PROJECT REPORT

ON

ARDUINO REMOTE CONTROL CAR

SUBMITTED BY :

C.LALRAMNGHINGLOVA
ROLL NO : 2023BCAOO1
REGN. NO : 2001178

PC.LALPEKHLUA
ROLL NO :2023BCA006
REGN. NO : 2001182

BACHELOR OF COMPUTER APPLICATIONS (BCA)
DEPARTMENT OF COMPUTER SCIENCE,
HIGHER AND TECHNICAL INSTITUTE, MIZORAM (HATIM)
KAWMZAWL, PUKPUI, LUNGLEI - 796691

MIZORAM UNIVERSITY, TANHRIL: AIZAWL

2023

HIGHER AND TECHNICAL INSTITUTE, MIZORAM
KAWMZAWL, LUNGLEI – 796701

CERTIFICATE

This is to certify that C. LALRAMNGHINGLOVA and PC LALPEKHLUA has fully completed the project entitled, “ARDUINO REMOTE CONTROL CAR” in order to meet the requirement of the Mizoram University for the VI Semester Bachelor of Computer Application in the year 2023 (January - May). It is to certify that all the corrections/suggestions indicated for internal assessment has been incorporated in the project. The project report has been approved as it satisfies the academic requirements in respects of the project work prescribed for the BCA Course.

	Vuansanga Vanchhawng	
(H. LALRUATKIMA)	(VUANSANGA VANCHHAWNG)	(K. LALMUANPUIA)
Project guide	Principal	Head of Department
Dept. of Computer Science	HATIM	Dept. of Computer Science

H.O.D.
Department of Computer Science
Higher & Technical Institute,Mizoram
Lunglei : Mizoram

Name of Examiner

LALREMPUIA SAILO

Signature with date

ACKNOWLEDGEMENT

With a sincere heart we express our gratitude to all who have supported us in order to accomplish our project and make it a reality. We are obligated to say thanks to the following persons for their support and directions.

Firstly, we would like to give thanks to God for His guidance all throughout the processing period of our project. With a grateful heart we thank and acknowledge God, for without Him we would not be able to accomplish this project of ours. He has given us the knowledge, wisdom and good health we needed for making this project into a reality.

We thank to our Principal Mr.Vuansanga Vanchhawng. We thank him for the support, encouragement and permission which we enjoy freely in order to accomplish our project.

Our heartfelt gratitude goes to Mr.H.Lalruatkima, Project Guide and Mr.K.Lalmuanpuia, Head of Department in Department of Computer Science, Higher and Technical Institute, Mizoram for their continual advice, backup, confidence, patience and their encouraging words which gives us courage and confidence without which we would be nowhere.

We thank all the people in the department of Computer Science for being always available and helpful over the semester. Last but not the least; we thank our parents for their moral support and encouragement.

PC.Lalpekhlua
C.Lalramnghinglova

VI Semester BCA
Higher and Technical Institute, Mizoram

Abstract

This project report includes the description and implementation for Arduino Remote Control Car. This project deals with the construction using Firebase (for back-end) which plays a vital role in this project.

This application mainly aims display and produce the necessary information about the Institution, news will be provided through this apps by using the news page and Institution important Events will be display.

With the help of this project, finding and searching about the institution can easily be done and quicker than asking the Faculty or Principal. The entire project helps the users to spend less time than manually consulting the brochure.

Finally, this project deals with the easiest and simplest layout which made it simple to understand and to work with it.

DECLARATION

We do hereby that the project entitled "Arduino Remote Control Car" is not submitted to any other university or institution for the award of any degree, diploma of fellowship or published any time before. This project is prepared under the guidance of our project guide Mr. H.Lalruatkima which forms our partial fulfilment of the requirements for the three years Bachelor Degree in Computer Applications of Mizoram University.

List of figures

Page No.

Fig.1 *ER Diagram……………………………………………………………..17*

Fig.2 *Data Flow Diagram…………………………………………………….18*

Fig.3 *Firebase Realtime Database table………………………………...19*

Fig.4 *Firebase Storage table……………………………………………….20*

Fig.5 *Home……………………………………………………………………..21*

Fig.6 *Bluetooth………………………………………………………………..22*

Fig.7 *Contact...…………………………………………………………………23*

Fig.8 *Mic…..……………………………………………………………………..24*

CONTENTS

Page No.

Acknowledgement……………………………………………………………………..…. 3

Declaration…………………………………………………………………..……………. 5

Abstract……………………………………………………………………..………………. 4

List of Figures…………………………………………………………..…………………. 6

1. INTRODUCTION (PROJECT ANALYSIS) *…………………………….9*

1.1 Overview of the Project

1.2 Objectives of the Project

1.3 Scope of the Project

2. SYSTEM REQUIREMENTS*…………………………………..……………11*

2.1 Hardware Requirements

2.2 Software Requirements

2.3 Details of Hardware Used

2.4 Details of Software Used

2.5 Overview of Front-End

2.6 Overview of Back-End

3. SYSTEM ANALYSIS*…………………………………………………………..17*

3.1 Entity Relationship (ER) Diagram

3.2 Data Flow Diagram

3.3 Physical and Logical structure

4. SYSTEM DESIGN AND IMPLEMENTATION*…………….………… 21*

4.1 User Interface and approach

5. CODING SYSTEM*…………………………………………………..…… 25*

***6. USER'S GUIDE**…………………………………………………………….………. 34*

6.1 Installation Guide

6.2 User operating instruction

***7. LIMITATIONS AND DRAWBACKS OF THE PROJECT**………… 36*

***8. FUTURE ENHANCEMENT**……………………………………..……………... 37*

8.1 Future Implementations and Advantages

***9. CONCLUSION**……………………………..……………………………..….…...38*

1. INTRODUCTION

In recent years, the use of Arduino and smartphone technologies has become increasingly popular in the field of electronics and automation. In this project, we explore the combination of these technologies to build an Arduino remote controlling car that can be controlled via a smartphone app.

This project record book provides a step-by-step guide on building an Arduino remote controlling car using a smartphone. It includes information on the materials and equipment required, the methodology for building the car, the programming of the Arduino and the developing method of the smartphone app.

1.1 Overview of the Project

This project aims at developing multi-function car using the Arduino platform. That will be able control using a smartphone app that communicates with the Arduino via Bluetooth instead of using remote controller which will be much easier for user who like this kind of stuff, this will include obstacle avoidance, Bluetooth control and voice control function. It mainly uses an ultrasonic sensor and Bluetooth module.

