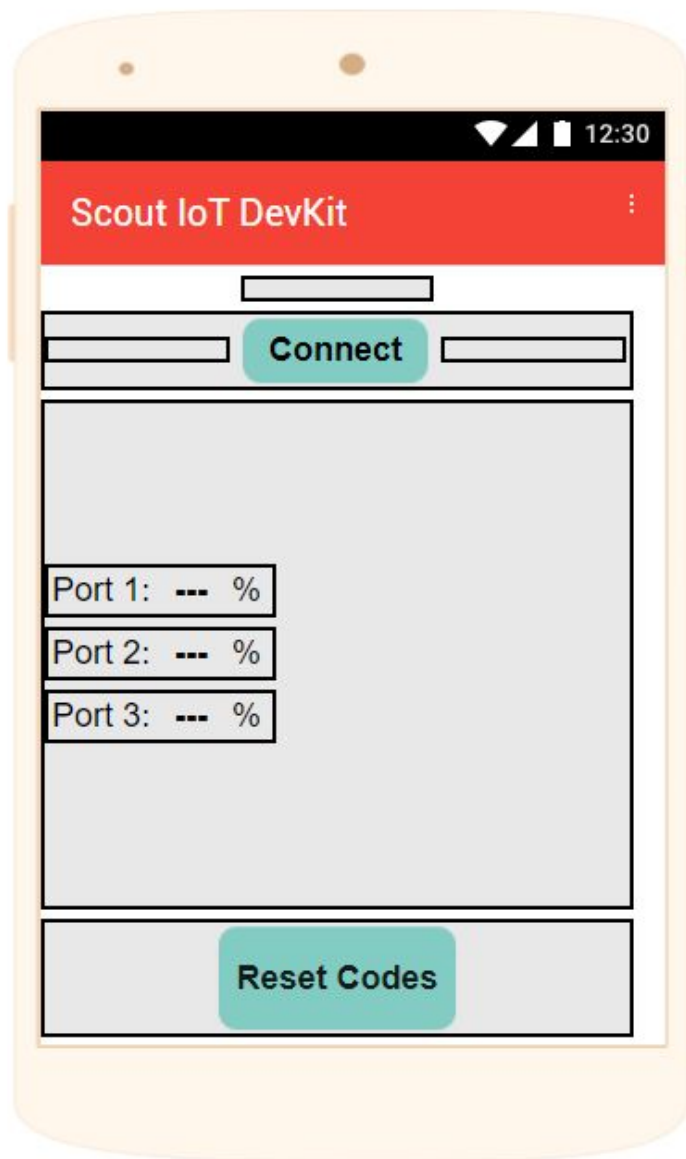


# How to Use and Modify the Scout IoT application

This application is available for Android devices only. When you open the application, you will be met with the screen seen below.



Select the 'Search' button to scan for available Bluetooth devices. To connect to your kit, tap on it. Your kit's name may be different if you have changed it in the kit's code. Once your device is connected, you will see this screen and readings from each port of the device.

Set the telemetry code for each port by selecting "Reset Codes". This code is also set in the ESP32 software. This allows the data to be displayed in the correct port

