**Prototyping of Ardubotics**

**Team No.: 3**

**Members:**

1. G. Bhaskar 20121A0473 ECE
2. G. Dheeraj 20121A0480 ECE
3. H. Yogananda Reddy 20121A0483 ECE
4. G. Ajay Kumar Reddy 20121A0477 ECE
5. G. Rama Krishna 20121A0481 ECE
6. G. Khaleel Basha 20121A0474 ECE
7. K. Sujan Reddy 20121A0493 ECE

**Workshop:**

**Using Arduino UNO:**

1. LED example
2. Traffic Light Simulation
3. Line Follower Robot
4. Obstacle Avoider Robot
5. Zero robot
6. MIT App Inventor
7. Voice Controlled Robot

**Using Node MCU(esp8266) Module:**

1. LED example
2. Object detection using IR sensor
3. Environment Monitoring using dht11 sensor
4. MIT App Inventor
5. Voice Control Home Automation
6. WiFi scan
7. Print IP Address of ESP8266
8. Node MCU web server glowing LEDs
9. Things peak + dht11
10. Blynk + dht11 + 2 LEDs

**Using Arduino UNO:**

1. **LED example:**

* Connected an LED to digital pin 7 of Arduino and programmed to turn ON and OFF.
* Also programmed to turn on for 5 seconds and again turn off for 5 seconds.

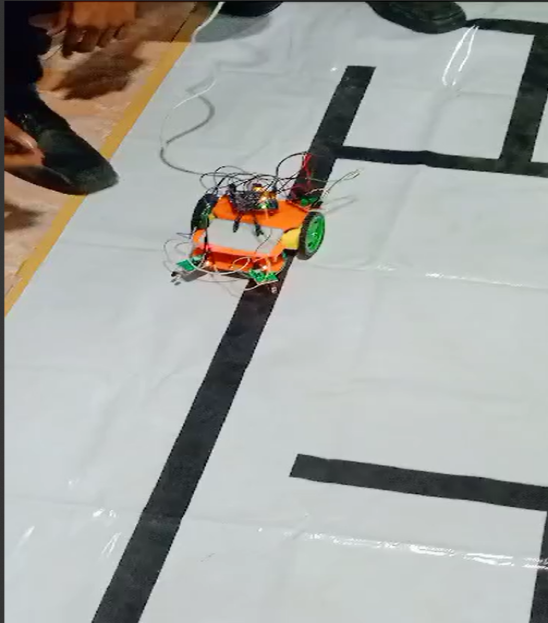
1. **Traffic Light Simulation:**

* Connected 3 LEDs to digital pins 7,8,9 respectively of Arduino and programmed to simulate the traffic light which will turn on and off one by one for 2 seconds.

1. **Line Follower Robot:**

* Materials required: 2 geared motors, 2 IR sensors, robot frame, knot screws, jumpers wire – male to male, male to female, breadboard.
* Connected all the connections as instructed by the trainer/instructor.
* We made both IR sensors downward to detect black and white surfaces and follow the line.
* Programmed the Arduino code for the Line Follower Robot.
* Tested in the path followed by black line as in the below spot taken picture.





