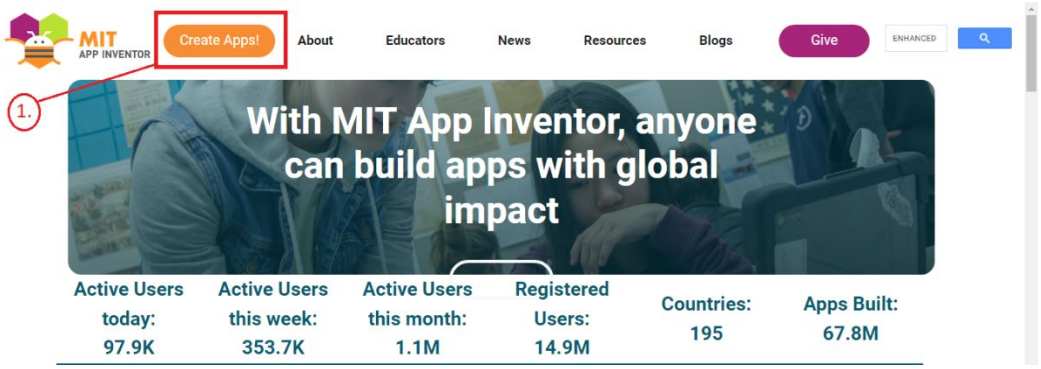
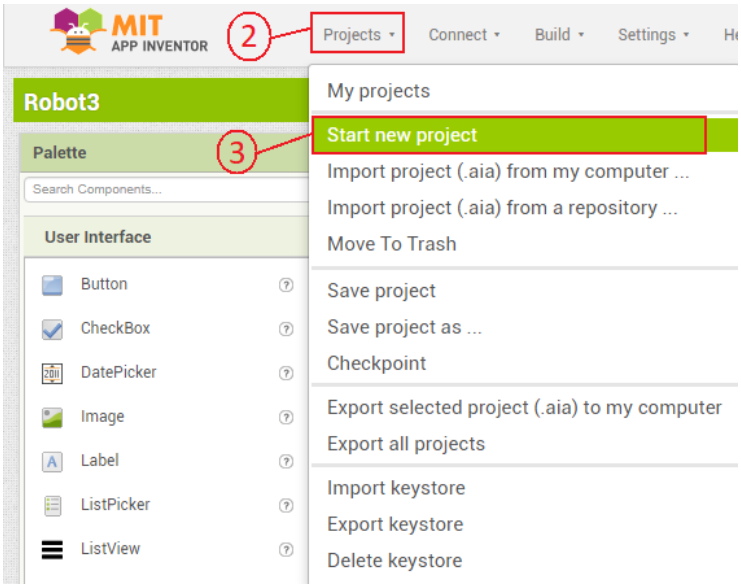
This paper shows the steps needed to import application program from existing (.aia) file.

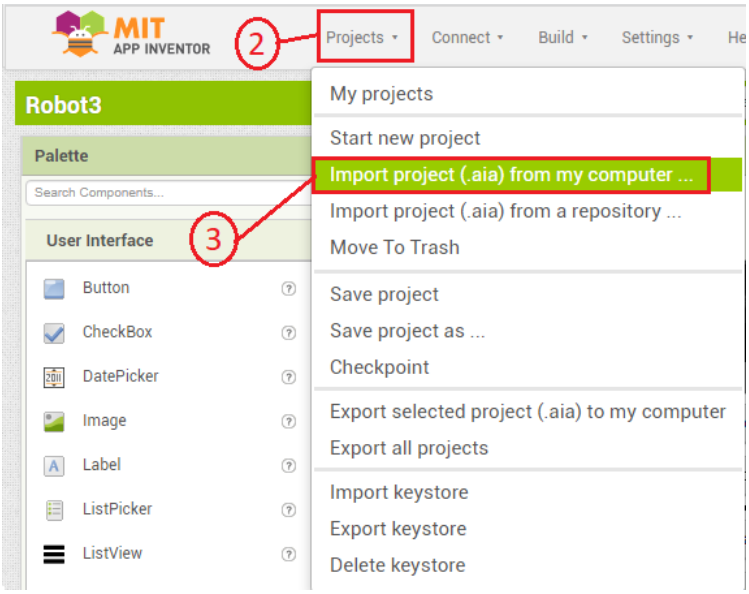
1. Use the following link to access the MIT app inventor web page.
   http://appinventor.mit.edu/
2. Click on “Create Apps!”, and sign in using your email.