1.2 Objectives of the project

1. *To design and build a remote-controlled car that can be controlled via a smartphone app.*
2. *To demonstrate the capabilities of Arduino and smartphone technology to create a low-cost and efficient remote-controlled vehicle.*
3. *To provide a step-by-step guide on building an Arduino remote controlling car using a smartphone.*
4. *To inspire others to explore the possibilities of Arduino and smartphone technology in their own projects*

1.3Scope of the project

1. *Designing and building a remote-controlled car from scratch.*
2. *Programming the Arduino microcontroller to receive commands from the smartphone app and control the motors.*
3. *Creating a smartphone app to communicate with the Arduino via Bluetooth.*
4. *Testing and debugging the system to ensure it works as intended.*
5. *Documenting the project process and providing a comprehensive guide on building an Arduino remote controlling car using a smartphone.*

2. SYSTEM REQUIREMENTS

2.1 Hardware Requirements

Following are the hardware requirement for creating Arduino remote controlling car:

1.Arduino UNO

2. Motor driver

3.Gear motor

4. Wheels

5.Battery

6.Bluetooth Module

7.Jumper wires

8.Chassis

9.Ultrsonic Sensor

10.Servo Motor

11.Smartphone

2.2Software Requirements

The software requirements that are used for inserting the code to the Arduino board and for designing the User Interface (Front-end) and the database (Back-end) are as follows:

(i)Arduino IDE
(ii)MIT App Inventor
(iii)Firebase

2.3 Details of Hardware used

1) *Arduino UNO Board: You will need an Arduino board to act as the main controller for the car. The most common boards used for this project are the Arduino Uno or Arduino Nano.*
2) *Motor Driver: A motor driver is necessary to control the speed and direction of the motors. L298N motor driver is a commonly used driver for this project.*
3) *Gear Motors: You will need four motors to power the car. The motor speed and direction will be controlled by the motor driver.*
4) *Wheels: You will four wheels to attach to the gear motors. You can purchase pre-made wheels or create custom ones.*
5) *Battery: A battery is required to power the motors and Arduino board. A rechargeable lithium-ion battery is commonly used in this project.*
6) *Bluetooth Module: You will need a Bluetooth module to communicate between the Arduino and the smartphone app. HC-05 Bluetooth module is a commonly used module for this project.*
7) *Jumper Wires: You will need a variety of jumper wires to connect the electronic components together.*
8) *Chassis: You will need a chassis to hold all the electronic components and create the body of the car. You can purchase pre-made chassis or create custom ones.*
9) *Ultrasonic Sensor: You will need an Ultrasonic sensor for detecting object that are on the way.*
10) *Servo Motor: you will need servo motor for rotating ultrasonic sensor. Smartphone: You will need a smartphone with Bluetooth capability to control the car via the smartphone app.*

2.4 Details of Software Used

***(i)Arduino IDE**: Arduino IDE is an open-source electronics platform that is designed to make it easy for anyone to create interactive projects. The Arduino IDE (Integrated Development Environment) is an application used to write, compile, and upload code to Arduino boards.*

With the Arduino IDE, users can create programs using a programming language that is similar to C++. The IDE provides a user-friendly interface that allows users to write code, edit, and debug their programs. The code is written in the form of sketches, which are essentially programs that control the behaviour of the Arduino board.

***(ii)MIT App Inventor**: MIT App Inventor is an online, cloud-based software tool that allows people to create application for*
smartphone.

The platform features a graphical drag-and-drop interface that enables users to create apps by simply dragging components onto a visual design screen and then programming them with blocks of code that are similar to puzzle pieces.

***(iii)Firebase:** Firebase is a comprehensive mobile and web application development platform owned by Google. It offers a wide range of tools and services that help developers build, improve, and manage their applications efficiently.*

Here are some key components and features of Firebase:

1. *Realtime Database: Firebase provides a cloud-hosted NoSQL database that allows developers to store and sync data in real-time across clients. It enables building collaborative and interactive applications.*

2. *Authentication: Firebase offers an authentication service that simplifies user management and authentication processes. It supports various*

authentication methods like email/password, social login (Google, Facebook, Twitter), and more.

3. *Cloud Firestore: Firestore is a flexible and scalable NoSQL database provided by Firebase. It offers seamless data synchronization, offline capabilities, and real-time updates. Firestore allows developers to structure data hierarchically and query it efficiently.*

4. *Cloud Storage: Firebase provides a secure and scalable cloud storage solution for storing user-generated content such as images, videos, and files. It simplifies the process of uploading, downloading, and managing files from the cloud.*

2.5 Overview of Front-End:

MIT App Inventor: *MIT App Inventor is a visual development platform that allows users, including beginners and non-programmers, to create mobile applications for Android devices. It simplifies the process of building mobile apps by providing a visual interface and drag-and-drop blocks-based programming. Here's an overview of MIT App Inventor:*

1. *Visual Development Environment: MIT App Inventor provides a web-based development environment where users can create their mobile apps using a visual interface. It offers a palette of components and blocks that can be assembled to build the desired functionality.*

2. *Drag-and-Drop Programming: Instead of writing code manually, users can build the logic of their mobile apps by dragging and dropping blocks onto the workspace and configuring their properties. The blocks correspond to different app components and actions, allowing users to create interactive behaviors without coding.*

3. *Components and Events: MIT App Inventor offers a variety of components that users can add to their apps, such as buttons, labels, textboxes, images, sensors (like GPS or accelerometer), and more. These components have associated events, such as button clicks or sensor data changes, which trigger specific actions or behaviors.*

4. *Block-Based Programming: Each component and event in MIT App Inventor has corresponding blocks with predefined functions and properties. Users can visually connect these blocks to create the desired app logic. The blocks cover a wide range of functionalities, including user input, data storage, media handling, communication, and more.*

5. *Real-Time Testing: MIT App Inventor provides a companion app that can be installed on Android devices. Users can connect their devices to the development environment, and as they build the app, they can instantly test*

and see the changes on their device without the need for compiling or deploying.

6. *App Deployment: Once the app development is complete, MIT App Inventor allows users to package their apps as APK (Android Package) files. These files can be installed and run on Android devices or published to the Google Play Store for distribution to a wider audience.*

2.6 Overview of the Back-End:

Firebase: *Firebase is a set of tools offered by Google to build excellent scalable applications in the cloud. It is a powerful service that helps in building applications quickly without reinventing the components or modules. Services like analytics, authentication, databases, configuration, file storage, push messaging are provided by firebase thereby making it easy for the developers to focus on the user experience of the application. Firebase allow syncing the real time data across all the devices-Android,IOS,and the web without refreshing the screen Firebase offers integration to Google Ads, AdMob, DoubleClick, Play Store, Data Studio, BigQuery, and Slack,to make you're app development with efficient and accurate management and maintenance Everything from the Database, analytics to crashing reports are included in Firebase. So the app development teams can stay focus on improving the user experience.*