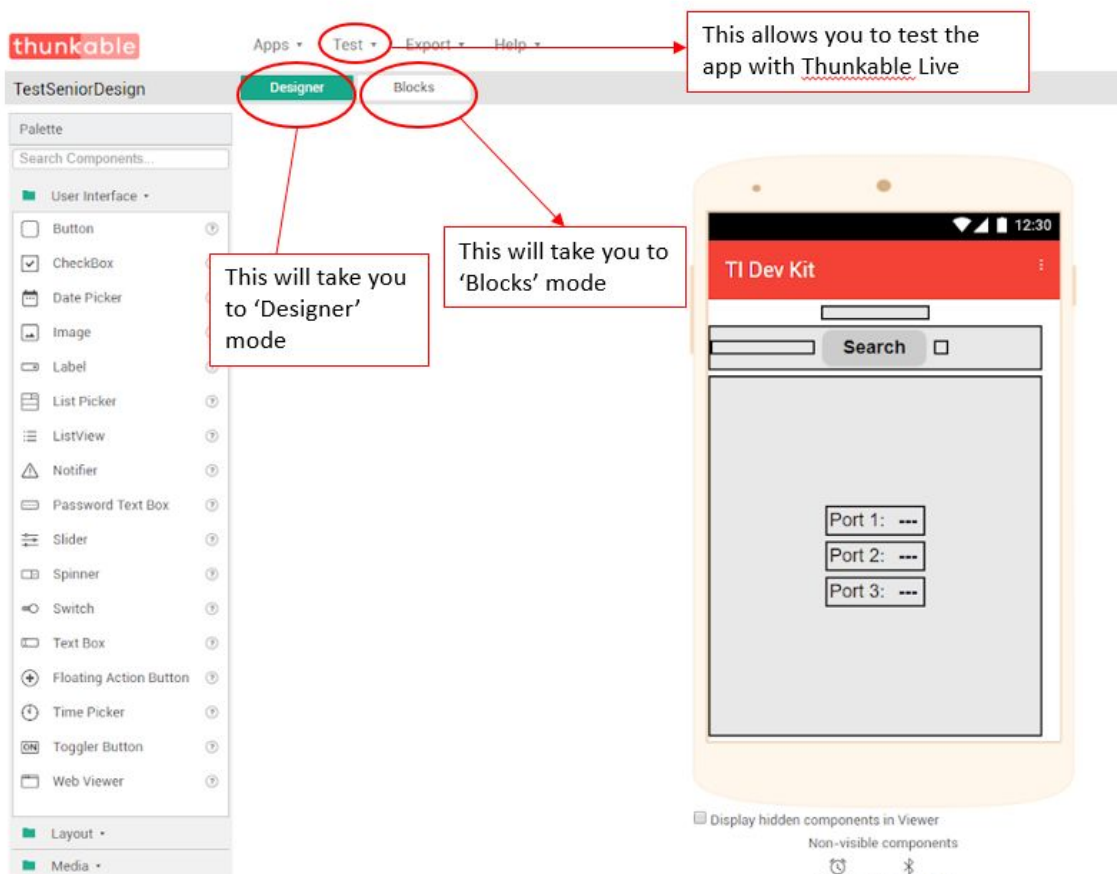
To disconnect, simply tap the disconnect button.

To view further documentation on Thunkable components, please visit <https://classic-docs.thunkable.com/thunkable-classic-android/create/components>

More information about the BLE extension is available here: <http://iot.appinventor.mit.edu/#/bluetoothle/bluetoothleintro>

To view this app in Thunkable and make your own modifications, download the .aia file. Then select the “Apps” tab in Thunkable Classic and select “Upload legacy project from my computer”. To run this app on an Android device, select ‘Export’ and save the .apk file to your Android device either via email or Google Drive.