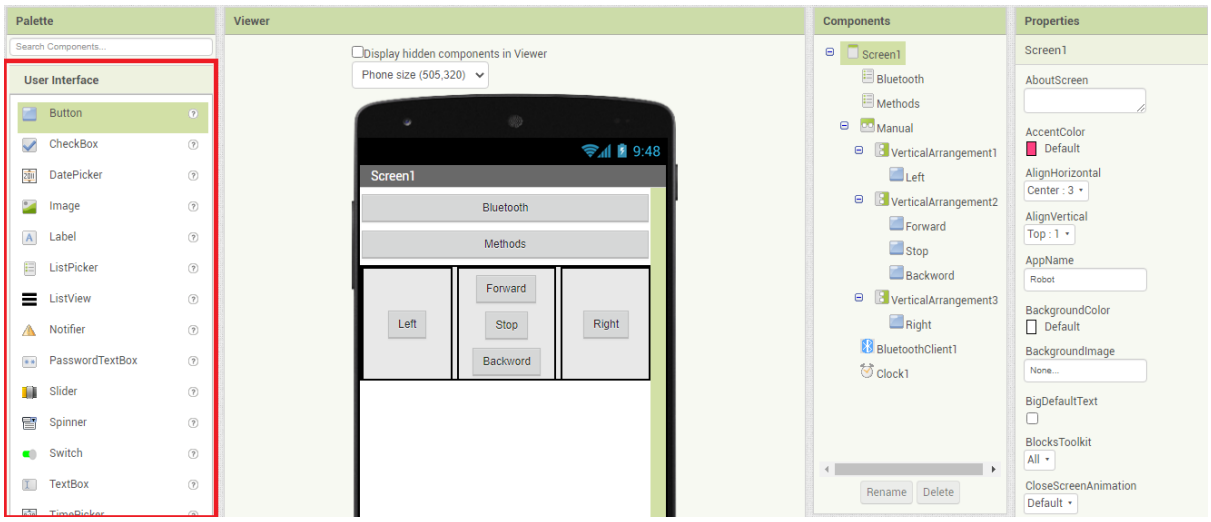
3. From “Projects” list chose “Start New Project”, to start new project



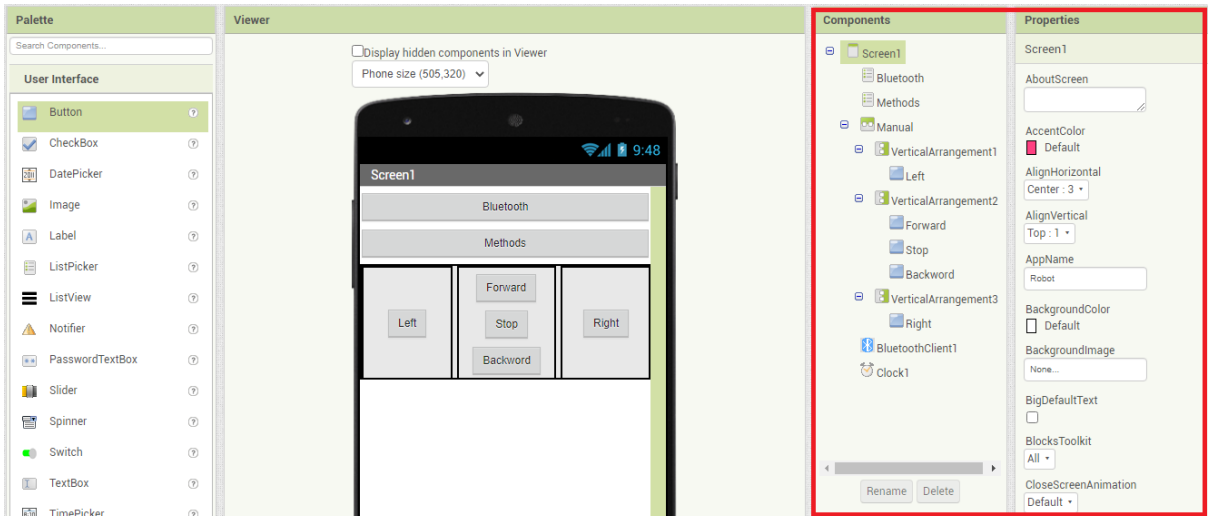
4. Or from “Projects” list chose “Import project (.aia) from your computer…”, to view and modify old program.



