hidden components will be **LocationSensor** and **ActivityStarter**.

Location sensor: It is a feature that communicates with the GPS of the phone. It can be used in an application by a user to learn lot of things about current location. For a complete list of location sensor's functionality, you can go to: <http://appinventor.googlelabs.com/learn/reference/components/sensors.html#LocationSensor>

Activity starter: This tutorial will use the concept of an "activity".

An Activity is very important in Android programming. Activity is an isolated thing that a user can interact with. For example, if you are making a game and present the user with a list of levels that he could play – that screen can be one activity. As soon as user selects a level, the next screen he might encounter could be another activity.

In App Inventor, there is a feature called ActivityStarter. It starts another activity - one that you have created already or an activity of an application that came with the phone. In this case, we will be launching the in-built Google Maps, passing in the proper parameters to find your location.

Design Editor Instructions:

Drag a button on the screen and name it Button1. This is the button you will press to start Google Maps. On the right side of the screen in the "Components" section, you can see the entity of the button. Click on it, and in the "Properties" section, set the text of the button to be "Find current location".

Drag three labels on to the screen, and set their texts to be "Longitude", "Latitude", and "Current Address" respectively.

Now go in the "Sensors" section and drag out the "Location sensor" onto the screen. This is an invisible element and will appear at the bottom of the screen.

From “other stuff”, drag Activity Starter onto the screen. This too is an invisible component and will appear below “Location Sensor”.

Click on “Location Sensor” and make sure it is enabled. Do the same with “Activity Starter”.

Name them “LocationSensor1” and “ActivityStarter1” respectively.

Location sensor constantly receives updates from the GPS and has information inside it such as current altitude, longitude, latitude, address, accuracy, etc.

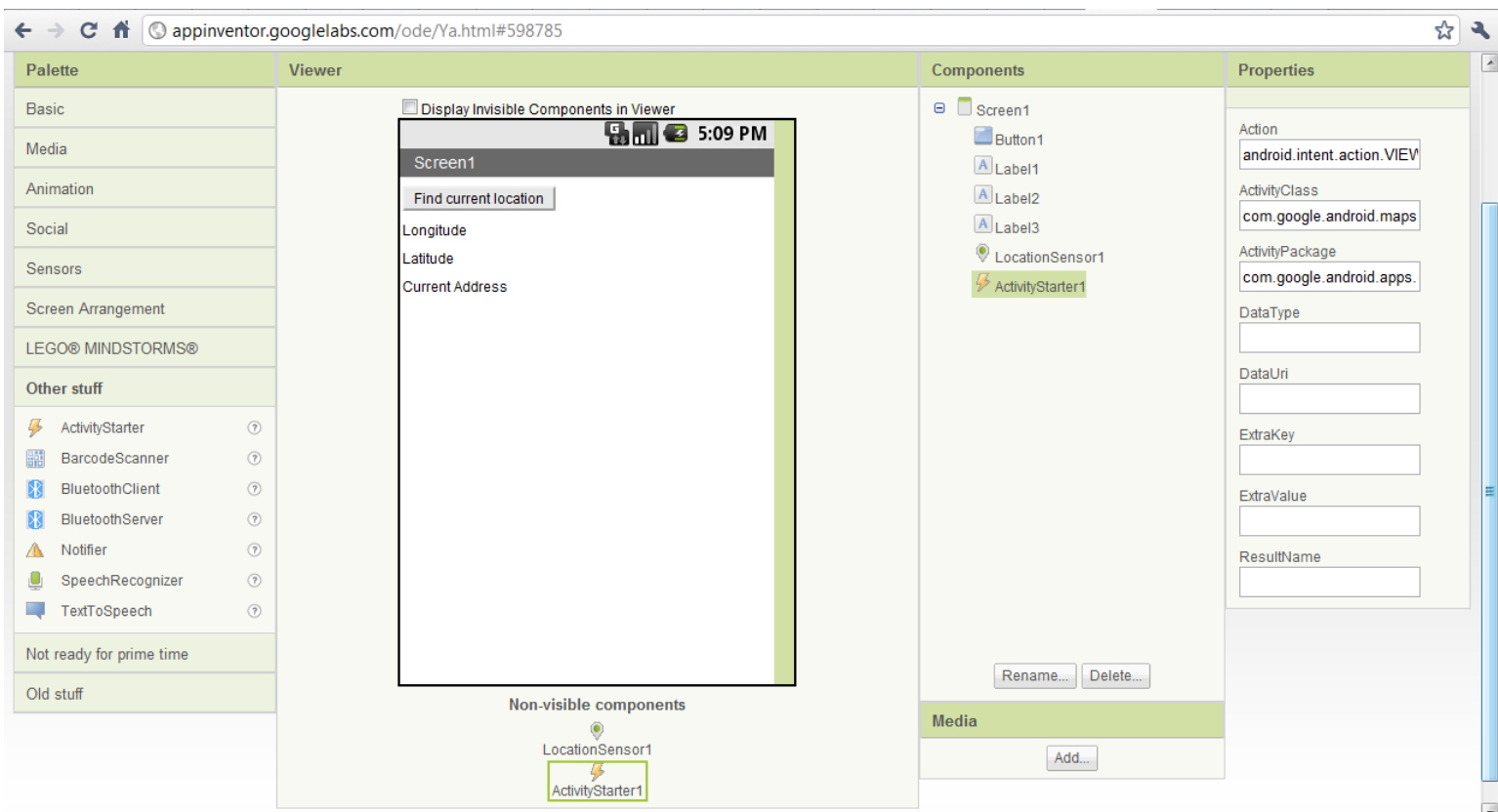
When you check “enabled” for Location Sensor, it becomes active and can start receiving data from GPS in the background. The way in which this transmission of data happens is part of the location sensor’s built-in functionality. We do not need to do anything to facilitate this communication.