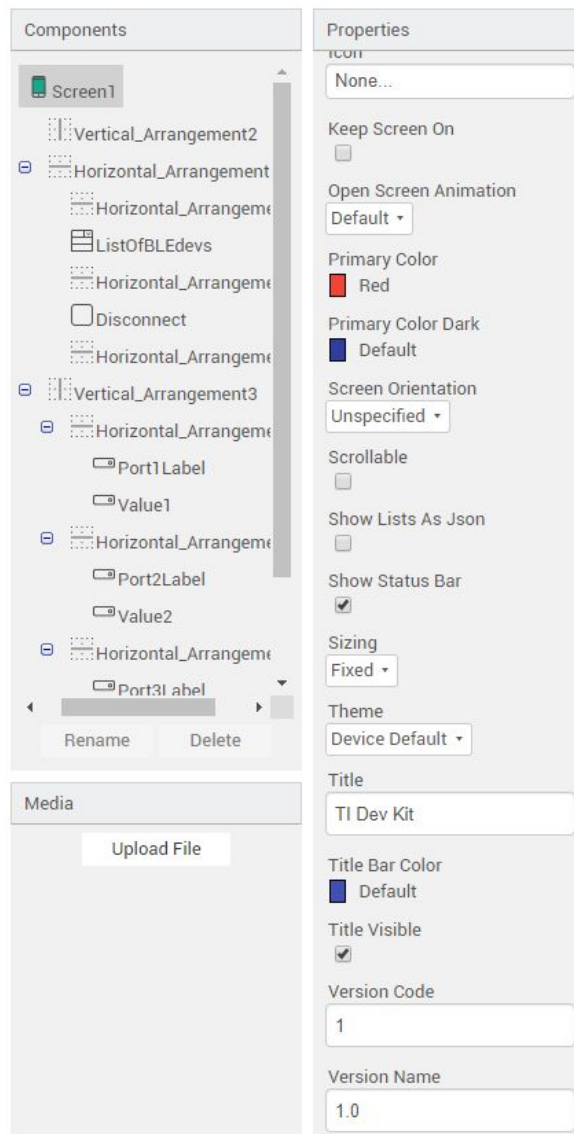
## The Basics of Thunkable



To modify this app to better suit your project, you can edit the app in Thunkable. The Designer mode allows you to add both visible and invisible components, such as buttons, text, and Bluetooth. Blocks mode allows you to create the logic for each of the components you add.

# Designer Mode

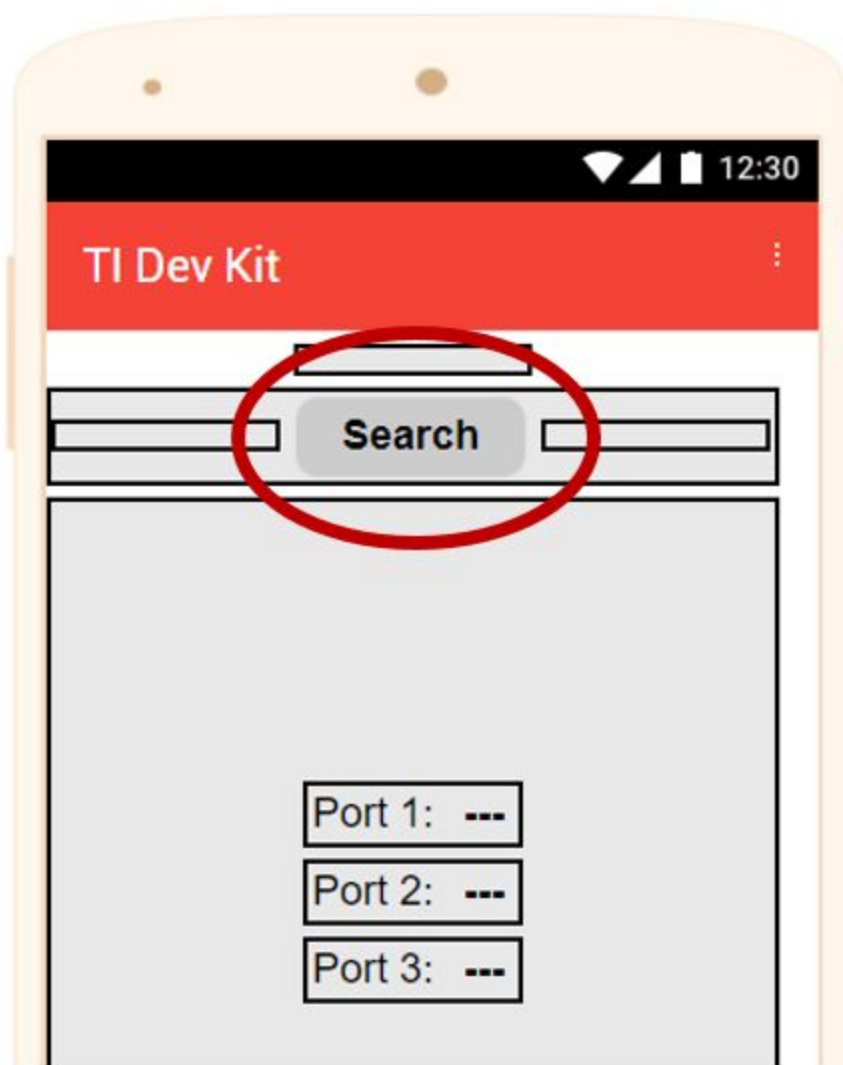
When you open the app in Thunkable, the default mode will be Designer mode. You can see the list of the components that make up the app on the right hand side of the screen. When you select a component, you will be able to edit things like font, size, and color in the 'Properties' column. The screen that is set up in designer mode is the opening screen only. If a component is not intended to be visible from the initial screen, you can unselect 'Visible' at the bottom of the 'Properties' column. Any changes to what is visible as you use the app will be done in 'Blocks' mode.



## Horizontal and Vertical Arrangements

You will see many instances of Horizontal and Vertical Arrangements in the component list. These are blocks that allow you to add things like buttons and text. The components will either show vertically (if Vertical) or horizontally (if Horizontal). To rearrange any component or arrangement, you can click and drag with your mouse.

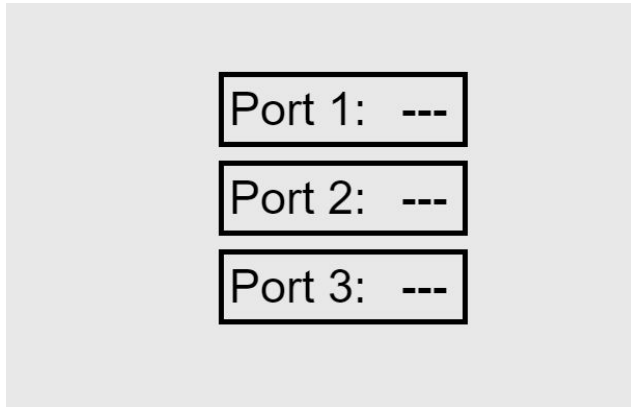
## List of Devices



A List Picker is used to show which devices are available for connection. Initially, it will show a button, in this case the button says 'Search'. Once pressed, a list will appear, in this case it will list available BLE devices. Once an item is selected, it will disappear.

## Labels

Labels are equivalent to text boxes.



Here, there are two labels in each horizontal arrangement. One label holds the text “Port #:” and the other will display the value, which is initialized to “---”. The initial value of the label should be written in the ‘Text’ box under properties. If you want the text in the label to change later, which we will do when displaying sensor readings, you must do it in the Blocks mode.

## Non-visible components

Below the phone screen you will see a selection of ‘non-visible’ components. Things that are added here are not visible in the GUI, but will be used in the ‘Blocks’ portion of the app. For this app you will see Bluetooth Low Energy and Clock 1 listed as non visible components. Bluetooth Low Energy is an extension provided by MIT App Inventor, which can be added under ‘Extensions’. To view further documentation for this component please visit <http://iot.appinventor.mit.edu/#/bluetoothle/bluetoothleintro>

☐ Display hidden components in Viewer

Non-visible components



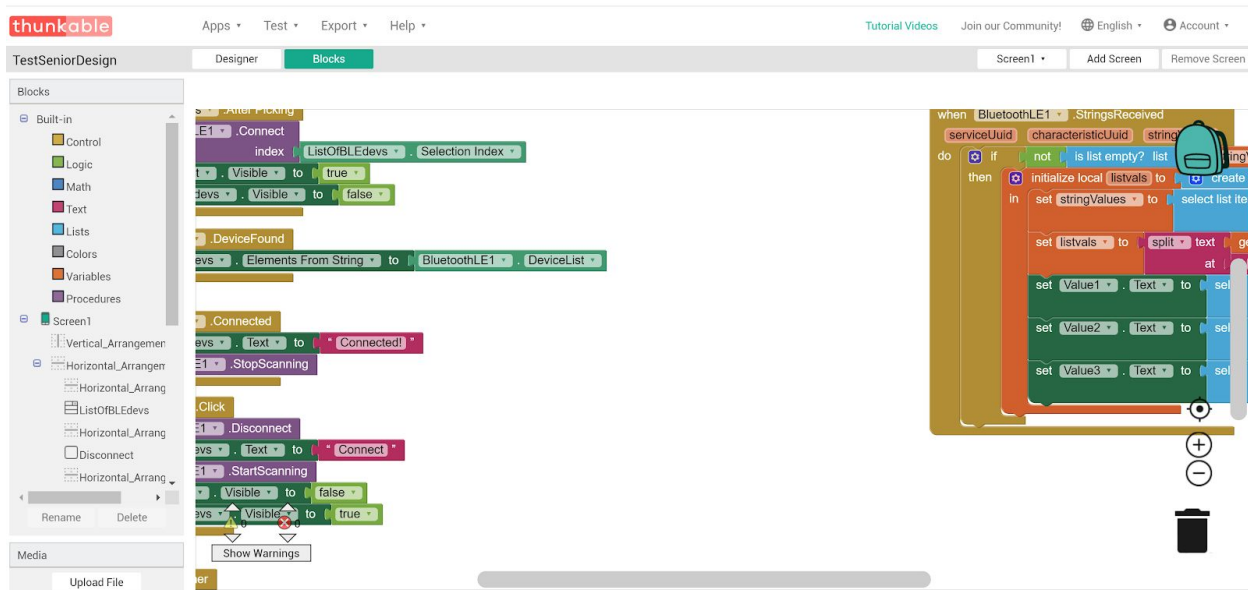
Clock1



BluetoothLE1

# Blocks Mode

Blocks mode is used to implement the logic of the app, like what happens when buttons are pressed. When you switch to Blocks mode, the screen will look like this:



On the left, you have a selection of different functions. On top are basic logic functions such as 'if, else' statements, and variable declarations. On the bottom is a list of the components you've added in 'Designer' mode, and when you select one you will be given a number of statements relating to that component. Thinkable works by connecting the logic blocks with the component blocks. Below is a description of some of these blocks.

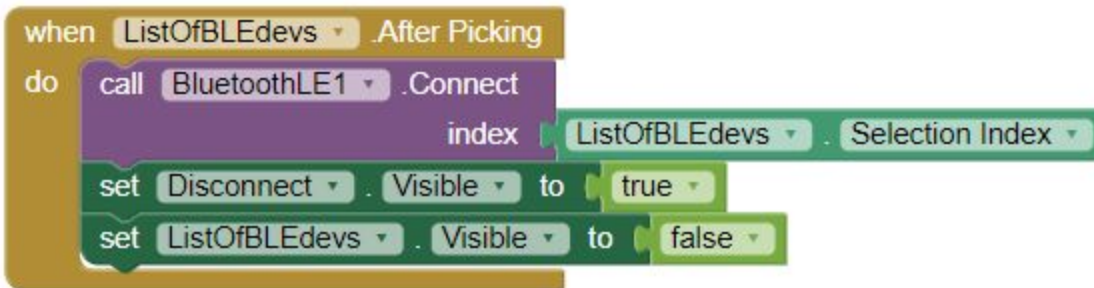
## Bluetooth



The purpose of this block is to begin scanning for BLE devices when the app opens. The yellow block's purpose is to do whatever is nested inside it when screen 1 (the only screen in this app) opens for the first time. The purple block's purpose is to call the BLE scan function. The yellow block will be found under 'Screen 1' and the purple block will be found under 'BluetoothLE1', both in the component list.



