ROBOT For CONTROLLING WILDFIRES

Advance Mechatronics
-Term Project-

Shivam Joshi
Deep Trivedi
Yadukrishna BG

WHAT IS THE DIFFERENCE ?

URBAN FIREFIGHTING

Clear APPROACH
Extinguishing METHODS

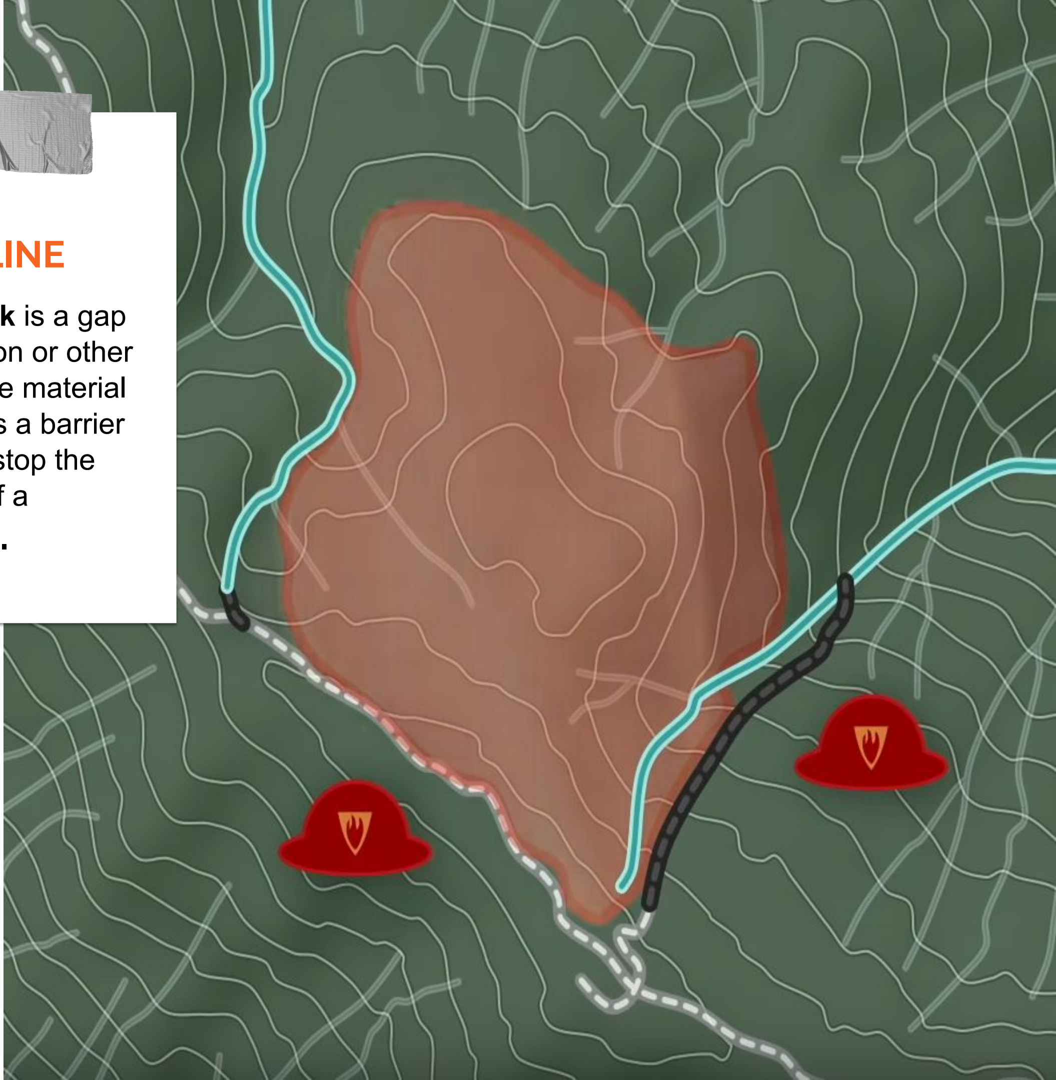
WILDLAND FIREFIGHTING

Gap in Research
Anchor Points | Firelines

Source: https://standard.tv/

FIRELINE

A **Firebreak** is a gap in vegetation or other combustible material that acts as a barrier to slow or stop the progress of a **Wildfire.**



PROBLEM

We're **too slow** and can not go too close to the fire.

PROBLEM

They're **too Expensive**

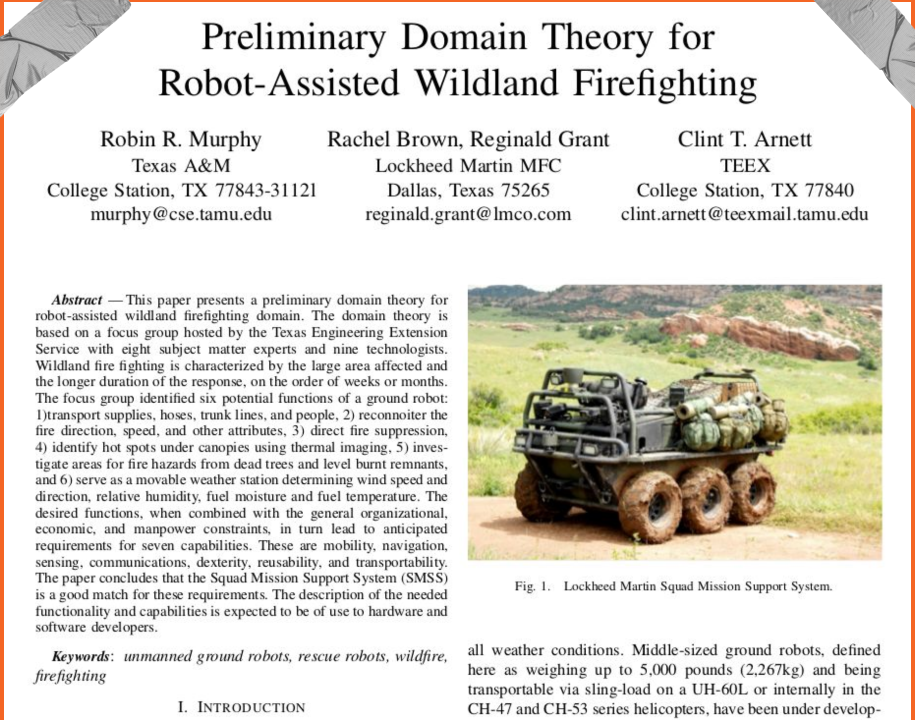
$60K = NYU Fees

1 gallon of Phos Chek

$3

6 FUNCTIONS:

1. Hualing
2. Direct Fire Suppression
3. Mobile Weather Station
4. Reconnoiter
5. **Hot Spot Identification**
6. **Investigate Fire Hazard Zone.**



Preliminary Domain Theory for Robot-Assisted Wildland Firefighting

Robin R. Murphy
Texas A&M
College Station, TX 77843-31121
murphy@cse.tamu.edu

Rachel Brown, Reginald Grant
Lockheed Martin MFC
Dallas, Texas 75265
reginald.grant@lmco.com

Clint T. Arnett
TEEX
College Station, TX 77840
clint.arnett@teexmail.tamu.edu

Abstract — This paper presents a preliminary domain theory for robot-assisted wildland firefighting domain. The domain theory is based on a focus group hosted by the Texas Engineering Extension Service with eight subject matter experts and nine technologists. Wildland fire fighting is characterized by the large area affected and the longer duration of the response, on the order of weeks or months. The focus group identified six potential functions of a ground robot: 1)transport supplies, hoses, trunk lines, and people, 2) reconnoiter the fire direction, speed, and other attributes, 3) direct fire suppression, 4) identify hot spots under canopies using thermal imaging, 5) investigate areas for fire hazards from dead trees and level burnt remnants, and 6) serve as a movable weather station determining wind speed and direction, relative humidity, fuel moisture and fuel temperature. The desired functions, when combined with the general organizational, economic, and manpower constraints, in turn lead to anticipated requirements for seven capabilities. These are mobility, navigation, sensing, communications, dexterity, reusability, and transportability. The paper concludes that the Squad Mission Support System (SMSS) is a good match for these requirements. The description of the needed functionality and capabilities is expected to be of use to hardware and software developers.