3. SYSTEM ANALYSIS

3.1 Entity Relationship (ER) Diagram

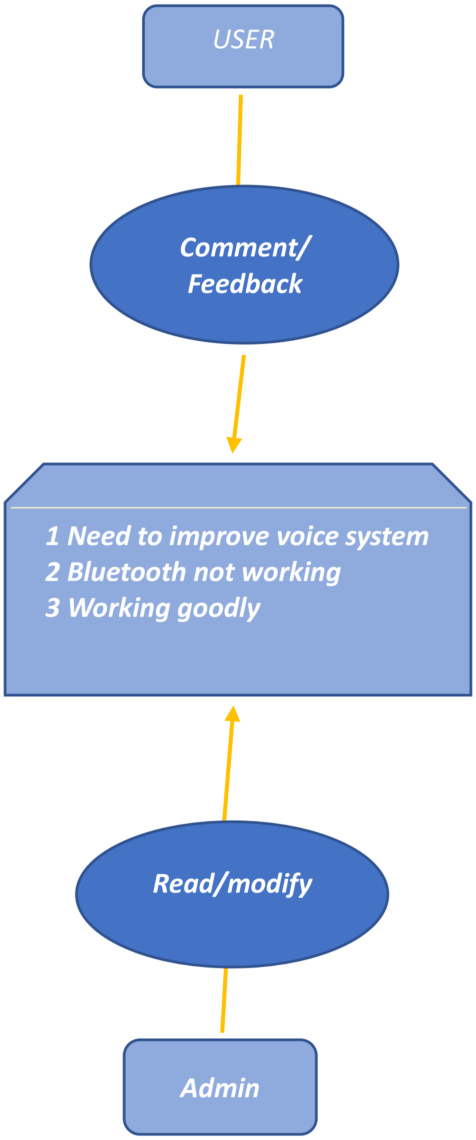


Fig.1 Entity Relationship Diagram

3.2 Data Flow Diagram (DFD)

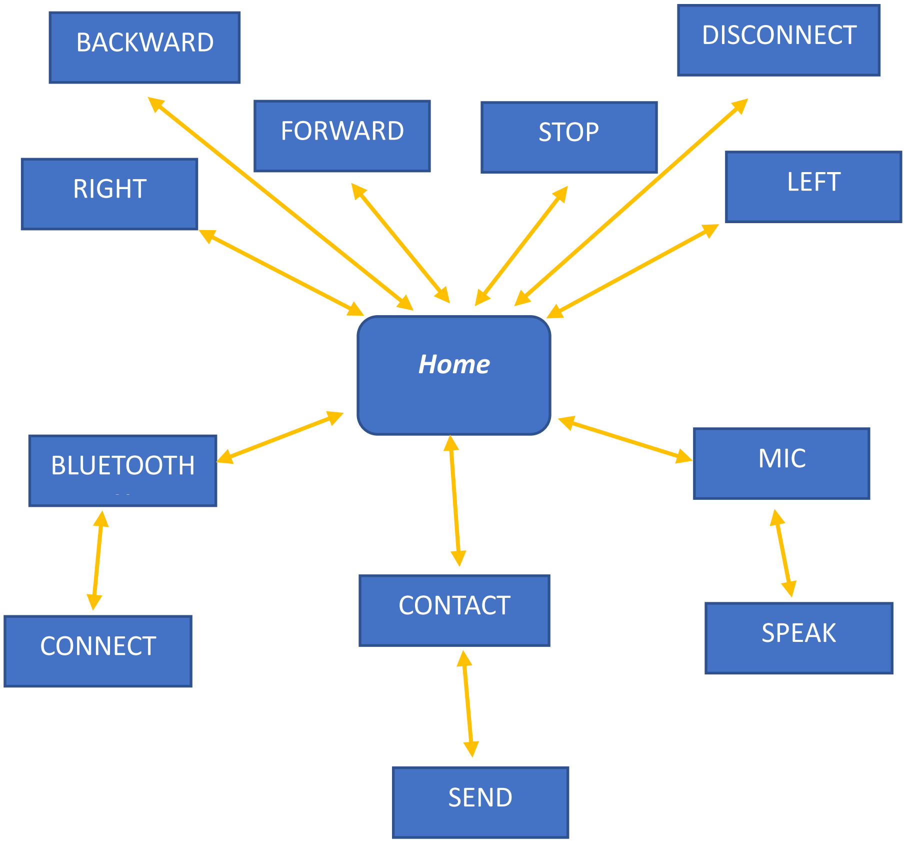
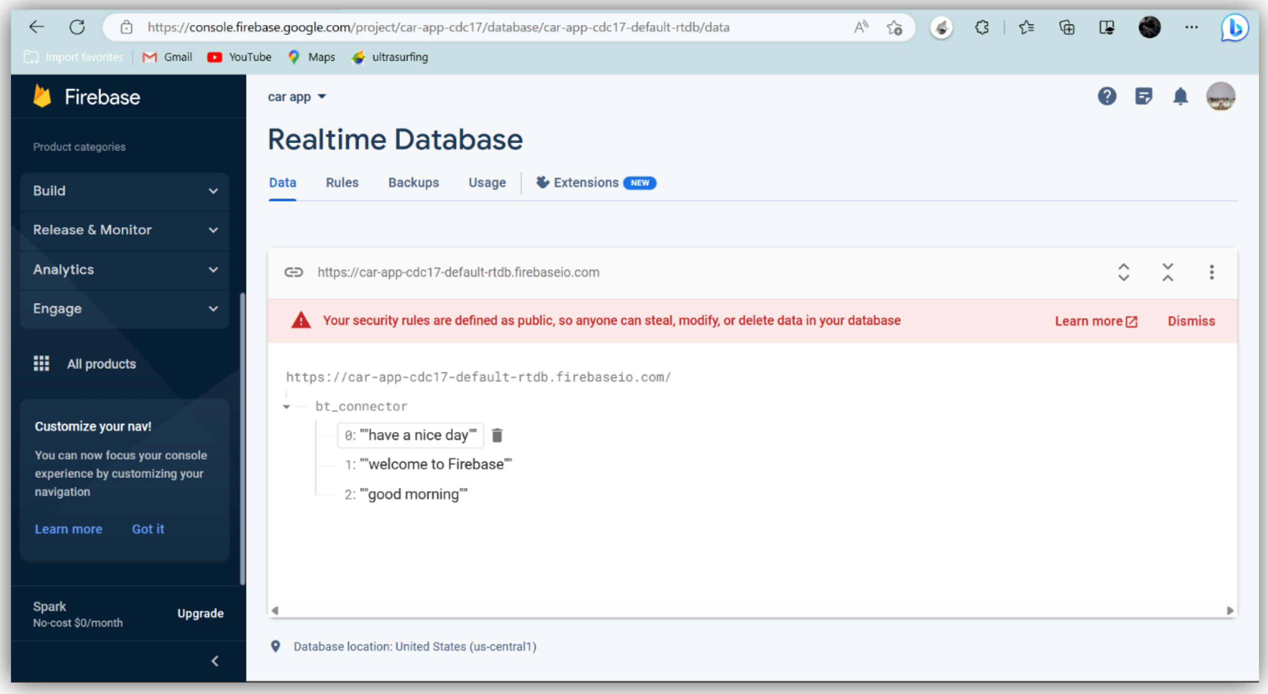


Fig.2 Data Flow Diagram

3.3Physical and Logical Structure

Database Design: *The back-end of the project is created using Firebase Realtime Database**.** When user send data through the application, it will be received and stored in realtime database.*

Firebase Realtime Database:



https://console.firebase.google.com/project/car-app-cdc17/database/car-app-cdc17-default-rtdb/data

https://car-app-cdc17-default-rtdb.firebaseio.com

Your security rules are defined as public, so anyone can steal, modify, or delete data in your database

Customize your nav!

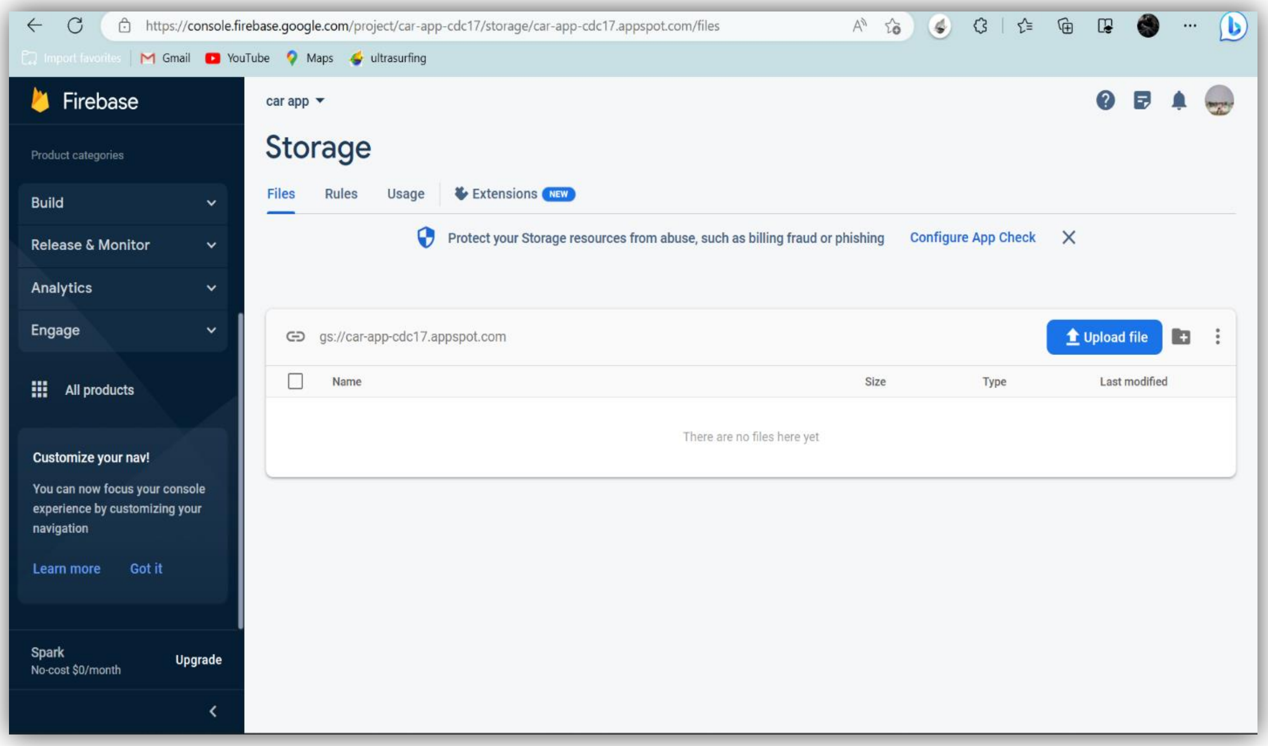
You can now focus your console experience by customizing your navigation

```
https://car-app-cdc17-default-rtdb.firebaseio.com/
▾ bt_connector
    0: ""have a nice day""
    1: ""welcome to Firebase""
    2: ""good morning""
```

Database location: United States (us-central1)

Fig.3 Database

Firebase Storage:



You can now focus your console experience by customizing your navigation

Protect your Storage resources from abuse, such as billing fraud or phishing

There are no files here yet

Fig.4 Storage

4. SYSTEM DESIGN AND IMPLEMENTATION

4.1 User Interface and approach

1) HOME: *Home Screen is the page where user can know certain Details of how the car will be connected and control.*

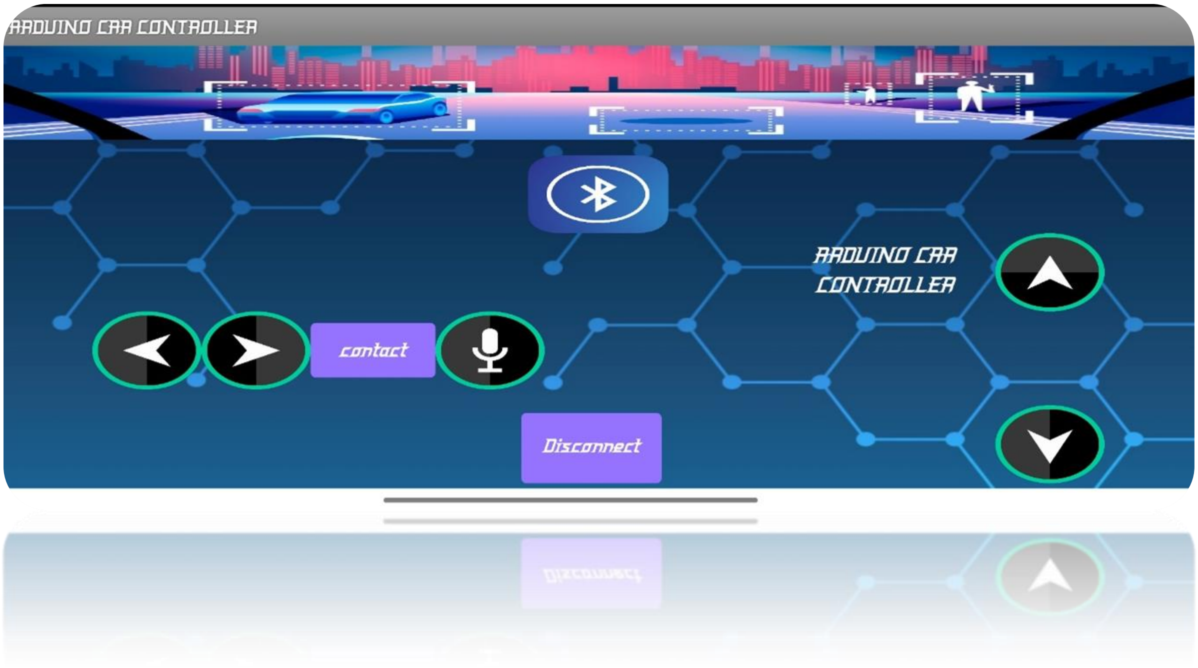
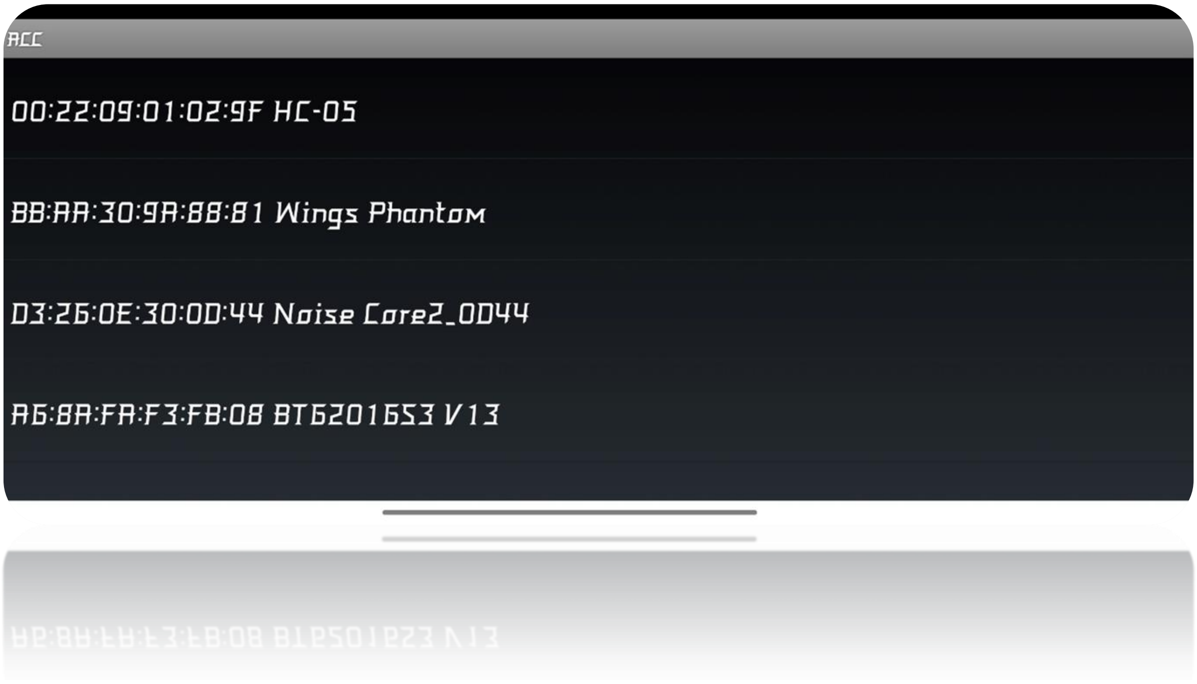


Fig.5 Home

2) BLUETOOTH: *BY using this button we will be able to connect to the car. It will list out all the paired device and we will connect it.*



ACC

00:22:09:01:02:9F HC-05

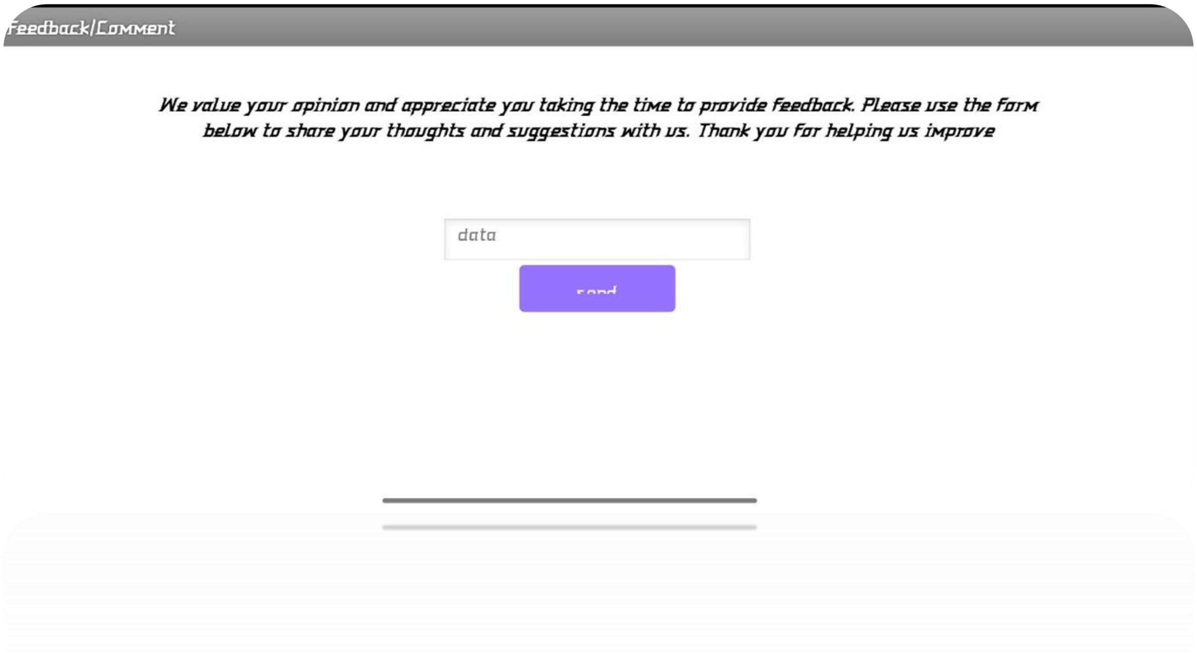
BB:AA:30:9A:88:81 Wings Phantom

D3:26:0E:30:0D:44 Noise Core2_0D44

A6:8A:FA:F3:FB:08 BT6201653 V13

Fig.6 Bluetooth

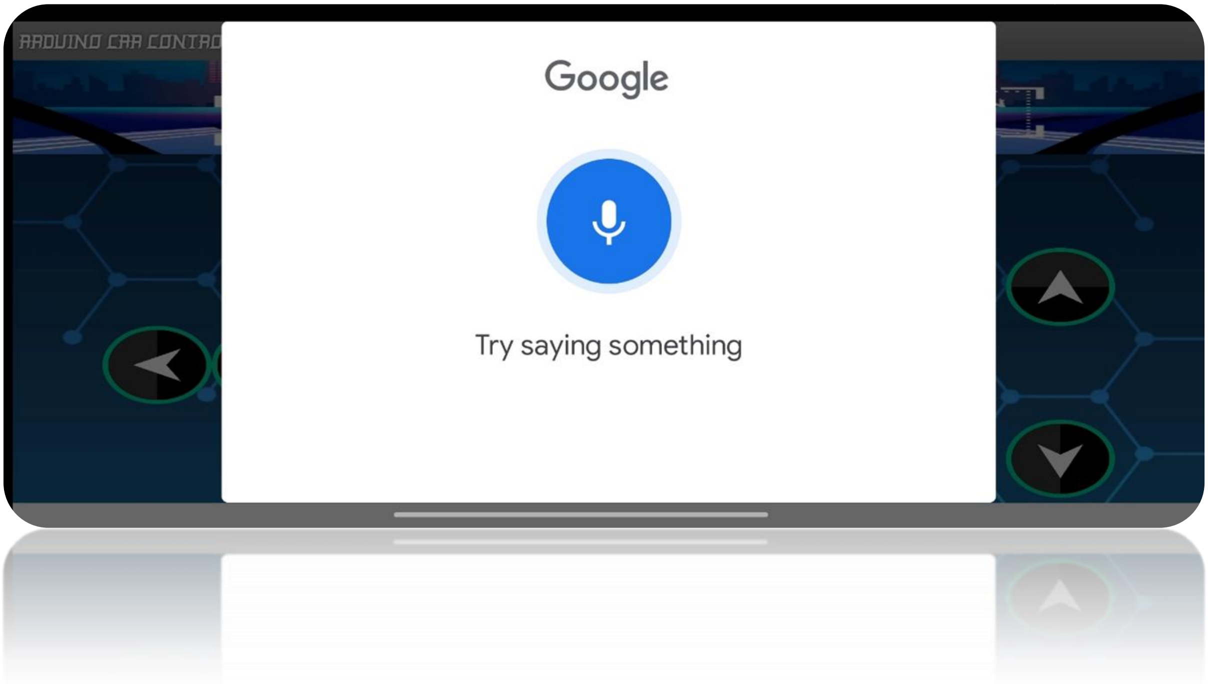
3) CONTACT: *This will lead to a new page, provide a textbox and through that the user will be able to send comment or feedback.*



We value your opinion and appreciate you taking the time to provide feedback. Please use the form below to share your thoughts and suggestions with us. Thank you for helping us improve

Fig.7 CONTACT

4) MIC : *This button will call your device speech recognizer, when you give command it will transform the speech into text.*