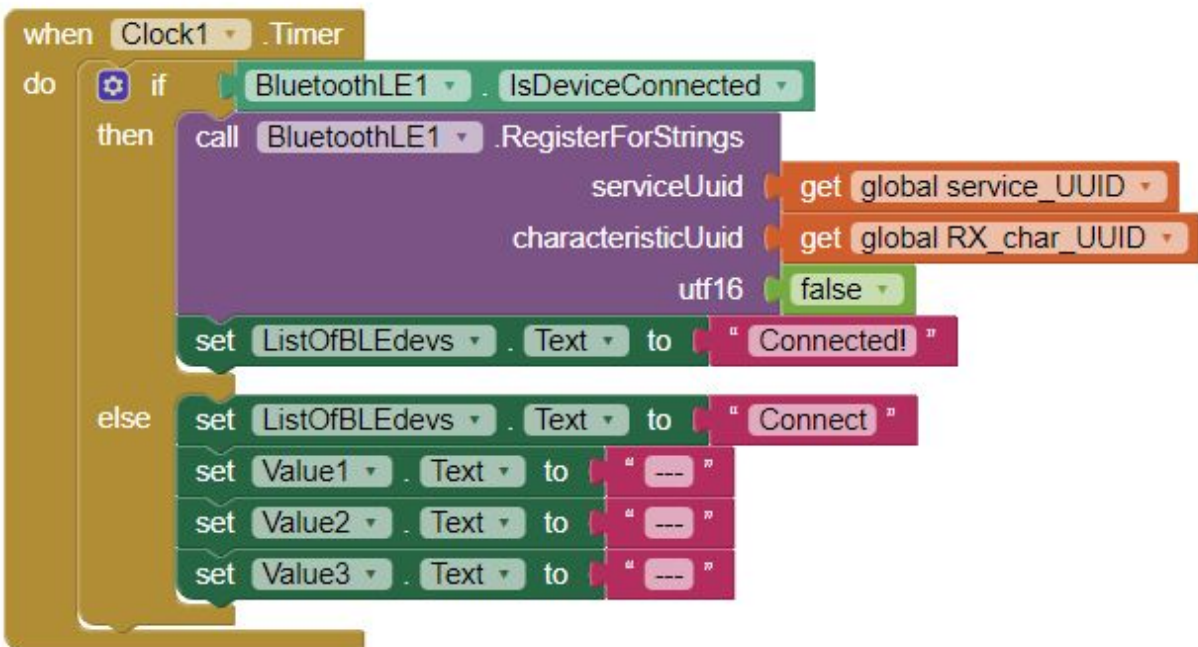
The purpose of this block is to put any device found by the BLE scan function into the list picker so we can select which device is ours.



This block describes what action to take after picking an item from the list picker. When a device is chosen it will connect using BLE, set the disconnect button to visible, then set the list picker to invisible. The 'true' and 'false' components are found under 'Logic' in the top section of components.