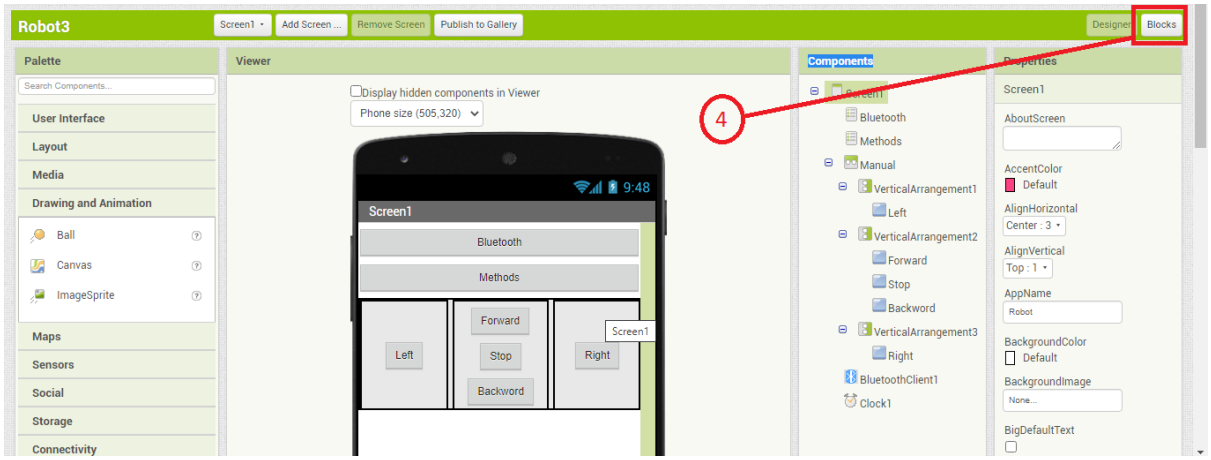
5. On your left hand side, you can add “User interface Components” (ex. Buttons, lists, etc..).



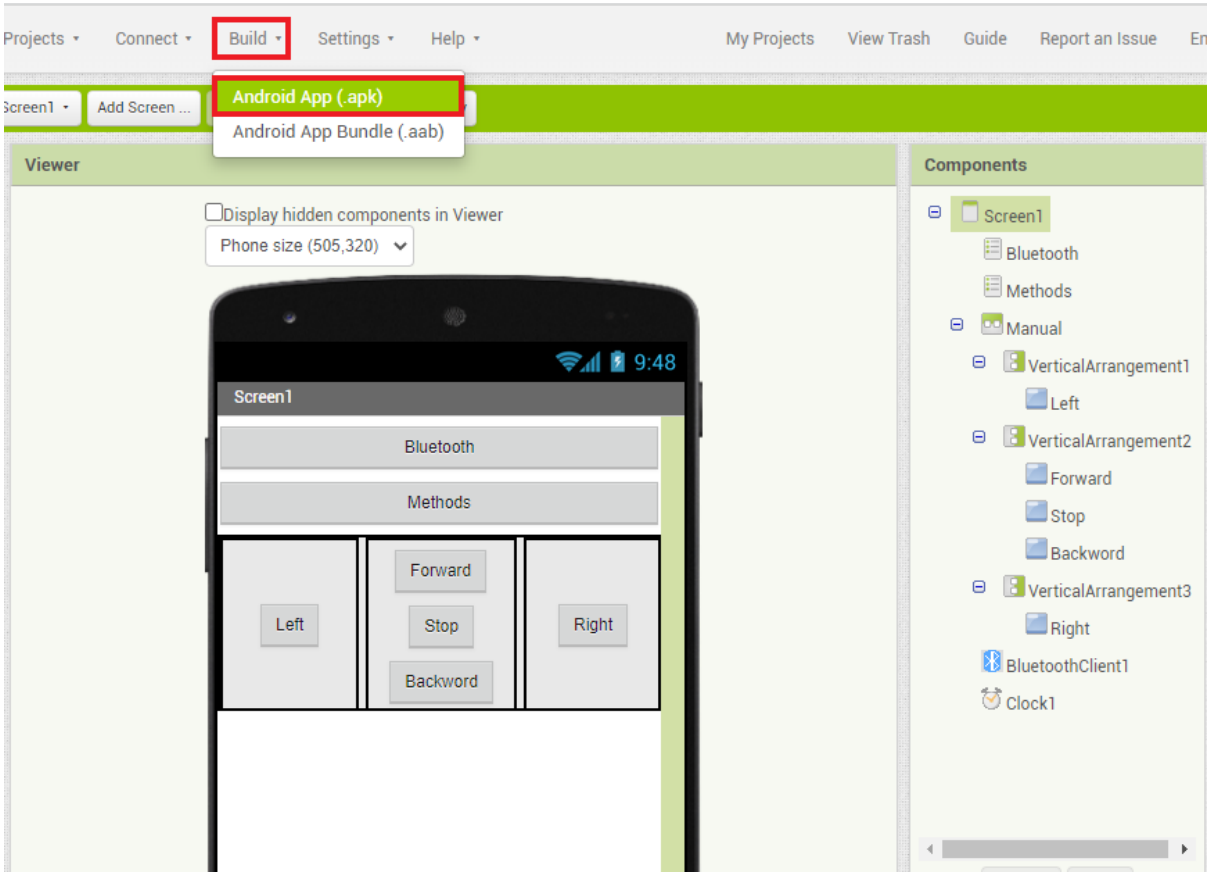
6. It is good to use “Layout” on the left side to organize them
7. On your right side you can view and modify properties of your Components.