1. **Obstacle Avoider Robot:**

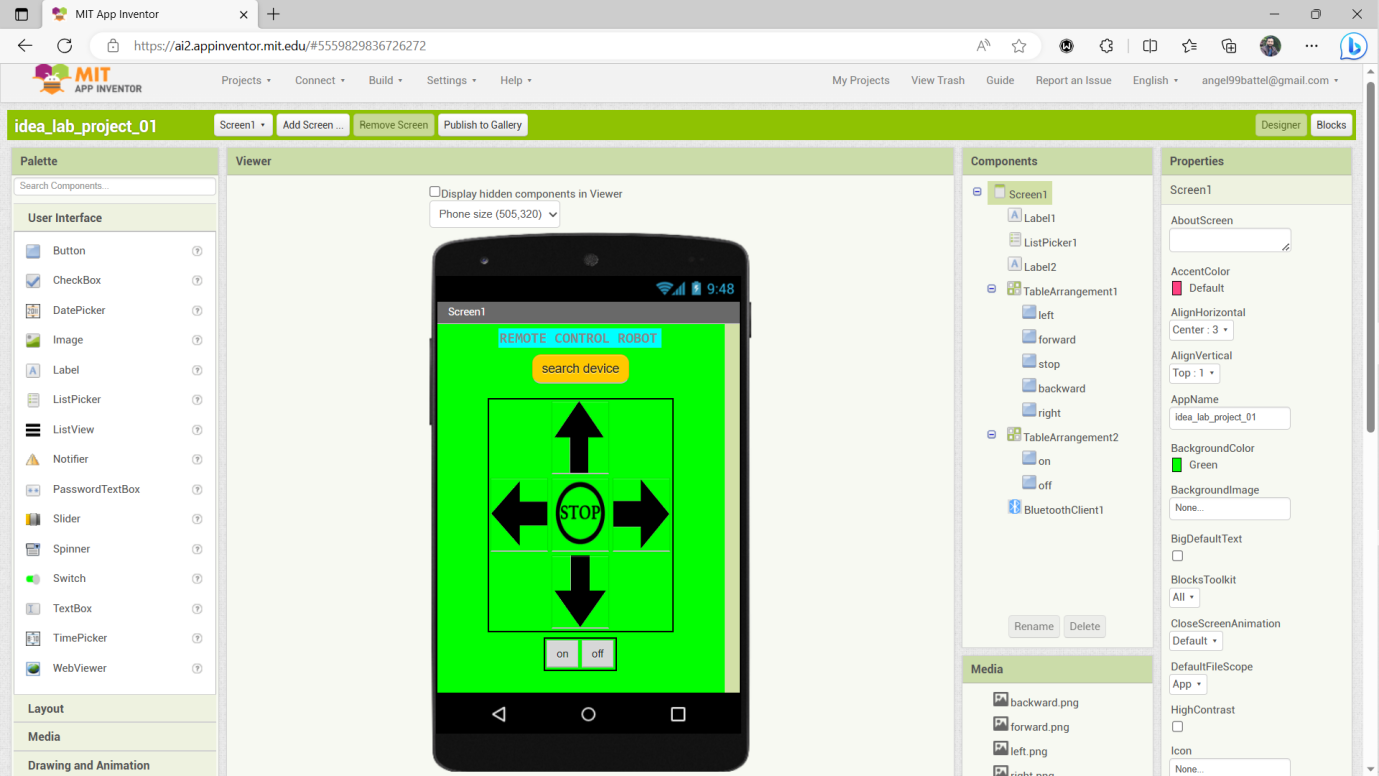
* All the connections and materials are the same as the Line Follower Robot.
* In this one, we faced both IR sensors vertically instead of downward to check obstacles ahead in the environment.
* We have changed the code to stop the robot for the first test.
* The second test reprogrammed the robot to avoid all the obstacles coming in front and take another path where there is no obstacles.