For the “Activity Starter”, in the “Properties” section, set the

- “Action” field to “android.intent.action.VIEW”
- “Activity Class” field to “com.google.android.maps.MapActivity”, and
- “ActivityPackage” field to “com.google.android.apps.maps”

Here, we just specified which activity we want to start and which package the implementation of that activity is located in.

Your screen should look something like below now:



Block Editor Instructions:

Every time Location Sensor receives information from GPS, we can use its method “Location Changed” to capture that event.

From “My Blocks” in LocationSensor1, drag out “when LocationChanged”. Now we can update what we show in our labels by setting them to the current latitude, longitude and address.

For each label, click on it, and drag out “set Label.Text to” – and set the text to proper attribute of Location Sensor.

For example if your “Label1” was going to display the latitude, then you would drag “set Label1.Text to” and fit in attribute “LocationSensor1.Latitude”.

We put all of these building blocks into “when LocationChanged”.

Now we move to actually launching Google Maps when user clicks “Find current location” button.

Before the launch the Google Maps activity, we need to pass some information to it as to the location we want to see on the map.

Go to your button block and pull out “when Button.Click”. Then inside this block, we will set up the Activity Starter properly and launch it.

The Google Maps can take its parameters as a string formatted the following way (other formats are also possible):

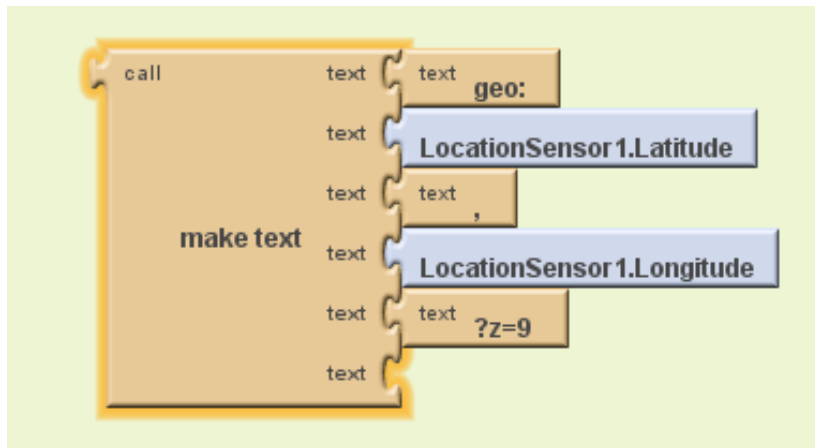
geo: latitude, longitude,

where the words “latitude” and “longitude” are to be replaced with strings that represent real coordinates.

Go to the ActivityStarter1 block and pull out “set ActivityStarter1.setDataUri to” . As argument to it you have to pass a text. First you have to make a complete string of the format described above.

Go to “Built-in”section, click on “Text”, and pull out “call make text”. This method will take numerous text arguments and combine them into one text.

To create one textual string, do something like below where you will combine static text blocks like “geo” and “,” with dynamic values from your Location Sensor.



Now plug this text into “set ActivityStarter1.setDataUri to” block and then drag out the block from ActivityStarter component labeled “ActivityStarter1.StartActivity”.

Your overall block diagram should look like this:

