Team LunaTechs
Team ID 49

Kit for Visually Impaired

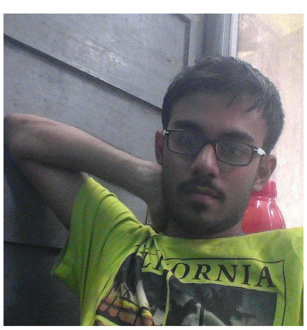
INTRODUCTION

The basic idea behind our project is to make a device especially for the differently abled ones and make their day to day activities much better and hassle free.

MOTIVATION

The motivation behind our project lies in the lack of quality products which helps the daily working of a differently abled person. Our project aims to provide a cheap, reliable and efficient device to help fill this big hole.

TEAM MEMBERS



Mandar Bhurchandi

Mohit Patil

Fenil Vanvi

Gaurav Waratkar

150070049 150050017 150050003 150100038

WORKING OVERVIEW

The project involves use of a regular shoe integrated with multiple IR, ultrasonic sensors which are controlled by an Arduino-Uno board. The sensors have been used in such a way that as many possible objects and obstacles which are encountered often are possible to detect. The Arduino board then processes the code with respect to the values from the sensors and sends an appropriate code through a Bluetooth Module to an android smartphone app. This app then processes the code and prompts the user about the obstacles ahead.

DETAILED WORKING

- **Electronics Part**

1. Sensors :
   The working of basic sensors is given here:
   https://stab-iitb.org/electronics-club/tutorials/basic_electronics/
   Initially we worked with TCRT5000 because of its in-built noise reduction feature but found its outputs varying non-linearly with distance of the obstacle. So we decided to go for normal IR sensors. We tested these and found their values to be stagnant at some 15 cm or so. Hence we decided to go for 4 LEDs for one Photo-diode. This solved that problem and we got good change in outputs for distances over 25 cm which we needed.
   We tested ultrasonic sensors HC-SR04 for distances 2 cm to over 300 cm, (though we finally only used them for max 80 cm).

   The sensors were fit on the shoe as shown in the pictures.
   We had the IR sensor placed in the front end of the shoe pointing towards ground for quicker the detection of stairs, potholes, etc.
   The HC-SR04 sensors were fit as follows: Two sensors at the front of the shoe at angle of approx 7.5 degrees with the front pointing line. One sensor

was attached on the right for the right shoe and left for the left . The last sensor being at the position of the shoe laces pointing approx at 45 degrees with the floor for the detection of objects with small legs but bulk main bodies like chairs.

2. We connected these all sensors to an Arduino-Uno board.odd numbered pins were trigger pins, even numbered were echo pins for ultrasonic sensors. +Vcc and ground was common to all.In the usual IRled-photodiode circuit(https://stab-iitb.org/electronics-club/tutorials/basic_electronics/)we added three more IRled's parallel to the one already present and took the power supply to pin 10, so that led's will blink when a pulse is given to the pin, and analog output across photodiode is taken in from A0 (analog input) pin(this isn't necessary, but it gave us a little more battery life). Some basic working of the arduino-uno board and concepts related to it are mentioned here:
https://stab-iitb.org/electronics-club/tutorials/arduino/
Functions for NewPing library and the constructor for arduino and their explanations are given on this page:
http://playground.arduino.cc/Code/NewPing.
In practical use of the sensors, there are often some junk values.Now, any variable value which causes the arduino to pass an integer to the android in this project was called a stop signal, because to react to these the user will have to stop walking. To guard against the junk values, we took each input twice, only if the input was causing the stop signal.
3. The Arduino code we used for our project's left shoe is given in the file: Left_shoe.ino for left shoe. Same code with trivial changes(changes in name of variable) in the output was used for right shoe.
The IR sensors are calibrated according to the values we got for stairs and potholes, and may give different output for when in different light conditions.

4. The output was transmitted through a Bluetooth module HC05 to the android app, where the messages are given to the user.In this part we faced a unique

problem. Once we got a proper output on the serial monitor of the arduino IDE, we connected it to the android (via bluetooth) the ultrasonic started giving a wrong output!! It would show an obstacle in the front even when there wasn't any. If you face the same problem don't be baffled, the problem is probably in connection or serial comm. Check connections and take output from either the android or the arduino IDE but not both(not taking output from arduino means disconnecting the PC chord and running it on battery power.
http://www.robotshop.com/media/files/pdf/rb-ite-12-bluetooth_hc05.pdf .
The default Bluetooth name of the device is HC05 and password to pair with is 1234.

5. Delays were made to give time to spell out the messages, and output was also taken with a buzzer(that's the use of the outputPin).
The list of outputs at outputPin and their meanings for left shoe are as follows:
Short beep : obstacle in the front and on the right.
Long beep : obstacle in front only.
Double short beep : stair ahead.
Double long beep : : pothole ahead, no obstacle on the left
Long Beep then Short beep :

- **Android Part**

1. We started working on the Android part with the objective of processing the data from HC05 Bluetooth module and then sending simple prompt messages to the user describing the obstacle's direction.
2. We did the programming of the app using MIT App Inventor 2 because of its simplicity and predefined commands.
One can find the basics of the App Inventor here:
http://appinventor.mit.edu/explore/library.html
3. The basic run through of the code/working of the app:
When the app is opened we get a screen as given in the screenshot of the

designer block. One can select the bluetooth module of each leg respectively by clicking on the Select Bluetooth Module tab and selecting from a list of the devices that have been paired with the smartphone.

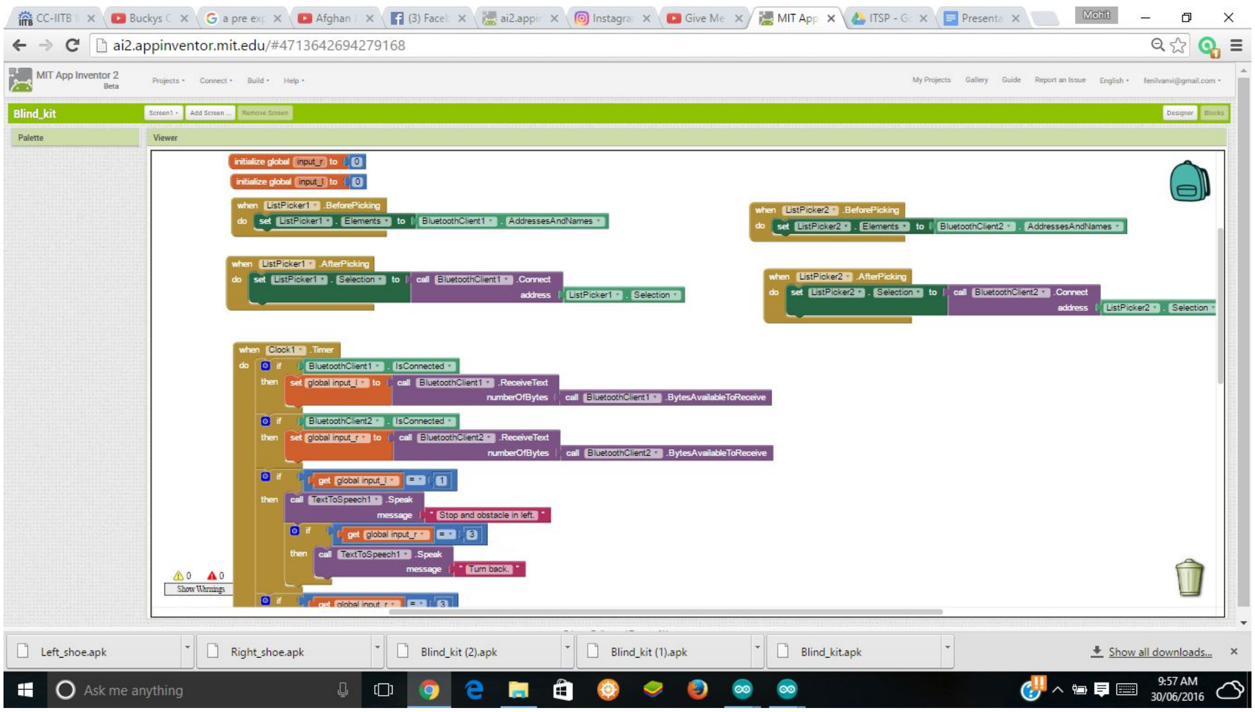
The processes being worked on can be seen in the block screenshot 1,2 given below.

We defined two global variables ‘input_r’ for right shoe and ‘input_l’ for left shoe and initialized both of them to 0. These variables would be later used to store the data coming from respective shoes.
The ListPicker1 implies the function ‘Select Bluetooth module 1’ and ListPicker2 for ‘Select Bluetooth Module 2’. When the bluetooth module isn’t selected i.e. ‘before picking’, we set the elements of ListPicker1 to the ‘Names and Addresses’ of the Bluetooth devices which have been paired and make them available for ‘Selection’.
When the user selects HC-05 from the list the app comes to know that an element from the list has been picked.
The app then directs the ‘BluetoothClient1’ i.e. the smartphone bluetooth device to connect to the Address of the device which was ‘Picked’. Same for ‘BluetoothClient2’ which works for right shoe.
So now a connection between the bluetooth modules of both the shoes is complete.
Then we store the data incoming from the Bluetooth module1 i.e. left shoe in the global variable ‘input_l’. Same for right shoe into ‘input_r’. The Bytes available to receive is the default format used to receive the data from the bluetooth device connected to the app. When the clock triggers the loop, the Bluetooth client receives data from the device connected i.e. HC05 in our case.
Once the data is received, we analyse the data and send messages to the user accordingly using the ‘Text to Speech’ command where we can type in any statement or phrase or words, the system will speak through speaker/earphones(if connected) the exact same text.
The data is analysed depending on the digit received. We are sending a

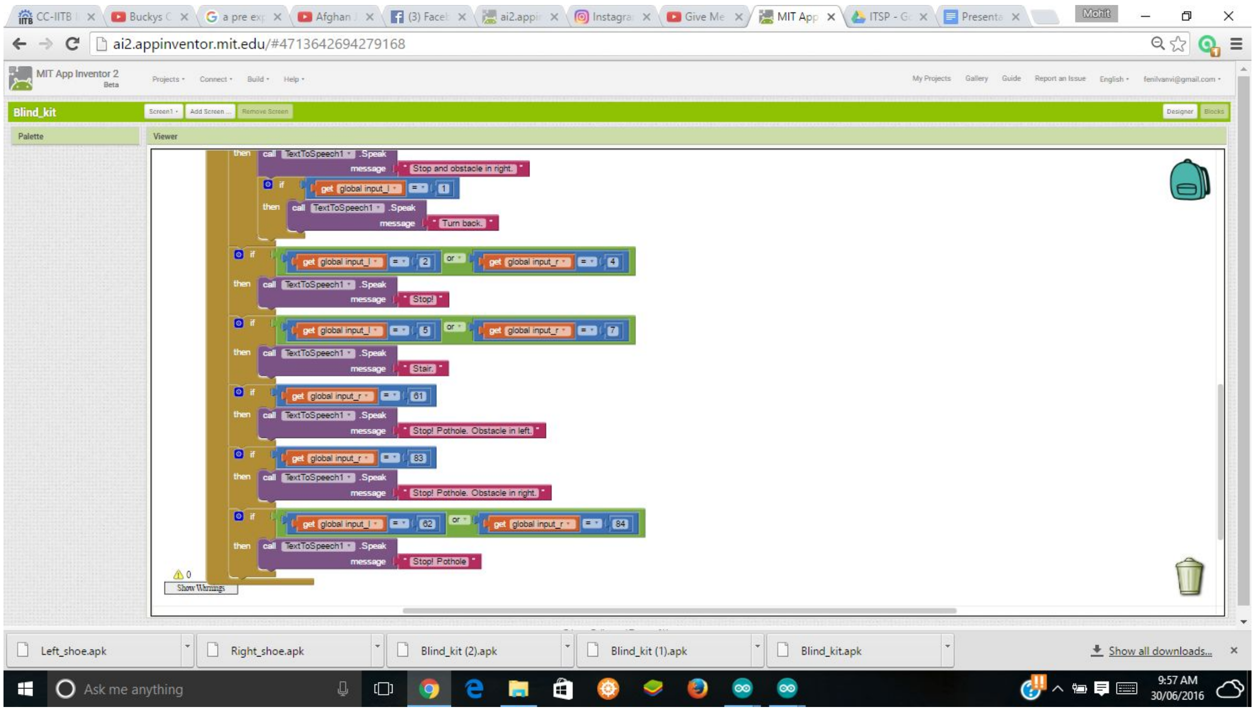
number from 1 to 9 regarding specific obstacles through Arduino and accordingly a message is given to the user. The IR values for common stairs, potholes, etc have been used to differentiate between them and notify the user of the same. The code is self-explanatory for this part.