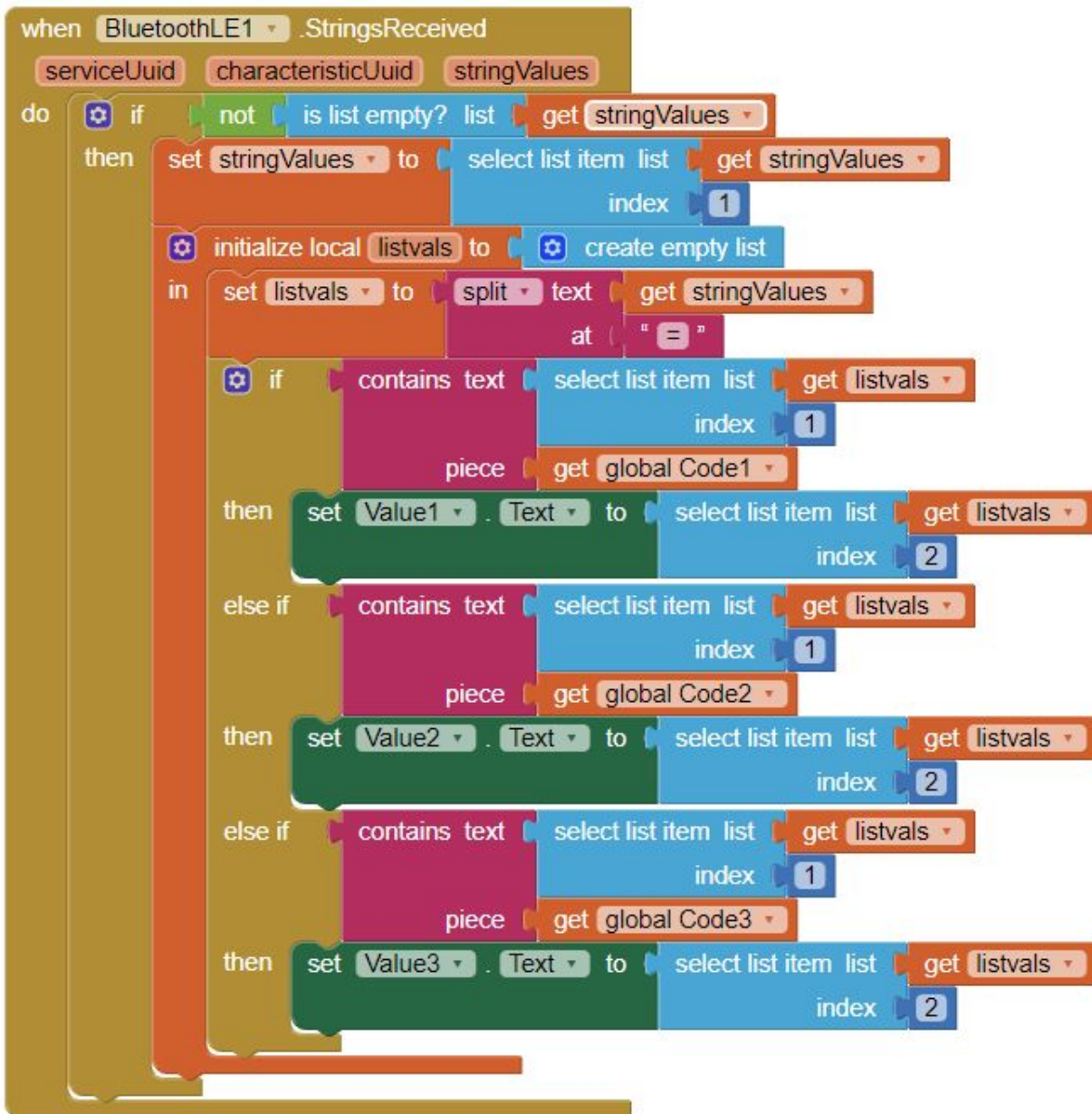


This block states what to do when a device is connected. The button on the list picker will now read "Connected!" and the BLE component will stop scanning. The hot pink text box can be found under 'Text'.



This block is controlled by a timer. The timer interval can be set in Designer mode. When the timer goes off, this block is executed. If a device is connected, the block will register the device to receive strings, rather than integers or floats. The UUID is the Universally Unique Identifier, the one used in this project is generic. It is set under global variables, which are variables that can be used in any block. Local variables are set in one block and can only be used in that block.





This block will run every time a BLE string is received. The block provides the variables of the transmit and receive UUID as well as the string value. If the value is not empty, the block will begin parsing the information. The data is sent in the format "light=50" with light being the code word and 50 being its value. A local variable, listvals, is initialized to hold the code word in the string as one list value, and the value as the other list value with the split text function. If this codeword matches one of the user input words, the value label will change its text to match the value in the list.



These blocks set the user input code words. When the code reset button is pressed, the screen arrangement will be set to the text input screen. Here the user will type in the words they want to use and the app will save them to a global variable. If a certain port is not being used, the text for that port will be set to invisible.