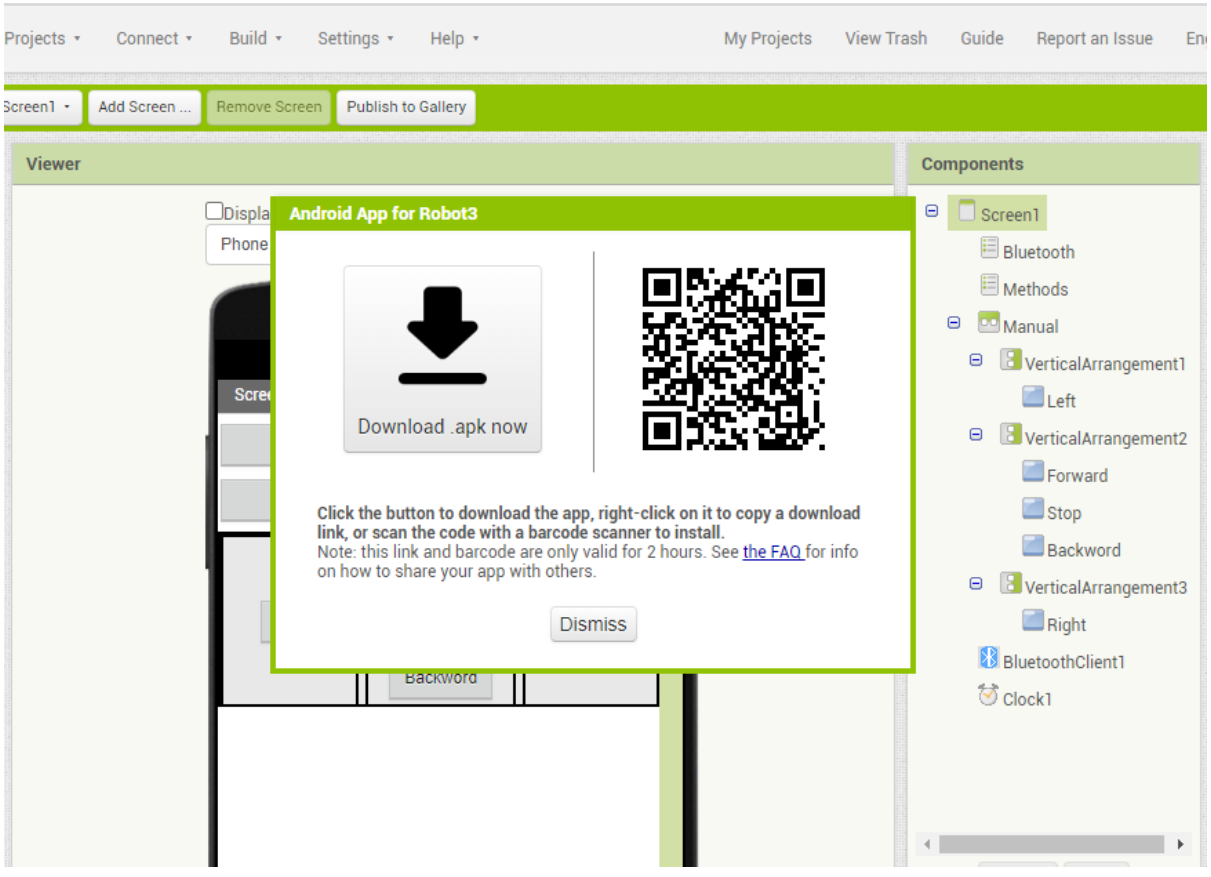
8. To start programing, click on “Blocks” on the upper right corner.