The blocks used in the app were:

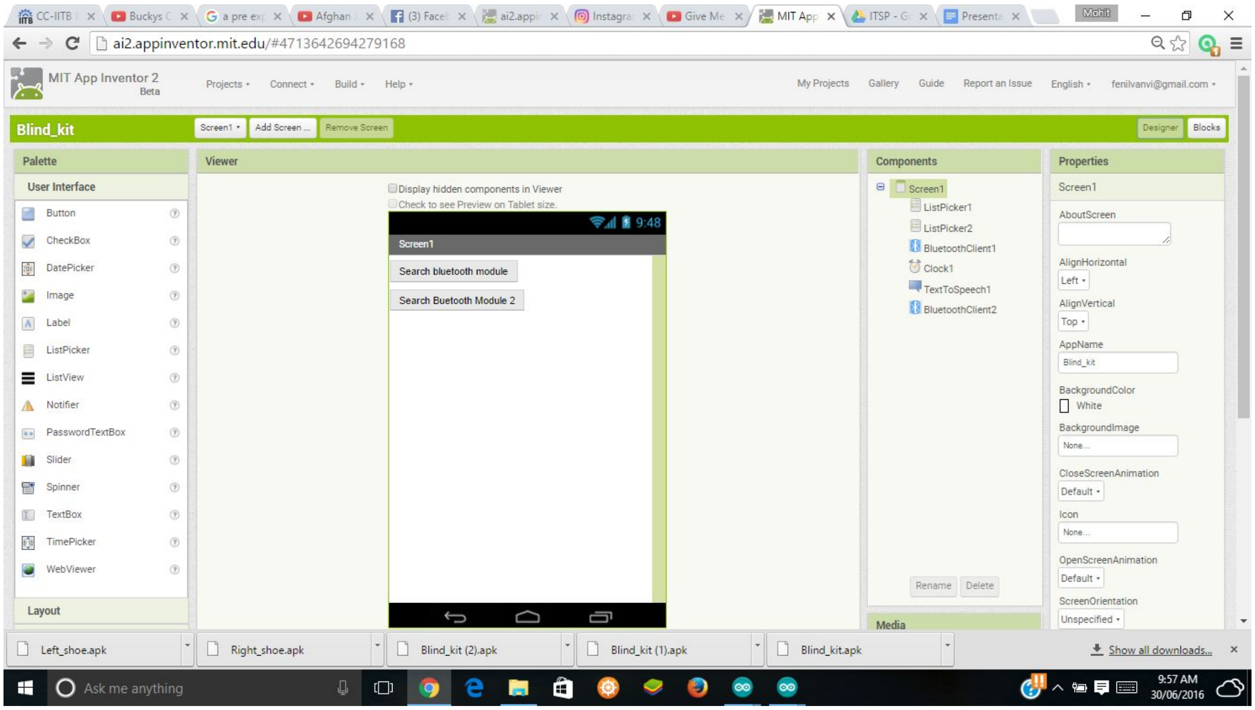
Part 1:



Part 2 :



The Design of the home page of the app is:



WORKING OF THE PROJECT

You can find the videos of the working of the project here:
https://drive.google.com/open?id=0B7Q2EPPm3igrUXV5OXZvemVEWUU
https://drive.google.com/open?id=0B7Q2EPPm3igrR0IzOU5hT3IwczQ

COMPONENTS USED

- 12 Ultrasonic sensors HC-SR04, 2 PIR sensors with 4 LEDs each
- 2 Arduino-Uno board
- 2 Bluetooth Module HC-05
- Components like jumper wires, connecting wires, 9V batteries etc.
- Basic stationary like DST, insulating tape, etc.

- Shoes
- Cotton, cloth used to hide the wires, boards, etc.

Total Approx Cost : ₹ 3800.