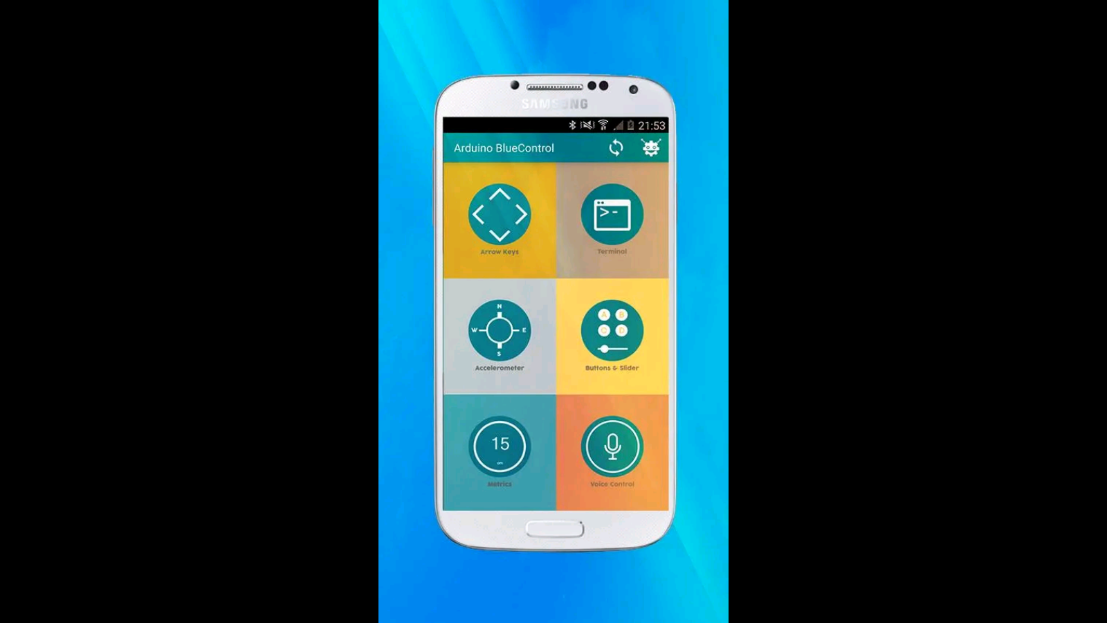
1. **Zero Robot:**

* It is a robot that works in loops forever without stopping.
* Materials required: 2 geared motors, Arduino, robot frame, knot screws, L293D motor driver module.
* Connections are made according to instructions.
* Programmed the robot to run straight for 3 seconds, take a 90-degree rotation to the right, and run straight for 3 seconds, which in loops will make a square forever.

1. **MIT App Inventor:**

* MIT App Inventor is a website where we can build our own app for controlling robot.
* We use designer tools and blocks of code that will make our required app for controlling the robot using a Bluetooth module(HC-05).
* We have designed the forward, backward, left, right turns, and stop buttons in the app and installed on Android phone.
* We have connected the HC-05 Bluetooth module to Arduino which is TX and RX pins.
* And then tested the app which allows the robot to move by clicking what actions we want to make.





1. **Voice Controlled Robot:**

* Materials Required: 2 geared motors, Arduino, robot frame, knot screws, jumpers wires, HC-05 Bluetooth Module, breadboard.
* Connections are done as per instructions, uploaded the code to the Arduino board.
* Downloaded the app called ‘***Arduino Bluetooth Control’*** *by broxcode* and configured the vocal commands for controlling the robot.
* Vocal commands: forward, backward, left, right, stop.
* Again added a headlight for the second test in the same robot and configured the commands for headlight – headlight on and headlight off.

**Using NodeMCU(esp8266) Module:**

1. **LED example:**

* We have installed the esp8266 board manager in Arduino IDE so that it can be recognized and programmed.
* Connected an LED to pin D5 of the esp8266 board and programmed the board using Arduino IDE.
* LED glow for 2 seconds and off for 2 seconds according the code.