9. After importing the program (and modifying it if needed), from “Build” list choose “Android App (.apk)”



10. Download the app to your device (by scanning the code, or directly downloading it).



In the following part, explanation of the code is provided as captions for the pictures.

This is for the file "Robooot.aia".

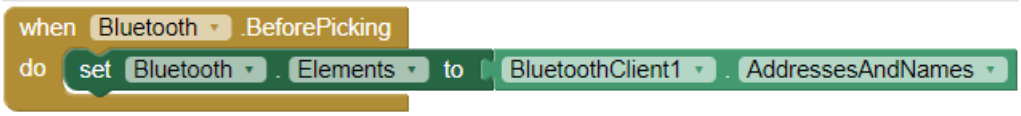


Figure AP-B 1 To include the available devices into the Bluetooth list

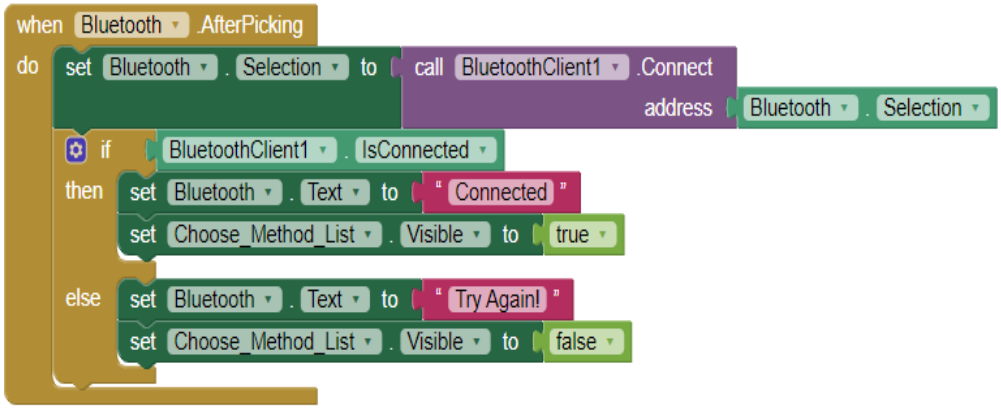


Figure AP-B 2 To make sure that the robot is connected to the chosen device successfully. If not, the user will not be able to view "methods" list.

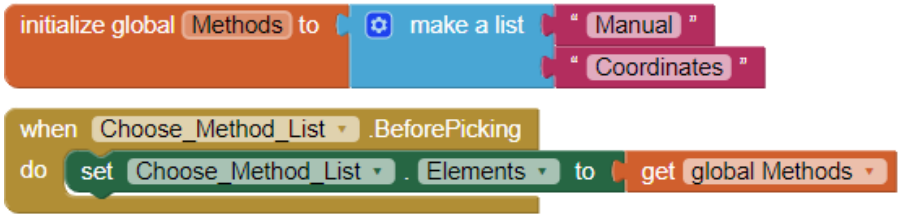


Figure AP-B 3 To set the elements in "Methods" list to be "Manual" and "Coordinates"

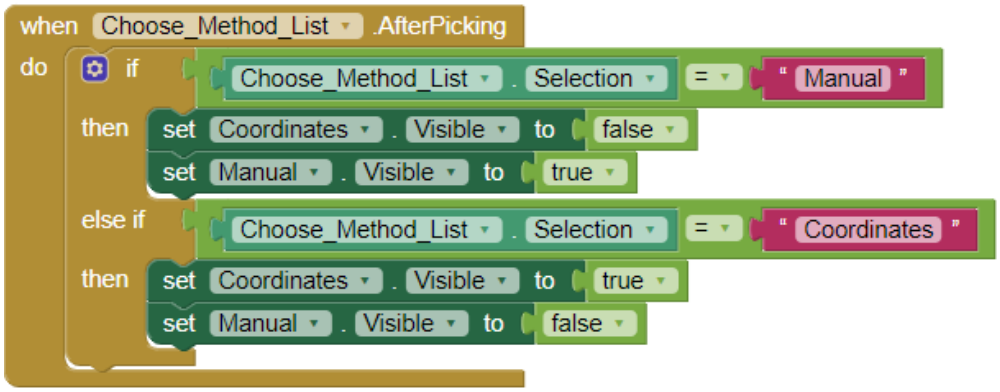


Figure AP-B 4 After picking the movement method, the user will be able to view only the related block. (ex. If user choses "Manual", only Forward, Backward, Right, Left, Stop, Restart and home buttons will be visible)

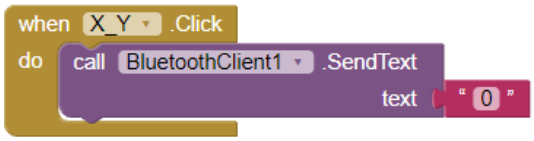


Figure AP-B 5 Code to call first method (it only sends “0” to the robot, read Arduino code to understand it)

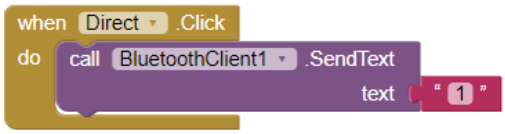


Figure AP-B 6 Code to call second method (it sends “1” to the robot)

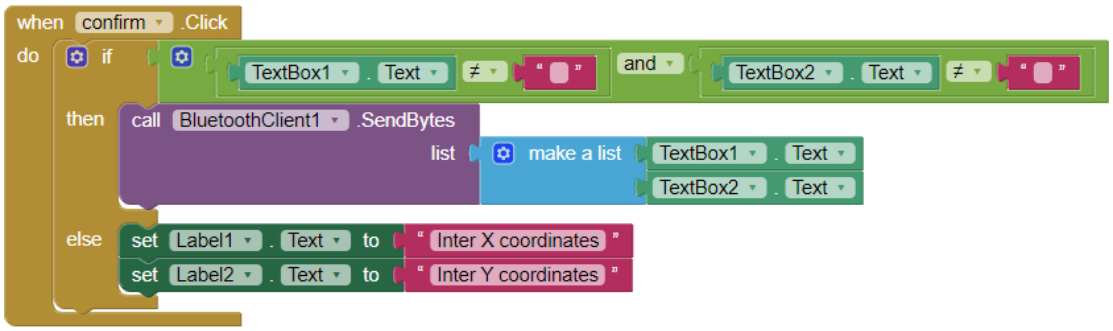


Figure AP-B 7 Code to send the coordinates when confirm button is clicked. (it sends two numbers to Arduino). If user didn’t enter any values, it notifies the user.

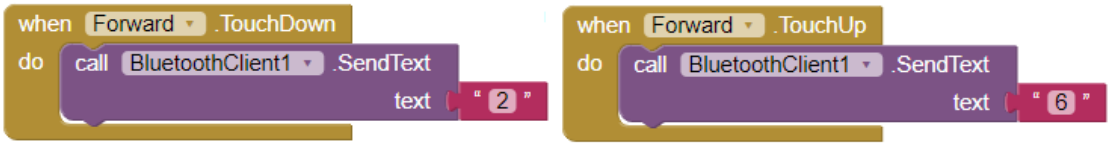


Figure AP-B 8 Code to call Forward move, (it sends “2” to the robot if the button is touched down, and sends “6” to the robot when the button is touched up)