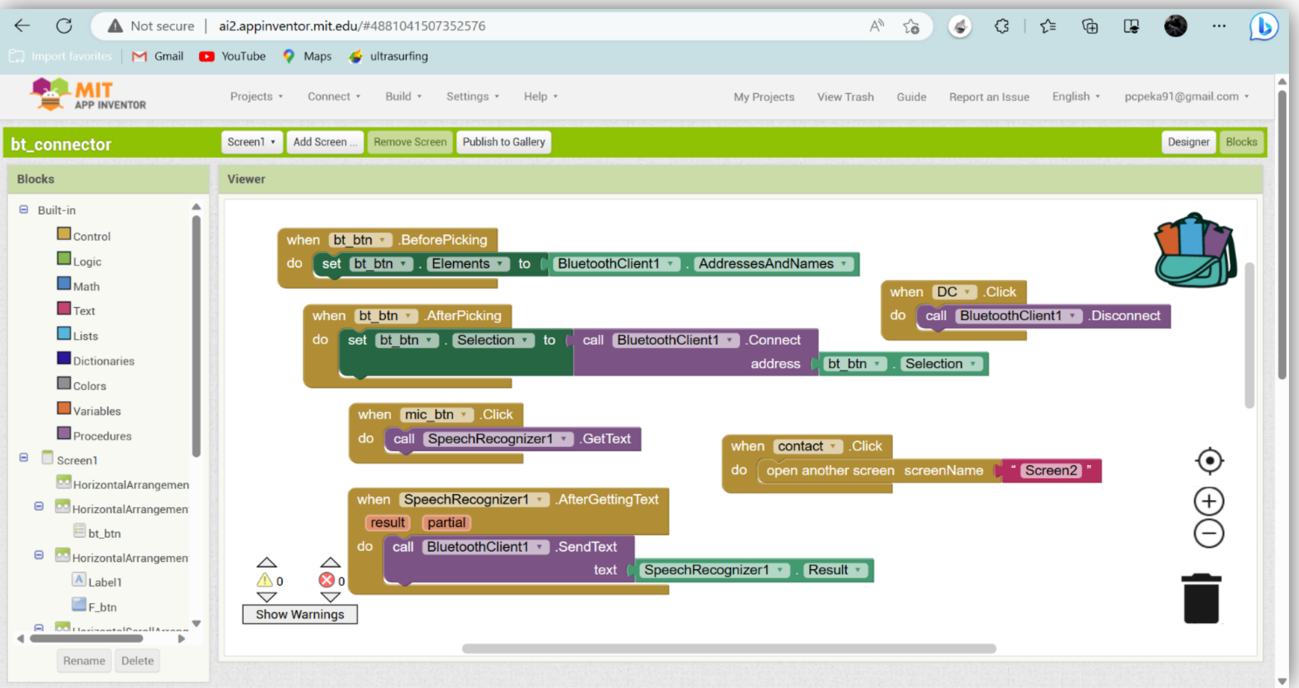
Try saying something

Fig.8 MIC

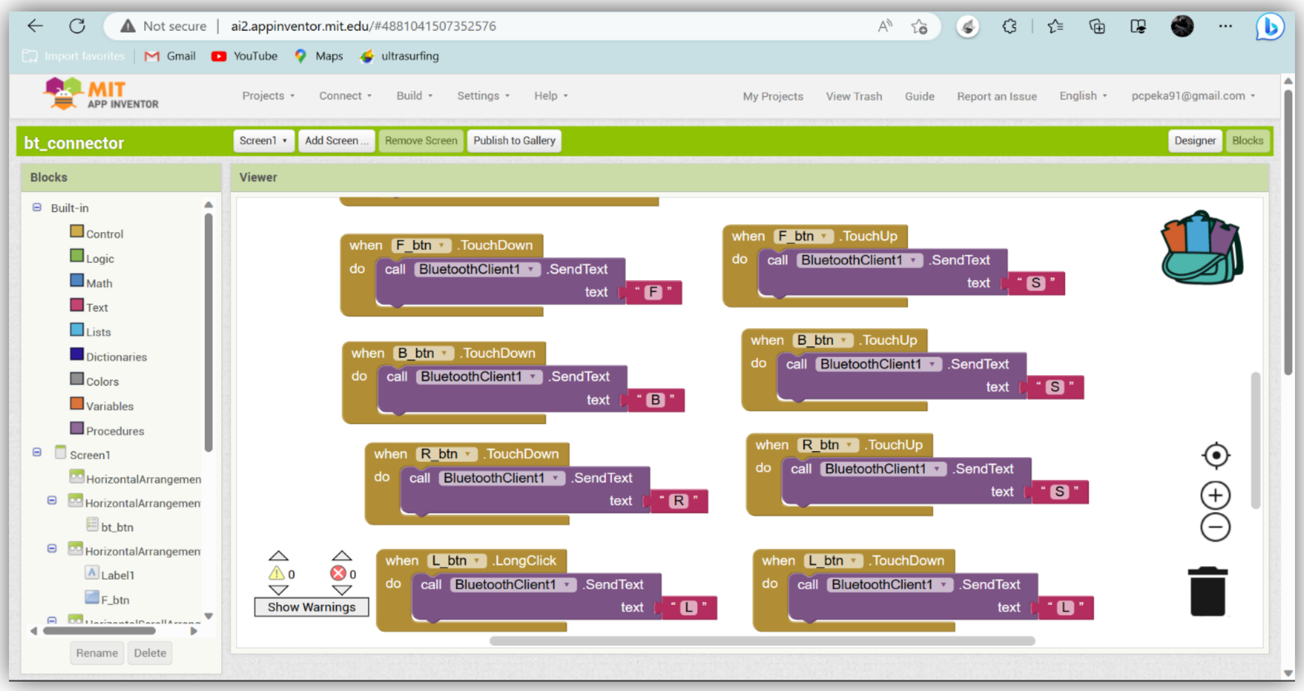
5. CODING SYSTEM

As the front-end is created using MIT App Inventor, It simplifies the process of building mobile apps by providing a visual interface and drag-and-drop blocks-based programming. The blocks codes used in the project are given below:

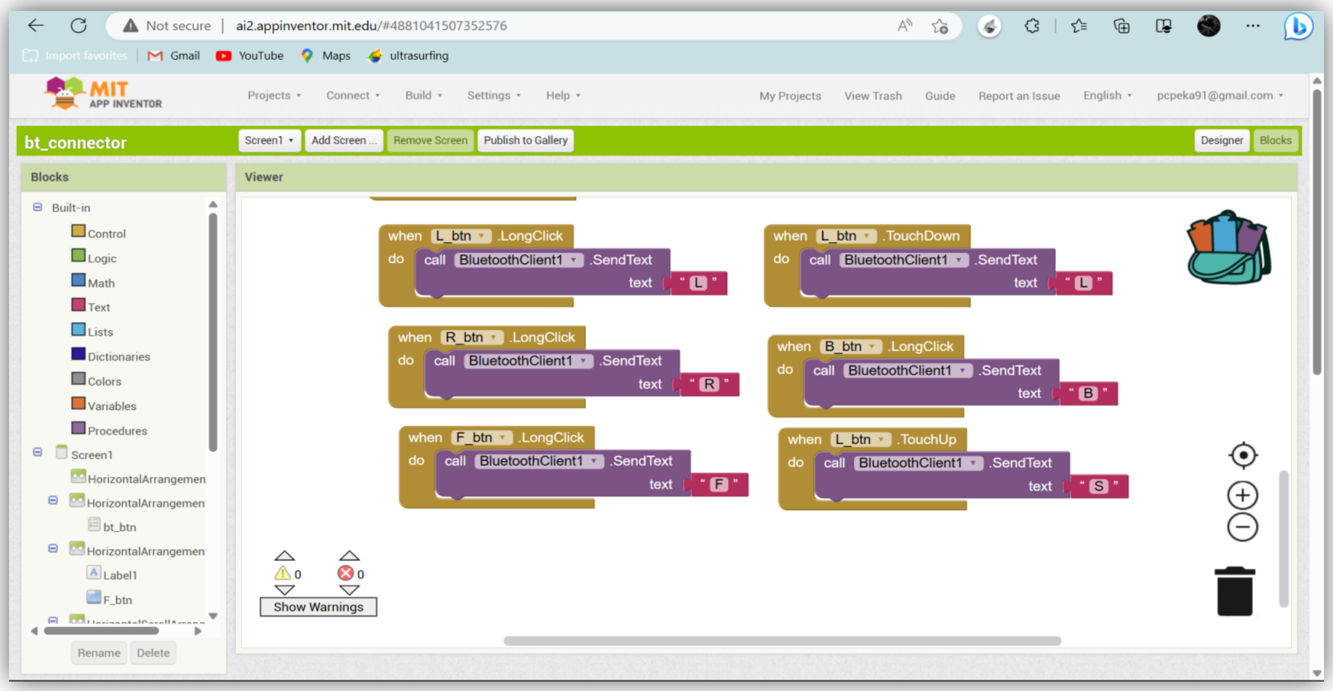
1) Block codes for Home:



```
when bt_btn .BeforePicking
do  set bt_btn . Elements to  BluetoothClient1 . AddressesAndNames

when bt_btn .AfterPicking
do  set bt_btn . Selection to  call BluetoothClient1 .Connect
                                                address  bt_btn . Selection

when DC .Click
do  call BluetoothClient1 .Disconnect

when mic_btn .Click
do  call SpeechRecognizer1 .GetText

when contact .Click
do  open another screen  screenName  " Screen2 "

when SpeechRecognizer1 .AfterGettingText
 result  partial
do  call BluetoothClient1 .SendText
                           text  SpeechRecognizer1 . Result
```

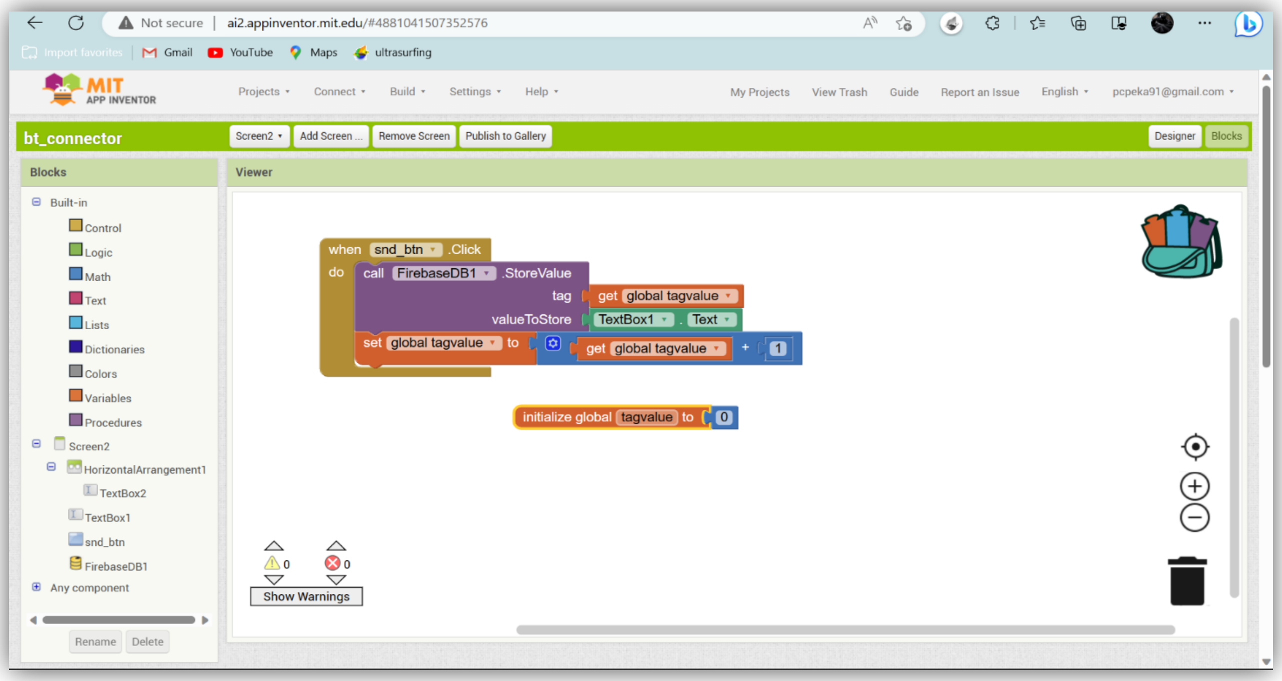


```
when F_btn .TouchDown
do  call BluetoothClient1 .SendText
                          text " F "

when F_btn .TouchUp
do  call BluetoothClient1 .SendText
                          text " S "

when B_btn .TouchDown
do  call BluetoothClient1 .SendText
                          text " B "

when B_btn .TouchUp
do  call BluetoothClient1 .SendText
                          text " S "

when R_btn .TouchDown
do  call BluetoothClient1 .SendText
                          text " R "

when R_btn .TouchUp
do  call BluetoothClient1 .SendText
                          text " S "

when L_btn .LongClick
do  call BluetoothClient1 .SendText
                          text " L "

when L_btn .TouchDown
do  call BluetoothClient1 .SendText
                          text " L "
```



```
when L_btn .LongClick
do  call BluetoothClient1 .SendText
                          text " L "

when L_btn .TouchDown
do  call BluetoothClient1 .SendText
                          text " L "

when R_btn .LongClick
do  call BluetoothClient1 .SendText
                          text " R "

when B_btn .LongClick
do  call BluetoothClient1 .SendText
                          text " B "

when F_btn .LongClick
do  call BluetoothClient1 .SendText
                          text " F "

when L_btn .TouchUp
do  call BluetoothClient1 .SendText
                          text " S "
```

1) Block codes for sending data (Feedback/Comment)



```
when snd_btn .Click
do  call FirebaseDB1 .StoreValue
                          tag  get global tagvalue
                 valueToStore  TextBox1 . Text
    set global tagvalue to  get global tagvalue + 1

initialize global tagvalue to 0
```

Arduino IDE is also used for creating the code that are inserted or uploaded in the Arduino board/car. The language used to create the program is c++. The code used are given below:

1) code for creating libraries:

```
#include <Servo.h>
#include <AFMotor.h>
```

2)Code for ultrasonic sensor pins, servo motor pin, motor speed, and servo motor starting point :

```
#define Echo A0
#define Trig A1
#define motor 10
#define Speed 170
#define spoint 103
```

3)Code for creating variables:

```
char value;
int distance;
int Left;
int Right;
int L = 0;
int R = 0;
int L1 = 0;
int R1 = 0;
```

4)Code for creating objects:

```
        Servo servo;
AF_DCMotor M1(1);
AF_DCMotor M2(2);
AF_DCMotor M3(3);
AF_DCMotor M4(4);
```

5)Code for creating setup function:

```
void setup() {
  Serial.begin(9600);
  pinMode(Trig, OUTPUT);
  pinMode(Echo, INPUT);
  servo.attach(motor);
  M1.setSpeed(Speed);
  M2.setSpeed(Speed);
  M3.setSpeed(Speed);
  M4.setSpeed(Speed);
}
```