Keywords: *unmanned ground robots, rescue robots, wildfire, firefighting*

I. Introduction

Fig. 1. Lockheed Martin Squad Mission Support System.

all weather conditions. Middle-sized ground robots, defined here as weighing up to 5,000 pounds (2,267kg) and being transportable via sling-load on a UH-60L or internally in the CH-47 and CH-53 series helicopters, have been under develop-

THIS PROJECT:

5. Hot Spot Identification
6. Investigate area

ROBOT, DO WHAT?

Mission: build a
fireline

Mobile - Manipulator:

We created a manipulator which can be used to collect material in order to inspect, observe, or survey, and we aim to mount it on top of a mobile base with rocker-bogie which can be used to move through uneven conditions in wildland.

FUN FACT:

At the moment mobile manipulation is a subject of major focus in development and research environments, and mobile manipulators, are used in many areas. Source:https://en.wikipedia.org/wiki/Mobile_manipulator#State_of_the_art

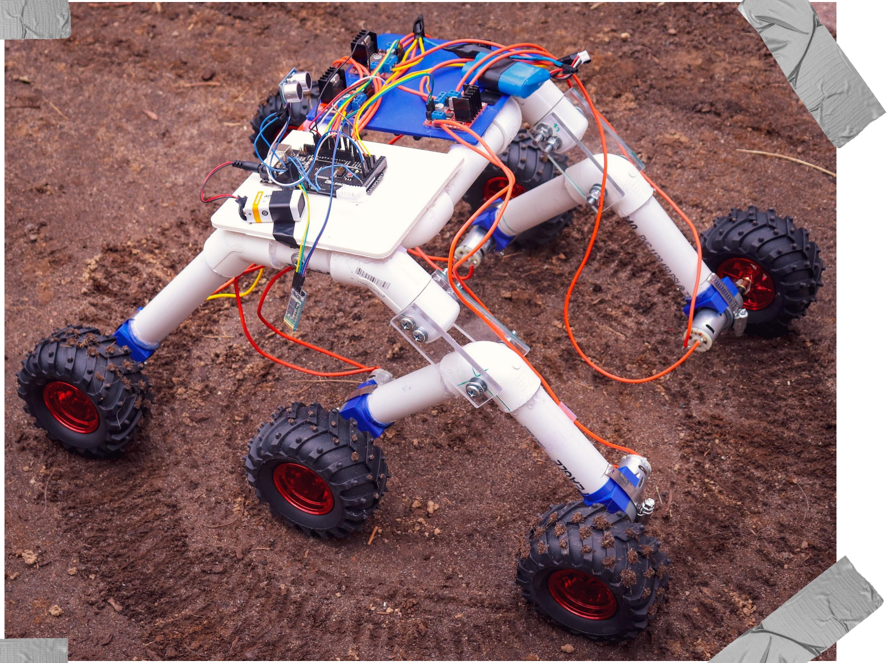
FUN FACT:

The **rocker-bogie** system is NASA's favored design for rovers.

It has been used in multiple mission robots:

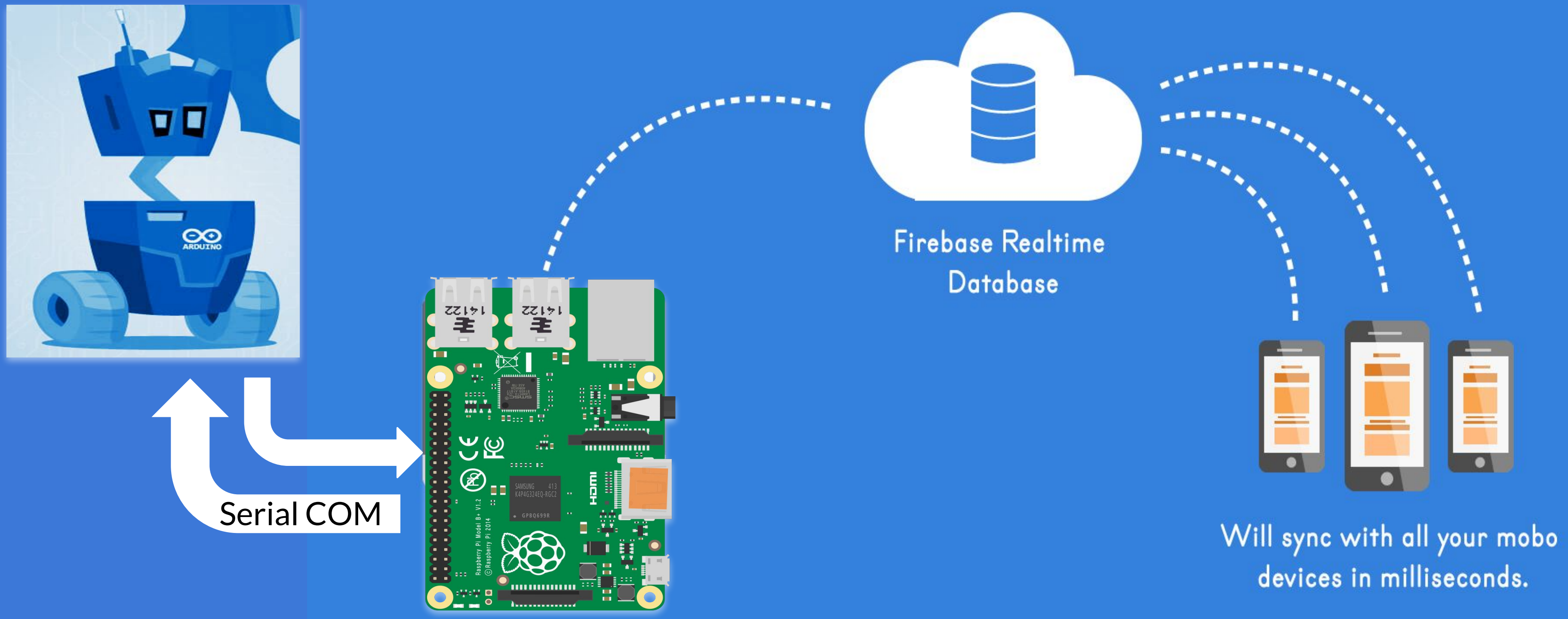
Sojourner, Spirit, Opportunity, and Curiosity