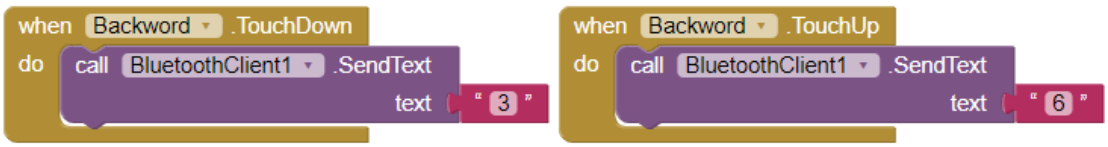


Figure AP-B 9 Code to call Backward move

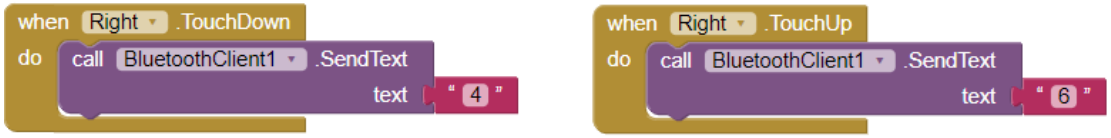


Figure AP-B 10 Code to call turn Right

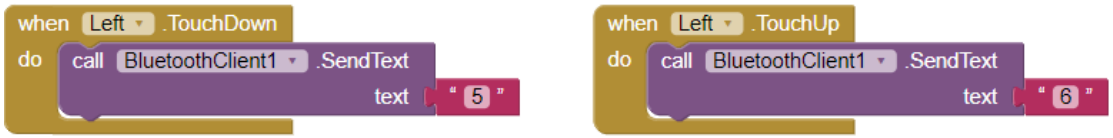


Figure AP-B 11 Code to call turn Left

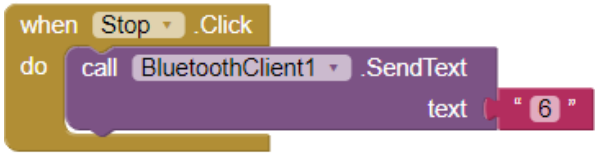


Figure AP-B 12 Code to call Stop

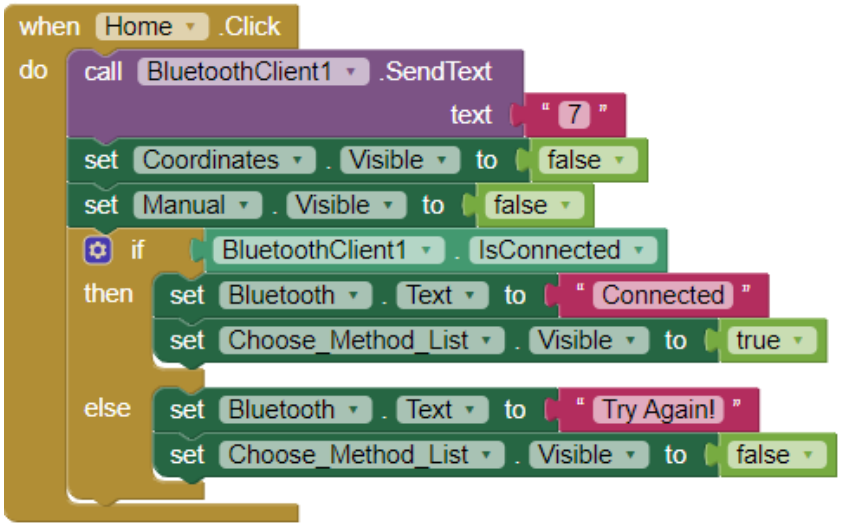


Figure AP-B 13 Code to call Home (it sends "7" to the robot, and send the user back to "Methods" list again. If the device losses its connection, it sends the user back to Bluetooth list