6)Code for creating loop function:

```
void loop() {
  //Obstacle();
  //Bluetoothcontrol();
  //voicecontrol();
}
```

7)Code for creating Bluetooth control:

```
void Bluetoothcontrol() {
  if (Serial.available() > 0) {
    value = Serial.read();
    Serial.println(value);
  }
  if (value == 'F') {
    forward();
  } else if (value == 'B') {
    backward();
  } else if (value == 'L') {
    left();
  } else if (value == 'R') {
    right();
  } else if (value == 'S') {
    Stop();
  }
}
```

8)Code for creating obstacle-avoiding:

```
void Obstacle() {
  distance = ultrasonic();
  if (distance <= 12) {
    Stop();
    backward();
    delay(100);
    Stop();
    L = leftsee();
    servo.write(spoint);
    delay(800);
    R = rightsee();
    servo.write(spoint);
    if (L < R) {
      right();
      delay(500);
      Stop();
      delay(200);
    } else if (L > R) {
      left();
      delay(500);
      Stop();
      delay(200);
    }
  } else {
    forward();
  }
}
```

9)Code for creating voice control:

```
void voicecontrol() {
  if (Serial.available() > 0) {
    value = Serial.read();
    Serial.println(value);
    if (value == 'Forward') {
      forward();
    } else if (value == 'Backward') {
      backward();
    } else if (value == 'Left') {
      L = leftsee();
      servo.write(spoint);
      if (L >= 10 ) {
        left();
        delay(500);
        Stop();
      } else if (L < 10) {
        Stop();
      }
    } else if (value == 'Right') {
      R = rightsee();
      servo.write(spoint);
      if (R >= 10 ) {
        right();
        delay(500);
        Stop();
      } else if (R < 10) {
        Stop();
      }
    } else if (value == '*') {
      Stop();
    }
  }
}
```

10)Code for ultrasonic sensor distance reading function:

```
int ultrasonic() {
  digitalWrite(Trig, LOW);
  delayMicroseconds(4);
  digitalWrite(Trig, HIGH);
  delayMicroseconds(10);
  digitalWrite(Trig, LOW);
  long t = pulseIn(Echo, HIGH);
  long cm = t / 29 / 2; //time convert distance
  return cm;
}
void forward() {
  M1.run(FORWARD);
  M2.run(FORWARD);
  M3.run(FORWARD);
  M4.run(FORWARD);
}
void backward() {
  M1.run(BACKWARD);
  M2.run(BACKWARD);
  M3.run(BACKWARD);
  M4.run(BACKWARD);
}
void right() {
  M1.run(BACKWARD);
  M2.run(BACKWARD);
  M3.run(FORWARD);
  M4.run(FORWARD);
}
void left() {
  M1.run(FORWARD);
  M2.run(FORWARD);
  M3.run(BACKWARD);
  M4.run(BACKWARD);
}
void Stop() {
  M1.run(RELEASE);
  M2.run(RELEASE);
  M3.run(RELEASE);
  M4.run(RELEASE);
}
int rightsee() {
  servo.write(20);
  delay(800);
  Left = ultrasonic();
  return Left;
}
int leftsee() {
  servo.write(180);
  delay(800);
  Right = ultrasonic();
  return Right;
}
```

6. USER'S GUIDE

6.1 Installation Guide

The application can be installed on any android devices by connecting the device with a USB cable to the computer which is used to developed the application or by using the readymade .apk file.

The procedures are as follows:

1) *Connect your android device with a USB cable to the computer which is used to developed the application.*
2) *Create android app bundle(.aab) in MIT App Inventor by clicking in the build button.*
3) *Then you can send the file through USB cable.*
4) *Another way to install the application is by creating android app (.apk) and you can just scan a QR code.*
5) *After the installation process is done you can freely access the application to your device.*

6.2 User operating instructions

After the application is installed successfully, the user can run and access the application whenever he/she need. The basic user operating instructions can be explained from the working of the different buttons.

1) ***Bluetooth button**: This button will display the paired Bluetooth device.*
2) ***Mic button**: This button will call speech recognizer for giving voice control.*
3) ***Contact button:** This button will lead you to a new screen where you can give feedback or comment.*
4) ***Right arrow button**: This button will give command to the car to rotate to right.*
5) ***Left arrow button**: This button will give command to the car to rotate to left.*
6) ***Forward arrow button**: This button will give command to the car to go forward*
7) ***Back arrow button:** This button will give command to the car to go backward*

7. LIMITATIONS AND DRAWBACKS

While Arduino car controlling projects offer an exciting opportunity for learning and experimentation, they also have some limitations and drawbacks that should be considered. Here are a few:

1. *Memory Constraints: Arduino boards typically have limited memory (both RAM and storage), which can pose challenges when working with larger datasets or complex programs. It may require optimization techniques to manage memory usage efficiently.*
2. *Limited I/O Pins: Arduino boards have a limited number of input/output (I/O) pins. Depending on the complexity of the car's functionality, you may need to prioritize and manage the usage of these pins carefully.*
3. *Power Limitations: Arduino boards are powered by a limited amount of voltage and current. Depending on the components used in the car project (such as motors, sensors, and LEDs), it is crucial to ensure that the power supply can handle the required load and that appropriate protective measures are in place to prevent damage.*
4. *Lack of Real-Time Operating System: Arduino boards do not have a real-time operating system (RTOS) by default. This means that timing and synchronization of tasks can be challenging. It may be necessary to implement custom timing mechanisms to ensure precise control and coordination.*
5. *Debugging and Troubleshooting: Identifying and resolving issues or bugs in Arduino car projects can sometimes be challenging, especially for beginners. Debugging techniques and tools, such as serial debugging or using additional debugging hardware, may be required.*

8. FUTURE ENHANCEMENT

The future implementation possibilities for Arduino car controlling projects are vast and exciting. Here are some potential areas where advancements and innovations could be made:

1)Upgrading the communication system between the user and developer.

2)Enhanced Sensor Integration, such as gyroscopes, magnetometers, and environmental sensors. These sensors could provide additional data for improved navigation, environmental monitoring, or enhanced user interaction.

9. CONCLUSION

We know that our project "Arduino Remote controlling Car" has many drawbacks and shortcomings. But on the other hand, when we think how much we give for it we think it deserves a care. This project work is the compilations of our ideas, views and thoughts, we have benefited a great deal from our interaction with our teachers, peoples and friends. We extend our sincere thanks to them.

It was build for the new student, existing student, teacher and parents so that their can easier way of sending information and knowing all the details about Hatim all the important events and the document of the college.

It is a great pleasure for us to express our immense regards to our project guide Mr. H. Lalruatkima and Head of Department Mr. K.Lalmuanpuia for their inspirational guidance who helps us in many ways from the beginning till the end, it is true that only by his help we can complete our project.

We would like to place and record sincere thanks and gratitude to our Principal, Mr Vuansanga Vanchhawng for extending his full support and contentment to this project.

Thought this project may not be the best we hope and pray that our project will find satisfaction in using it. This is a very painstaking work; however, we try our best to satisfy the needs of this project. With many efforts this project has come into being even though we are conscious of our limited knowledge and skills. But it would be our request than this project we have undertaken, be dealt with much consideration and acknowledgement.