Manipulator



Mobile Base

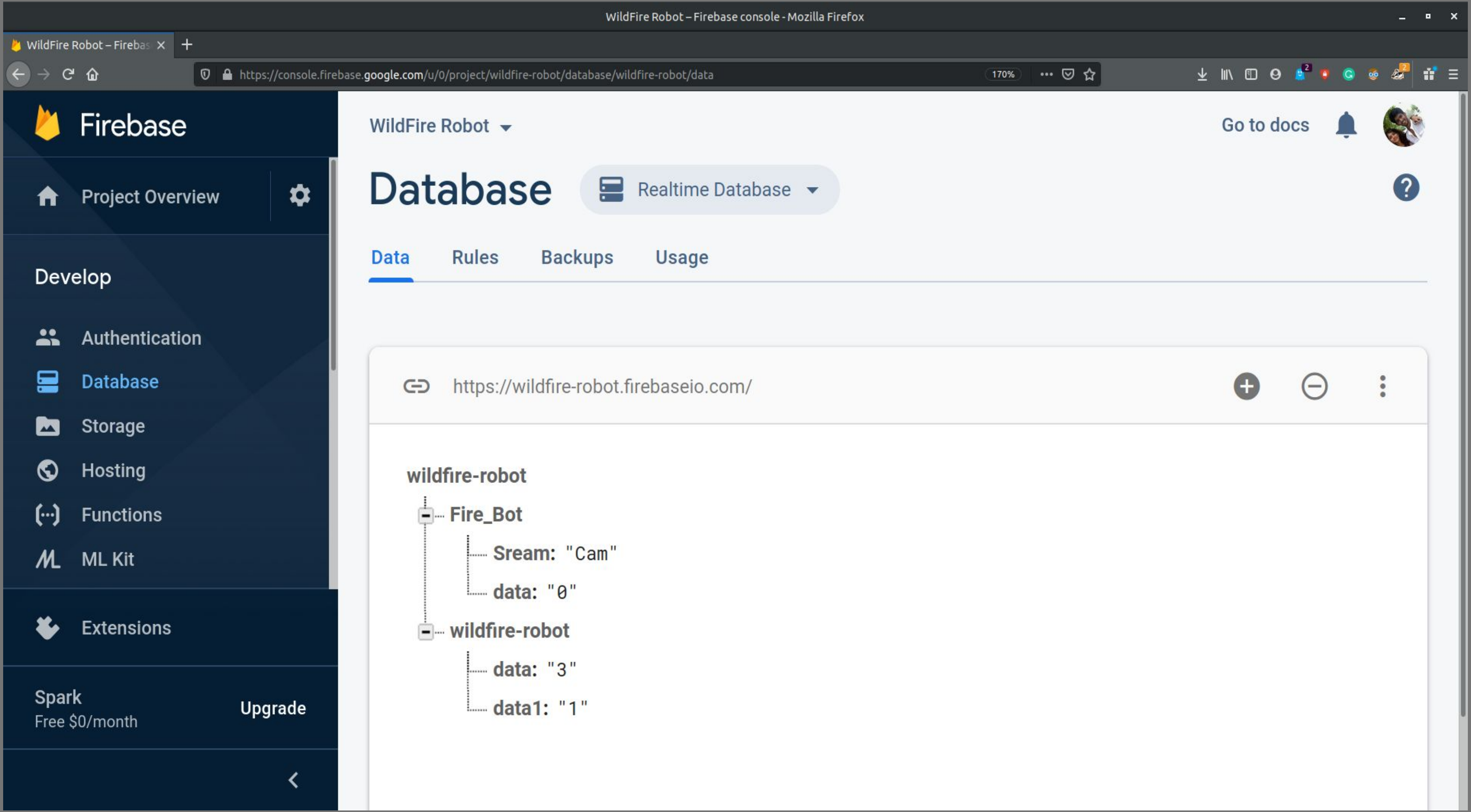
Firebase



Serial COM

Firebase Realtime Database

Will sync with all your mobo devices in milliseconds.



```
wildfire-robot
  Fire_Bot
    Sream: "Cam"
    data: "0"
  wildfire-robot
    data: "3"
    data1: "1"
```

How ?

Hardware:

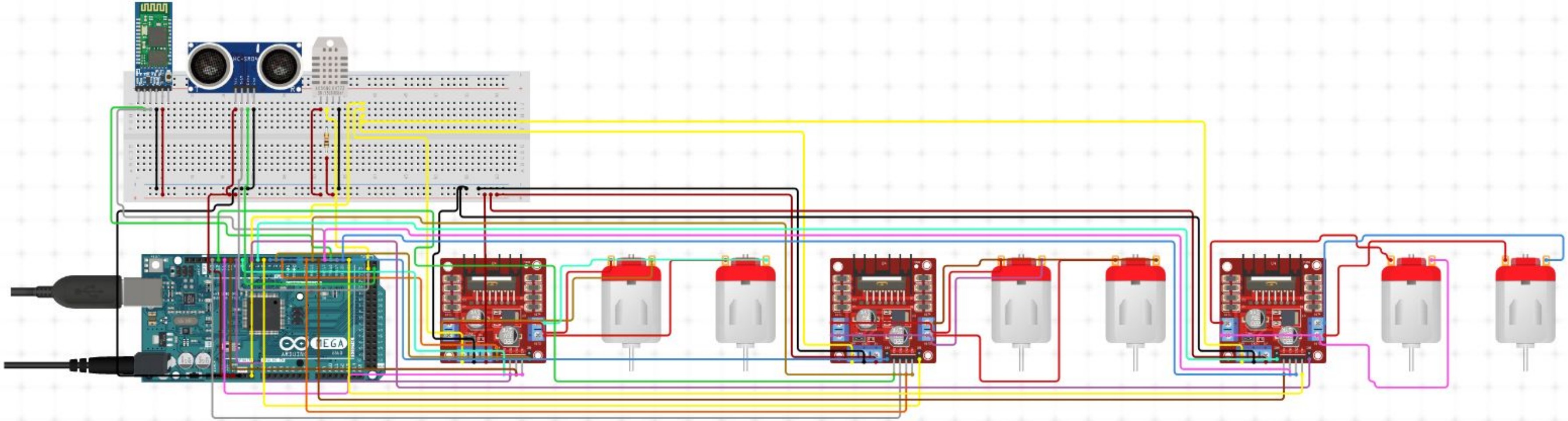
- Raspberry Pi
- Arduino Mega
- Raspberry Pi Camera
- 1 Bluetooth Module (HC-05)
- 3 Motor driver (L298N)
- Li-ion Battery (7.4V 1500mAh)
- Humidity and Temperature Sensor (DHT 22)
- Ultrasonic Sensor (HC - SR04)
- 9V Alkaline Cell
- Metal clamps for links
- Metal Gripper
- 6 DC Motors
- Thumper Wheels
- Clamps, Hex Coupling, etc

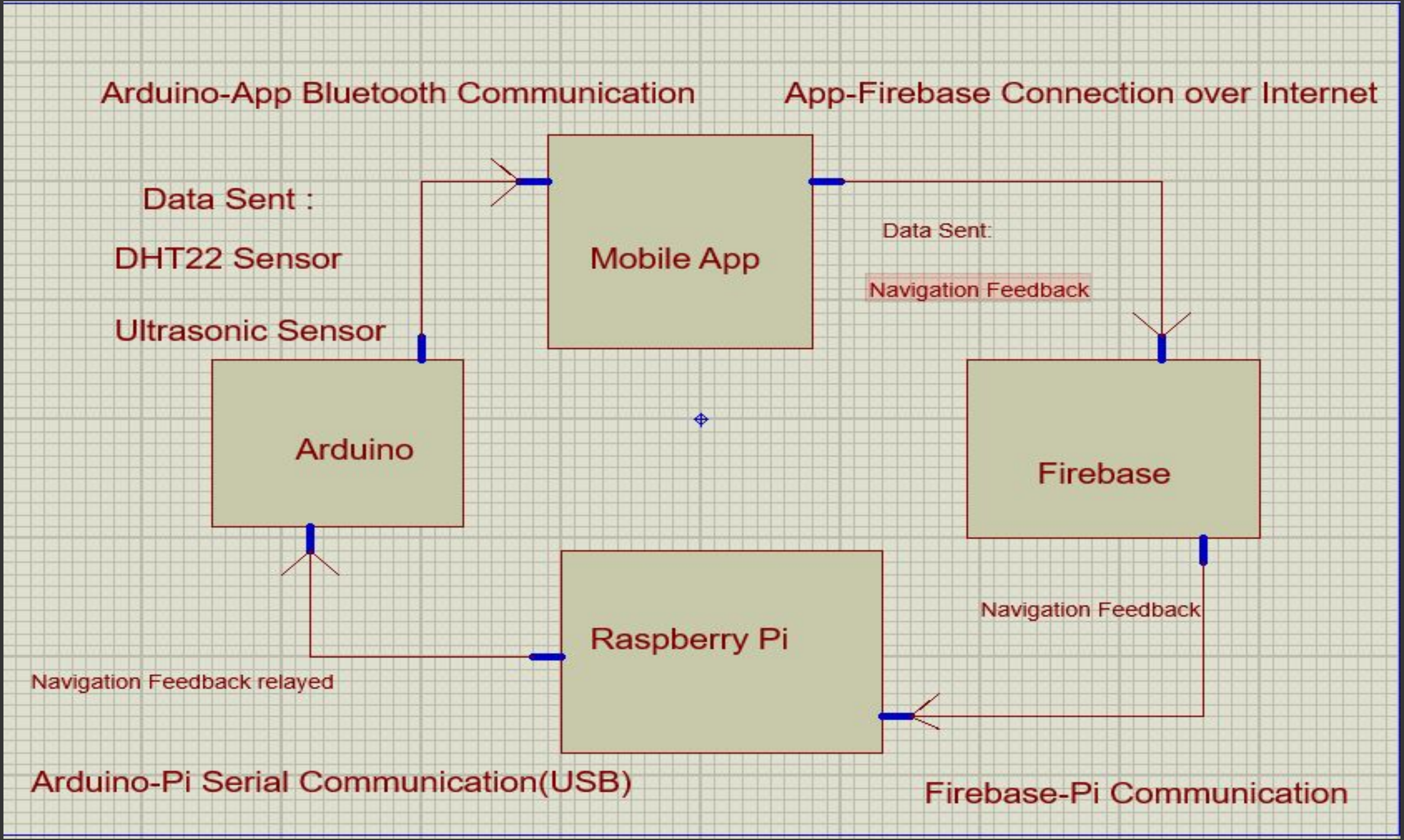
Software:

- MIT App Inventor - Link
- Google Firebase - Link
- RPi_Web_Interface - Link
- Arduino IDE
- Gedit on Raspbian
- Raspbian Shell

(And with little help from our smart phone)

CIRCUIT DIAGRAM:





Arduino-App Bluetooth Communication

App-Firebase Connection over Internet

Data Sent :

DHT22 Sensor

Ultrasonic Sensor

Data Sent:

Navigation Feedback

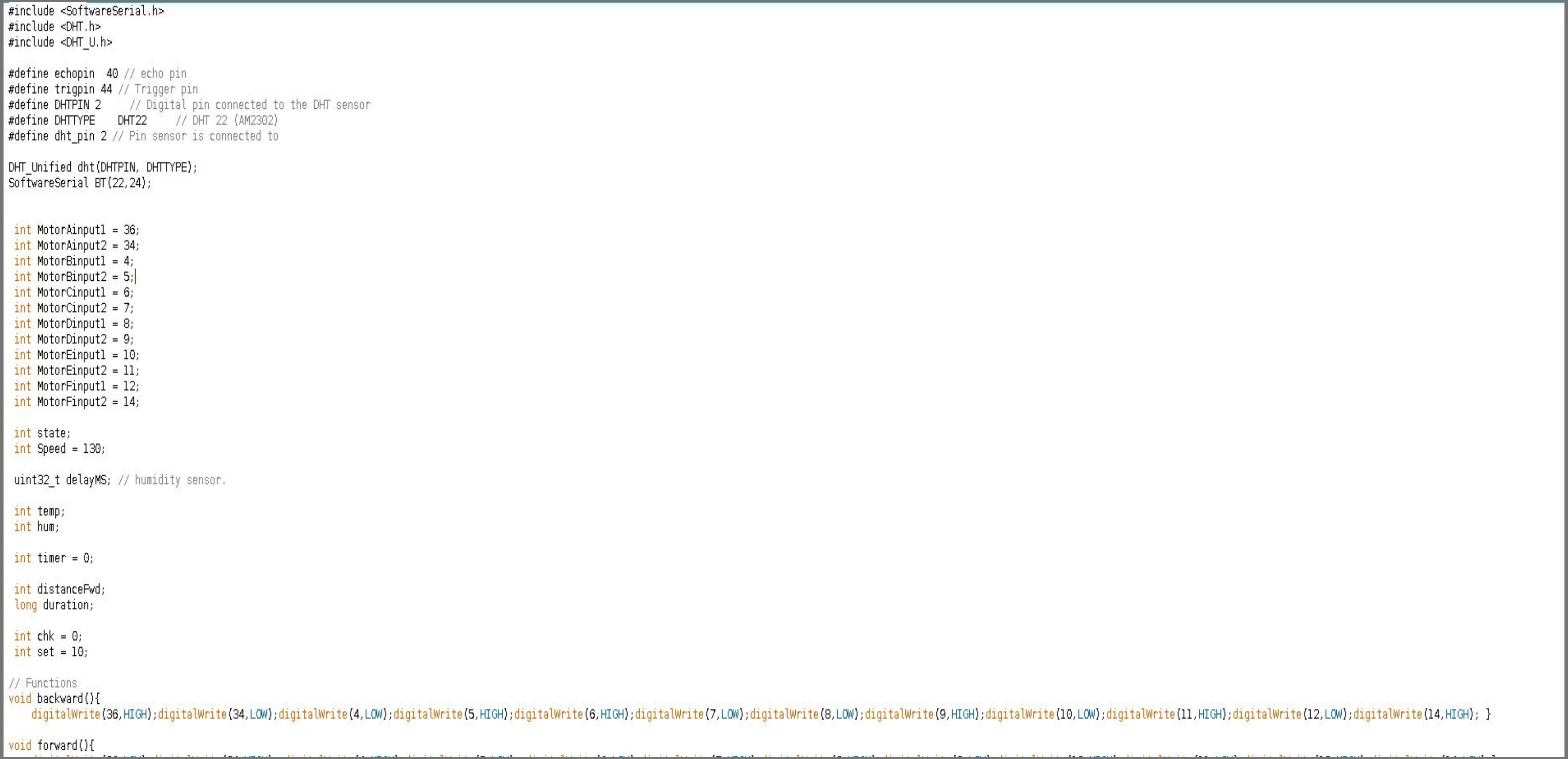
Navigation Feedback

Navigation Feedback relayed

Arduino-Pi Serial Communication(USB)

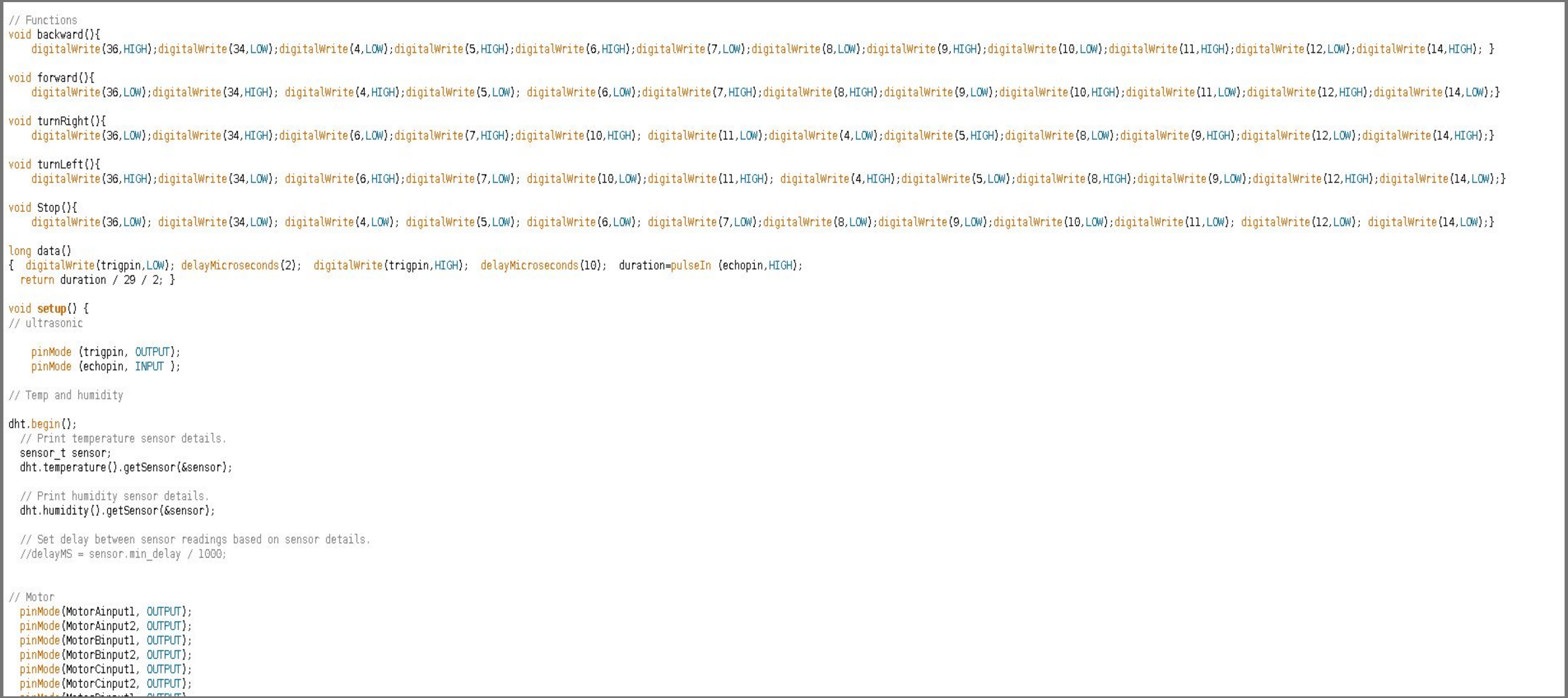
Firebase-Pi Communication

Code1: Arduino



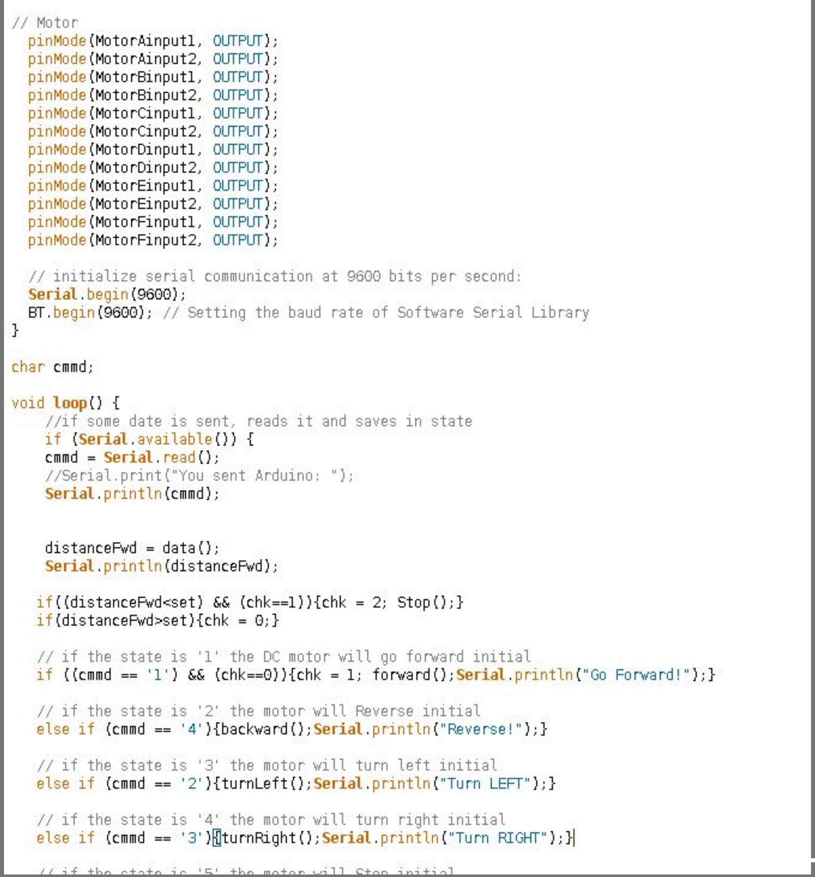
```
#include <SoftwareSerial.h>
#include <DHT.h>
#include <DHT_U.h>

#define echopin  40 // echo pin
#define trigpin 44 // Trigger pin
#define DHTPIN 2     // Digital pin connected to the DHT sensor
#define DHTTYPE    DHT22     // DHT 22 (AM2302)
#define dht_pin 2 // Pin sensor is connected to

DHT_Unified dht(DHTPIN, DHTTYPE);
SoftwareSerial BT(22,24);


 int MotorAinput1 = 36;
 int MotorAinput2 = 34;
 int MotorBinput1 = 4;
 int MotorBinput2 = 5;
 int MotorCinput1 = 6;
 int MotorCinput2 = 7;
 int MotorDinput1 = 8;
 int MotorDinput2 = 9;
 int MotorEinput1 = 10;
 int MotorEinput2 = 11;
 int MotorFinput1 = 12;
 int MotorFinput2 = 14;

 int state;
 int Speed = 130;

 uint32_t delayMS; // humidity sensor.

 int temp;
 int hum;

 int timer = 0;

 int distanceFwd;
 long duration;

 int chk = 0;
 int set = 10;

// Functions
void backward(){
    digitalWrite(36,HIGH);digitalWrite(34,LOW);digitalWrite(4,LOW);digitalWrite(5,HIGH);digitalWrite(6,HIGH);digitalWrite(7,LOW);digitalWrite(8,LOW);digitalWrite(9,HIGH);digitalWrite(10,LOW);digitalWrite(11,HIGH);digitalWrite(12,LOW);digitalWrite(14,HIGH); }

void forward(){
```

Code2: Arduino

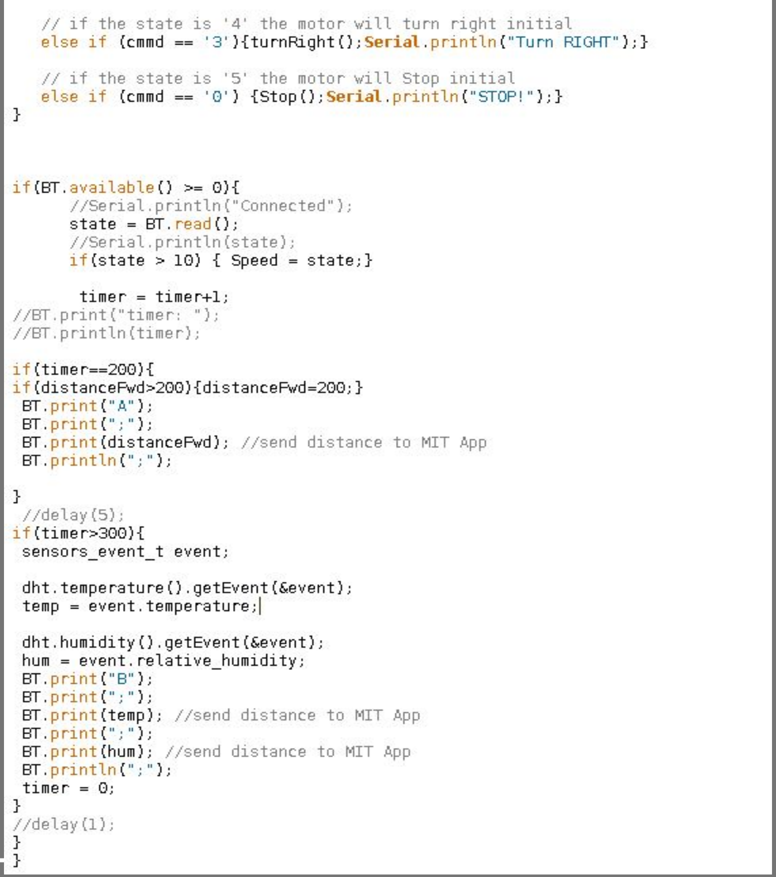


```
// Functions
void backward(){
    digitalWrite(36,HIGH);digitalWrite(34,LOW);digitalWrite(4,LOW);digitalWrite(5,HIGH);digitalWrite(6,HIGH);digitalWrite(7,LOW);digitalWrite(8,LOW);digitalWrite(9,HIGH);digitalWrite(10,LOW);digitalWrite(11,HIGH);digitalWrite(12,LOW);digitalWrite(14,HIGH); }

void forward(){
    digitalWrite(36,LOW);digitalWrite(34,HIGH); digitalWrite(4,HIGH);digitalWrite(5,LOW); digitalWrite(6,LOW);digitalWrite(7,HIGH);digitalWrite(8,HIGH);digitalWrite(9,LOW);digitalWrite(10,HIGH);digitalWrite(11,LOW);digitalWrite(12,HIGH);digitalWrite(14,LOW);}

void turnRight(){
    digitalWrite(36,LOW);digitalWrite(34,HIGH);digitalWrite(6,LOW);digitalWrite(7,HIGH);digitalWrite(10,HIGH); digitalWrite(11,LOW);digitalWrite(4,LOW);digitalWrite(5,HIGH);digitalWrite(8,LOW);digitalWrite(9,HIGH);digitalWrite(12,LOW);digitalWrite(14,HIGH);}

void turnLeft(){
    digitalWrite(36,HIGH);digitalWrite(34,LOW); digitalWrite(6,HIGH);digitalWrite(7,LOW); digitalWrite(10,LOW);digitalWrite(11,HIGH); digitalWrite(4,HIGH);digitalWrite(5,LOW);digitalWrite(8,HIGH);digitalWrite(9,LOW);digitalWrite(12,HIGH);digitalWrite(14,LOW);}

void Stop(){
    digitalWrite(36,LOW); digitalWrite(34,LOW); digitalWrite(4,LOW); digitalWrite(5,LOW); digitalWrite(6,LOW); digitalWrite(7,LOW);digitalWrite(8,LOW);digitalWrite(9,LOW);digitalWrite(10,LOW);digitalWrite(11,LOW); digitalWrite(12,LOW); digitalWrite(14,LOW);}

long data()
{  digitalWrite(trigpin,LOW); delayMicroseconds(2);  digitalWrite(trigpin,HIGH);  delayMicroseconds(10);  duration=pulseIn (echopin,HIGH);
  return duration / 29 / 2; }

void setup() {
// ultrasonic

    pinMode (trigpin, OUTPUT);
    pinMode (echopin, INPUT );

// Temp and humidity

dht.begin();
  // Print temperature sensor details.
  sensor_t sensor;
  dht.temperature().getSensor(&sensor);

  // Print humidity sensor details.
  dht.humidity().getSensor(&sensor);

  // Set delay between sensor readings based on sensor details.
  //delayMS = sensor.min_delay / 1000;


// Motor
  pinMode(MotorAinput1, OUTPUT);
  pinMode(MotorAinput2, OUTPUT);
  pinMode(MotorBinput1, OUTPUT);
  pinMode(MotorBinput2, OUTPUT);
  pinMode(MotorCinput1, OUTPUT);
  pinMode(MotorCinput2, OUTPUT);
  pinMode(MotorDinput1, OUTPUT);
```

Code3: Arduino

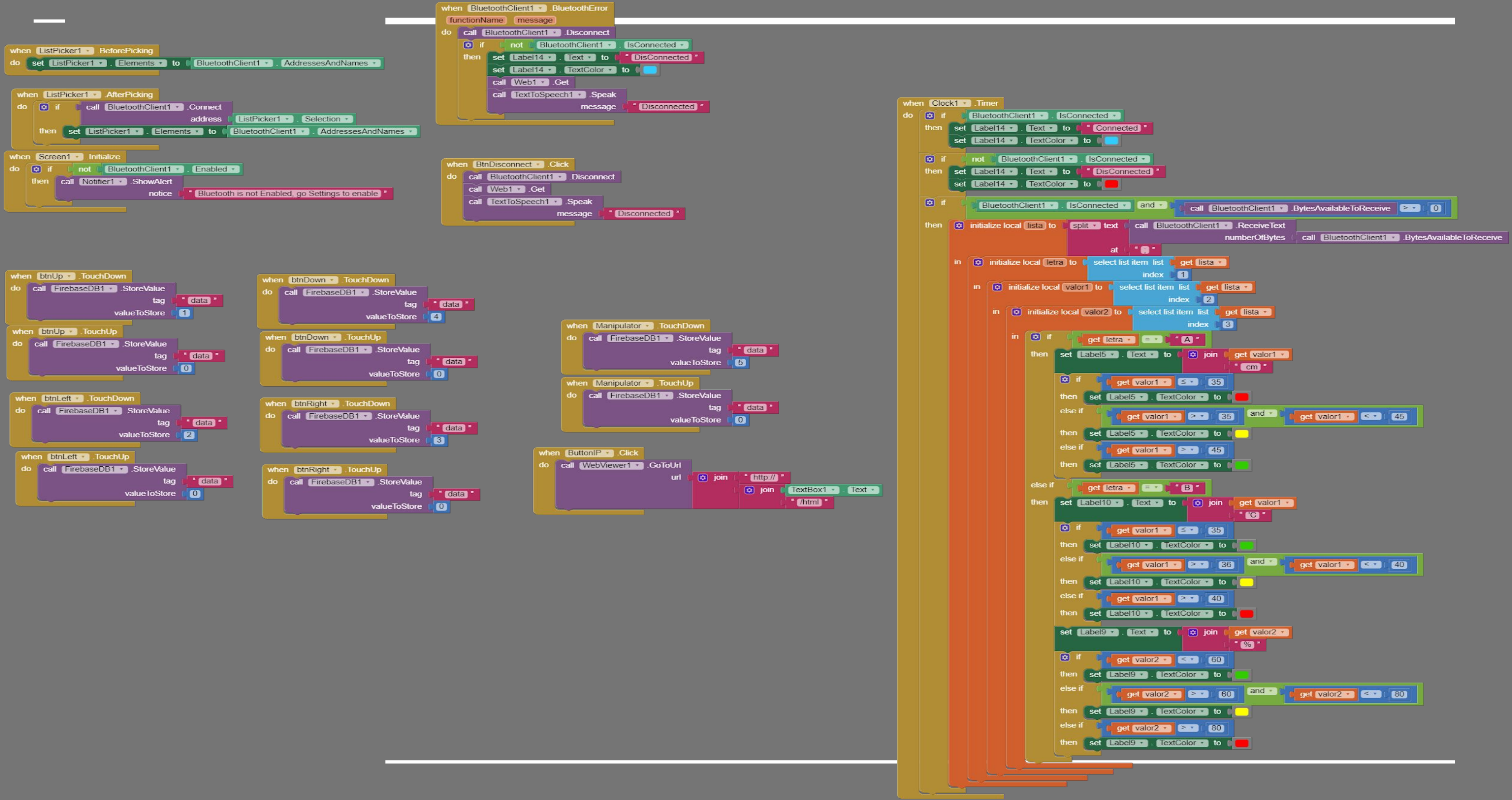


```
// Motor
  pinMode(MotorAinput1, OUTPUT);
  pinMode(MotorAinput2, OUTPUT);
  pinMode(MotorBinput1, OUTPUT);
  pinMode(MotorBinput2, OUTPUT);
  pinMode(MotorCinput1, OUTPUT);
  pinMode(MotorCinput2, OUTPUT);
  pinMode(MotorDinput1, OUTPUT);
  pinMode(MotorDinput2, OUTPUT);
  pinMode(MotorEinput1, OUTPUT);
  pinMode(MotorEinput2, OUTPUT);
  pinMode(MotorFinput1, OUTPUT);
  pinMode(MotorFinput2, OUTPUT);

  // initialize serial communication at 9600 bits per second:
  Serial.begin(9600);
  BT.begin(9600); // Setting the baud rate of Software Serial Library
}

char cmmd;

void loop() {
    //if some date is sent, reads it and saves in state
    if (Serial.available()) {
    cmmd = Serial.read();
    //Serial.print("You sent Arduino: ");
    Serial.println(cmmd);


    distanceFwd = data();
    Serial.println(distanceFwd);

   if((distanceFwd<set) && (chk==1)){chk = 2; Stop();}
   if(distanceFwd>set){chk = 0;}

   // if the state is '1' the DC motor will go forward initial
   if ((cmmd == '1') && (chk==0)){chk = 1; forward();Serial.println("Go Forward!");}

   // if the state is '2' the motor will Reverse initial
   else if (cmmd == '4'){backward();Serial.println("Reverse!");}

   // if the state is '3' the motor will turn left initial
   else if (cmmd == '2'){turnLeft();Serial.println("Turn LEFT");}

   // if the state is '4' the motor will turn right initial
   else if (cmmd == '3'){turnRight();Serial.println("Turn RIGHT");}

   // if the state is '5' the motor will Stop initial
```



```
   // if the state is '4' the motor will turn right initial
   else if (cmmd == '3'){turnRight();Serial.println("Turn RIGHT");}

   // if the state is '5' the motor will Stop initial
   else if (cmmd == '0') {Stop();Serial.println("STOP!");}
}



if(BT.available() >= 0){
      //Serial.println("Connected");
      state = BT.read();
      //Serial.println(state);
      if(state > 10) { Speed = state;}

       timer = timer+1;
//BT.print("timer: ");
//BT.println(timer);

if(timer==200){
if(distanceFwd>200){distanceFwd=200;}
 BT.print("A");
 BT.print(";");
 BT.print(distanceFwd); //send distance to MIT App
 BT.println(";");

}
 //delay(5);
if(timer>300){
 sensors_event_t event;

 dht.temperature().getEvent(&event);
 temp = event.temperature;

 dht.humidity().getEvent(&event);
 hum = event.relative_humidity;
 BT.print("B");
 BT.print(";");
 BT.print(temp); //send distance to MIT App
 BT.print(";");
 BT.print(hum); //send distance to MIT App
 BT.println(";");
 timer = 0;
}
//delay(1);
}
}
```

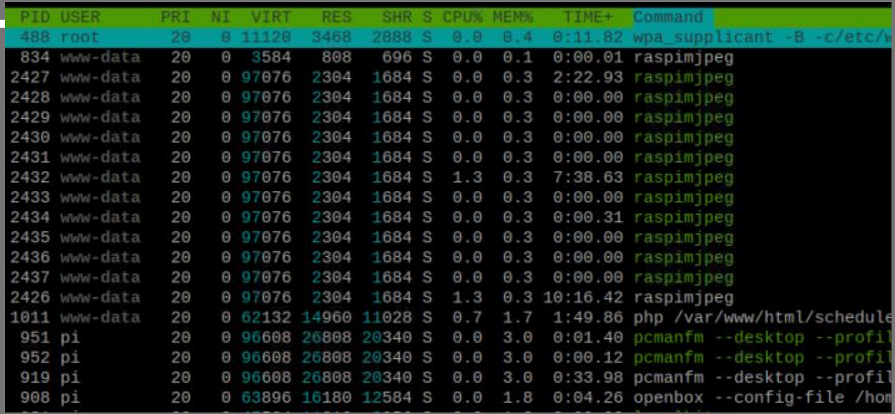
Code Blocks: App Inventor



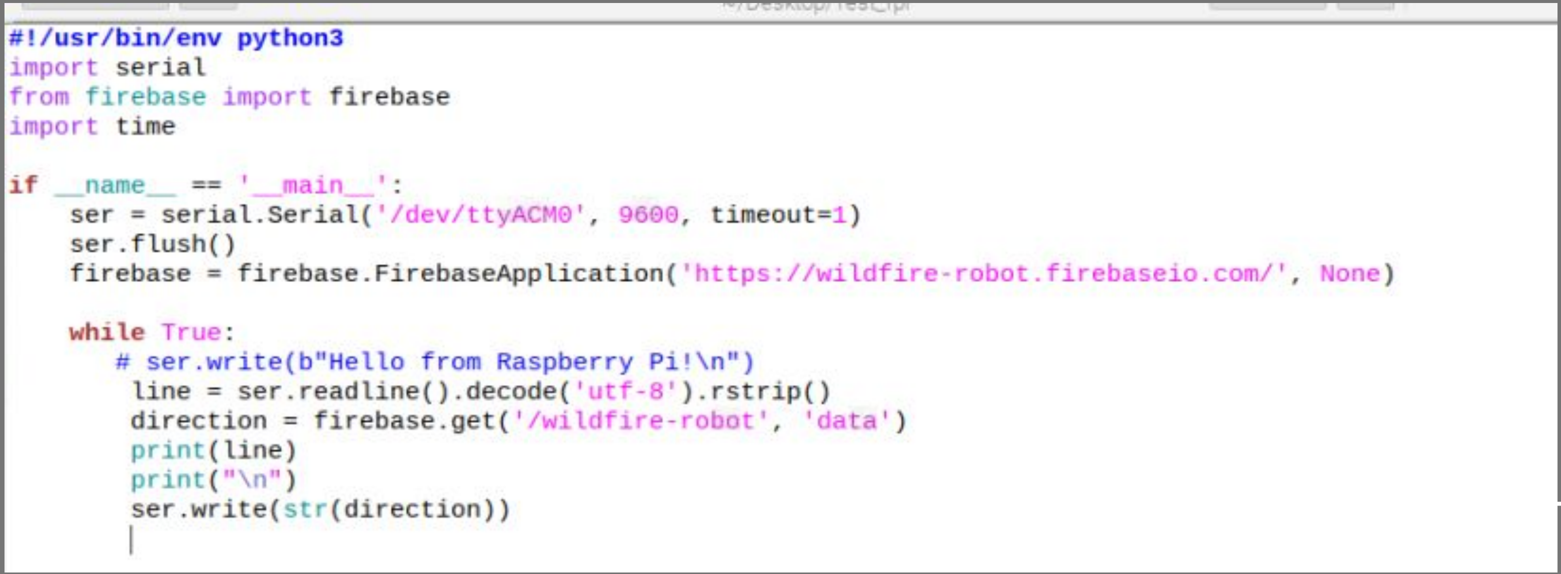
```
when ListPicker1 .BeforePicking
do  set ListPicker1 . Elements to BluetoothClient1 . AddressesAndNames

when ListPicker1 .AfterPicking
do  if   call BluetoothClient1 .Connect
                          address  ListPicker1 . Selection
    then set ListPicker1 . Elements to BluetoothClient1 . AddressesAndNames

when Screen1 .Initialize
do  if   not BluetoothClient1 . Enabled
    then call Notifier1 .ShowAlert
                     notice  " Bluetooth is not Enabled, go Settings to enable "

when BluetoothClient1 .BluetoothError
 functionName  message
do  call BluetoothClient1 .Disconnect
    if   not BluetoothClient1 . IsConnected
    then set Label14 . Text to " DisConnected "
         set Label14 . TextColor to 🟦
         call Web1 .Get
         call TextToSpeech1 .Speak
                         message  " Disconnected "

when BtnDisconnect .Click
do  call BluetoothClient1 .Disconnect
    call Web1 .Get
    call TextToSpeech1 .Speak
                    message  " Disconnected "

when btnUp .TouchDown
do  call FirebaseDB1 .StoreValue
                      tag  " data "
             valueToStore  1

when btnUp .TouchUp
do  call FirebaseDB1 .StoreValue
                      tag  " data "
             valueToStore  0

when btnLeft .TouchDown
do  call FirebaseDB1 .StoreValue
                      tag  " data "
             valueToStore  2

when btnLeft .TouchUp
do  call FirebaseDB1 .StoreValue
                      tag  " data "
             valueToStore  0

when btnDown .TouchDown
do  call FirebaseDB1 .StoreValue
                      tag  " data "
             valueToStore  4

when btnDown .TouchUp
do  call FirebaseDB1 .StoreValue
                      tag  " data "
             valueToStore  0

when btnRight .TouchDown
do  call FirebaseDB1 .StoreValue
                      tag  " data "
             valueToStore  3

when btnRight .TouchUp
do  call FirebaseDB1 .StoreValue
                      tag  " data "
             valueToStore  0

when Manipulator .TouchDown
do  call FirebaseDB1 .StoreValue
                      tag  " data "
             valueToStore  5

when Manipulator .TouchUp
do  call FirebaseDB1 .StoreValue
                      tag  " data "
             valueToStore  0

when ButtonIP .Click
do  call WebViewer1 .GoToUrl
                     url  join  " http:// "
                                join  TextBox1 . Text
                                      " /html "

when Clock1 .Timer
do  if   BluetoothClient1 . IsConnected
    then set Label14 . Text to " Connected "
         set Label14 . TextColor to 🟦
    if   not BluetoothClient1 . IsConnected
    then set Label14 . Text to " DisConnected "
         set Label14 . TextColor to 🟥
    if   BluetoothClient1 . IsConnected and call BluetoothClient1 .BytesAvailableToReceive > 0
    then initialize local lista to split text call BluetoothClient1 .ReceiveText
                                                     numberOfBytes  call BluetoothClient1 .BytesAvailableToReceive
                                         at " ; "
         in initialize local letra to select list item list get lista
                                                       index 1
            in initialize local valor1 to select list item list get lista
                                                           index 2
               in initialize local valor2 to select list item list get lista
                                                              index 3
                  in if   get letra = " A "
                     then set Label5 . Text to join get valor1
                                                    " cm "
                          if      get valor1 ≤ 35
                          then    set Label5 . TextColor to 🟥
                          else if get valor1 > 35 and get valor1 < 45
                          then    set Label5 . TextColor to 🟨
                          else if get valor1 > 45
                          then    set Label5 . TextColor to 🟩
                     else if get letra = " B "
                     then set Label10 . Text to join get valor1
                                                     " 'C "
                          if      get valor1 ≤ 35
                          then    set Label10 . TextColor to 🟩
                          else if get valor1 > 36 and get valor1 < 40
                          then    set Label10 . TextColor to 🟨
                          else if get valor1 > 40
                          then    set Label10 . TextColor to 🟥
                          set Label9 . Text to join get valor2
                                                    " % "
                          if      get valor2 < 60
                          then    set Label9 . TextColor to 🟩
                          else if get valor2 > 60 and get valor2 < 80
                          then    set Label9 . TextColor to 🟨
                          else if get valor2 > 80
                          then    set Label9 . TextColor to 🟥
```

Code: Raspberry Pi

```
$ sudo apt-get python-firefox
$ git clone
https://github.com/silvanmelchior/RPiCamWebInterface.git
$ cd RPi_Cam_Web_Interface
$ ./install.sh
```



```
  PID USER       PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
  488 root        20   0 11120  3468  2888 S  0.0  0.4  0:11.82 wpa_supplicant -B -c/etc/w
  834 www-data    20   0  3584   808   696 S  0.0  0.1  0:00.01 raspimjpeg
 2427 www-data    20   0 97076  2304  1684 S  0.0  0.3  2:22.93 raspimjpeg
 2428 www-data    20   0 97076  2304  1684 S  0.0  0.3  0:00.00 raspimjpeg
 2429 www-data    20   0 97076  2304  1684 S  0.0  0.3  0:00.00 raspimjpeg
 2430 www-data    20   0 97076  2304  1684 S  0.0  0.3  0:00.00 raspimjpeg
 2431 www-data    20   0 97076  2304  1684 S  0.0  0.3  0:00.00 raspimjpeg
 2432 www-data    20   0 97076  2304  1684 S  1.3  0.3  7:38.63 raspimjpeg
 2433 www-data    20   0 97076  2304  1684 S  0.0  0.3  0:00.00 raspimjpeg
 2434 www-data    20   0 97076  2304  1684 S  0.0  0.3  0:00.31 raspimjpeg
 2435 www-data    20   0 97076  2304  1684 S  0.0  0.3  0:00.00 raspimjpeg
 2436 www-data    20   0 97076  2304  1684 S  0.0  0.3  0:00.00 raspimjpeg
 2437 www-data    20   0 97076  2304  1684 S  0.0  0.3  0:00.00 raspimjpeg
 2426 www-data    20   0 97076  2304  1684 S  1.3  0.3 10:16.42 raspimjpeg
 1011 www-data    20   0 62132 14960 11028 S  0.7  1.7  1:49.86 php /var/www/html/schedule
  951 pi          20   0 96608 26808 20340 S  0.0  3.0  0:01.40 pcmanfm --desktop --profil
  952 pi          20   0 96608 26808 20340 S  0.0  3.0  0:00.12 pcmanfm --desktop --profil
  919 pi          20   0 96608 26808 20340 S  0.0  3.0  0:33.98 pcmanfm --desktop --profil
  908 pi          20   0 63896 16180 12584 S  0.0  1.8  0:04.26 openbox --config-file /hom
```



```python
#!/usr/bin/env python3
import serial
from firebase import firebase
import time

if __name__ == '__main__':
    ser = serial.Serial('/dev/ttyACM0', 9600, timeout=1)
    ser.flush()
    firebase = firebase.FirebaseApplication('https://wildfire-robot.firebaseio.com/', None)

    while True:
       # ser.write(b"Hello from Raspberry Pi!\n")
        line = ser.readline().decode('utf-8').rstrip()
        direction = firebase.get('/wildfire-robot', 'data')
        print(line)
        print("\n")
        ser.write(str(direction))
```

Video link :

https://drive.google.com/drive/u/1/folders/1cd9HPaMhE9OKRn-4DI9ZDl0HqJ6cy7ts

Thank You!
Suggestions + Questions?