1. **Object detection using IR sensor:**

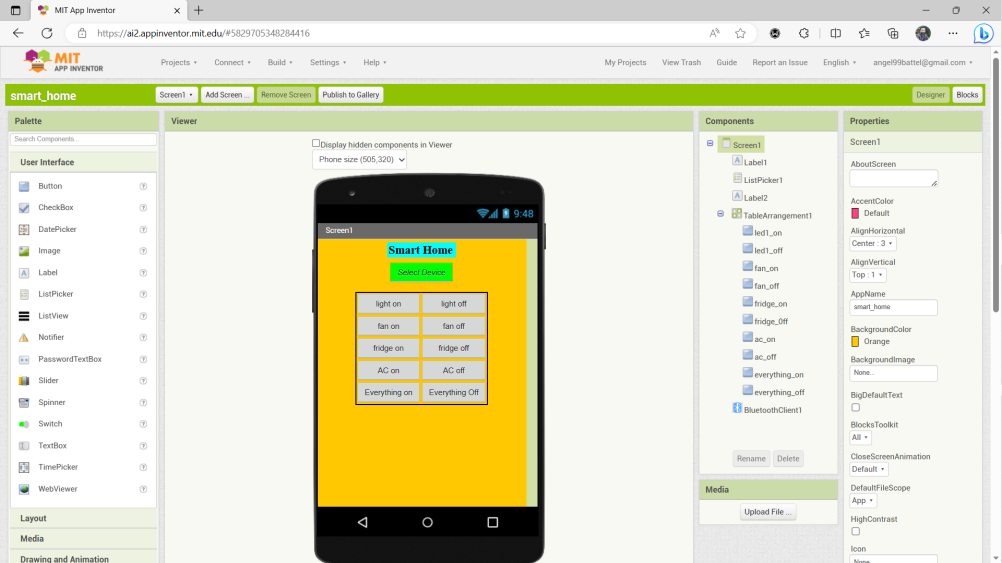
* Materials required: esp8266, a IR sensor, a LED, jumpers wire, and breadboard.
* Connections are made according to instructions.
* Programmed the board so that the object is detected then it turns on the LED if not detected the off the LED.

1. **Environment Monitoring Using dht11 sensor:**

* Materials required: esp8266, dht11 sensor, jumpers wires, breadboard, 10K resistors.
* Connections are made according to instructions and programmed the esp8266 by importing the dht11 library into code so that we only need to write small code and the rest will be handled by the dht11 library itself.
* We have calculated the temperature, and humidity values of both Fahrenheit and Celsius and the temperature index of the environment and printed in the Serial Monitor of the Arduino IDE.

1. **MIT App Inventor:**

* We have designed and built the app for Home automation.
* Added the control for on and off of different appliances like light, fan, fridge, AC.
* And also to on all appliances and off all appliances.



1. **Voice Control Home Automation:**

* Materials required: 4 LEDs for AC, fan, fridge, and light, HC-05 Bluetooth Module, esp8266, breadboard, jumpers wires.
* Downloaded the app from Play Store ‘***Arduino Bluetooth Control’*** *by broxcode* and set the commands like light on, light off, AC on, AC off, fan on, fan off, fridge on, fridge off, everything on, everything off.
* Tested the commands it worked perfectly.
* And in the second test, we tested in our own app created by MIT App Inventor and this time also worked perfectly.

1. **WiFi Scan:**

* Materials Required: esp8266
* Connect the board to the laptop and upload the program for scanning the available WiFi networks and print the SSID to the serial monitor and every 10 seconds it scans and prints the available networks.

1. **Print IP Address:**

* Upload the code for printing the IP Address of the esp8266 after connecting to the given WiFi SSID and password as in the code.
* Print the IP address of the esp8266 in the serial monitor.
* We got IP address as 192.168.137.111

1. **NodeMCU web server glowing LEDs:**

* Materials required: esp8266, jumpers wires, breadboard, 3 LEDs.
* Connect LEDs to D0, D1, and D2 of the esp8266 and program the esp8266 so that it will print the web address and paste the web address to the browser and the LEDs can be controlled from there.

1. **Thingspeak + dht11 :**

* Thingspeak is the cloud IoT and we program the esp8266 with different sensors and can upload, and monitor the sensors' activities from there.
* Materials required: esp8266, dht11, 10K resistors, jumpers wires.
* Connections are made as instructed.
* Programmed the esp8266 and set up the thinigspeak.com cloud and we can monitor the activities of the sensors' value and also we can share with anyone.

1. Blynk + dht11 + 2 LEDs

* Blynk is the cloud IoT similar to thingspeak.com cloud but it is more advanced and we have more features than thingspeak cloud.
* Connections are the same as above 9. No. and we have added 2 LEDs.
* Programmed the esp8266 and set up the Blynk cloud as the connections were made.
* Now we can control the LEDs and monitor the dht11 values from Blynk cloud.