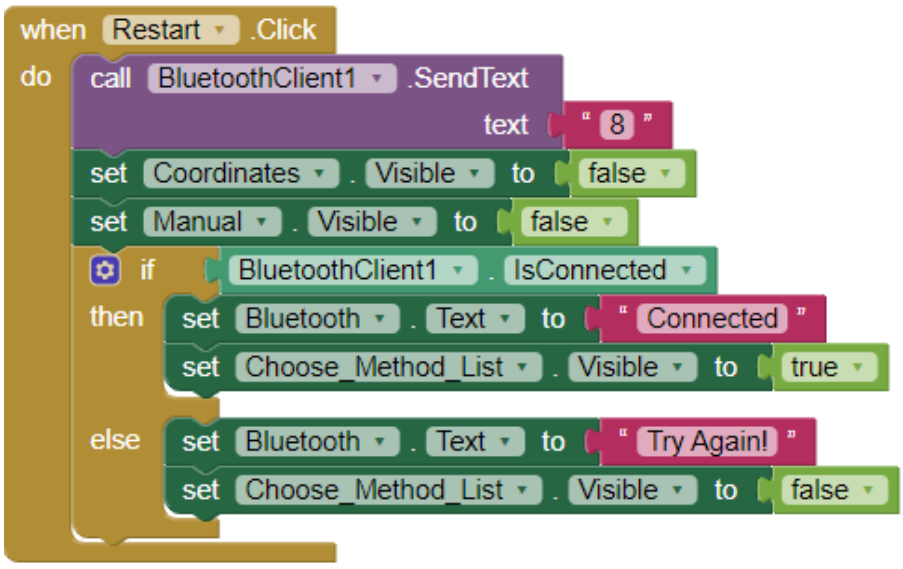
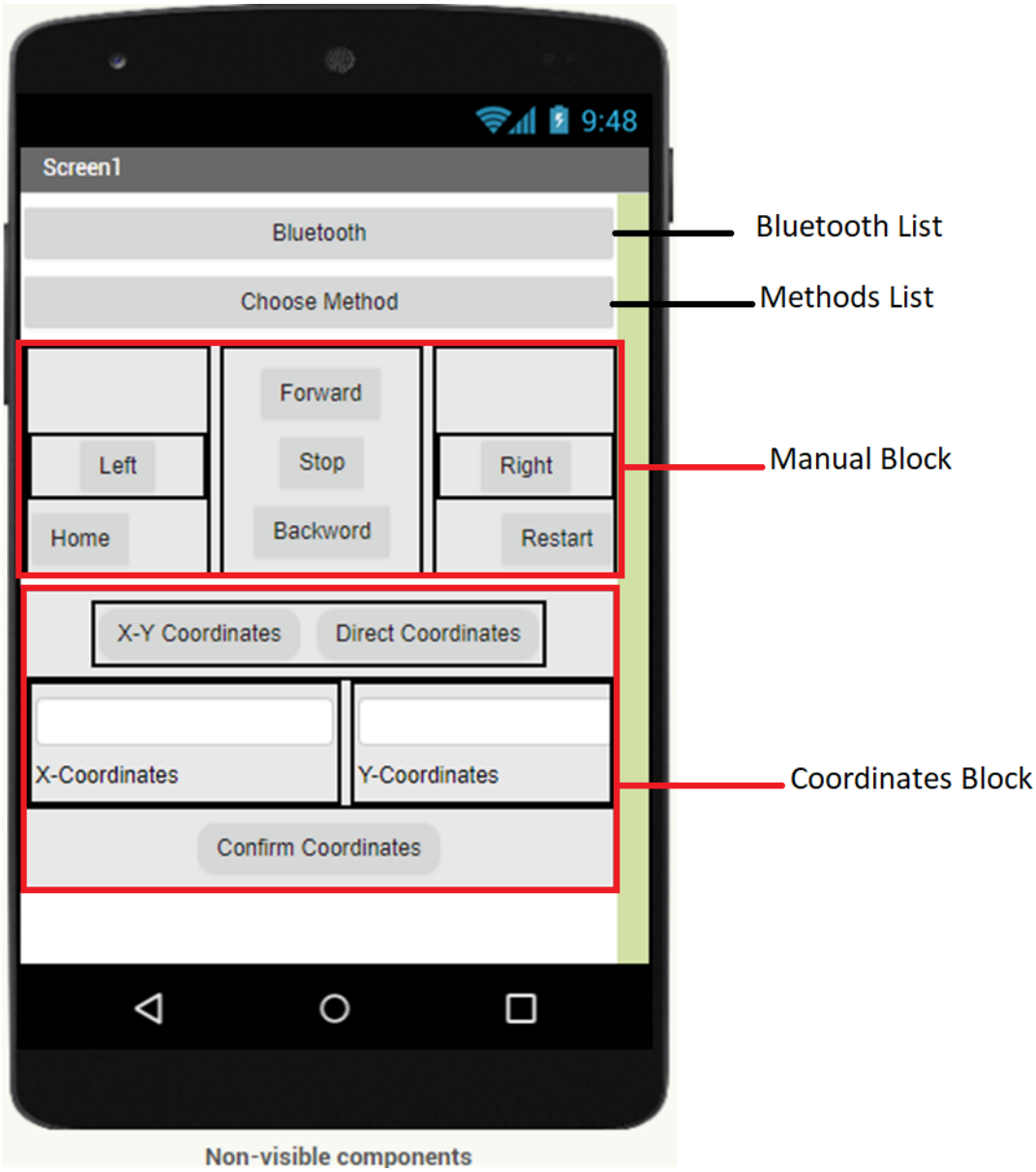


Figure AP-B 14 Code to call Restart (Similar to last one, the difference in Arduino code)


9:48
Screen1
Bluetooth
Choose Method
Forward
Left
Stop
Right
Home
Backword
Restart
X-Y Coordinates
Direct Coordinates
X-Coordinates
Y-Coordinates
Confirm Coordinates
Non-visible components
Bluetooth List
Methods List
Manual Block
Coordinates Block