Faculty of Engineering and Information Technology

Computer Engineering Department

The chair of hope system

Prepared by:

Rose Barakat and Raghad Marie.

Rosebarakat963@gmail.com

Raghadmarie@gmail.com

Supervised by:

Dr. Anas Toma.

Presented in partial fulfillment of the requirements for Bachelor degree in Computer Engineering.

Acknowledgment:

After working on our project, we'd like to thank our god (SWT), also thank our supervisor Dr. Anas Toma for his supervising and observing, we'd like to thank our family and friends for their support during our university years.

Disclaimer:

This report was written by students Raghad Marie and Rose Barakat at the Computer Engineering Department, Faculty of Engineering, An-Najah National University. It has not been altered or corrected, other than editorial corrections, as a result of assessment and it may contain language as well as content errors. The views expressed in it together with any outcomes and recommendations are solely those of the students. An-Najah National University accepts no responsibility or liability for the consequences of this report being used for a purpose other than the purpose for which it was commissioned.

The table of contents:

The table of Figures:

Abstract:

Our project is important because Palestine is considered one of the states which has high proportion of disabled people relative to its population. and we need to help them to live normal life as any one does.

One of the biggest obstacles facing a disabled person climbing stairs, there is a kind of chairs that can climb the stairs which were previously invented. Our project is a developed chair from this kind to make the chair integrated in terms of safety and multi-functionality.

Our idea is a system for the disabled people which have many features, first it guarantees climbing the stairs safely by sensing the tilt and balancing the chair based on the angle of the chair inclination, and ensures that the person is still fine and has not fallen from his chair, also make them control their house devices wirelessly.

So, the main objectives of our project providing an integrated support environment for people to live normal life.

We used a model, which is a children's toy available in the market to represent the idea of climbing the stairs (Through an elastic belt that has a strong cohesion with the ground and allows movement in the same time), in the lowest cost. As this:

THE TOY \ FIGURE

After that, we had made a mobile application on app inventor, which we used it as a controlling system, to control the movement and the speed, ensure that the person in fine situation, deal with emergency statues, and control the house devices using WIFI.

We used ESP32 module with Arduino IDE because it supports the features which we used.

And we used a sensor which can measure angular momentum or rotation along all the three axes in the chair to handle the seats' tilt to add safety while climbing the stair.

Chapter 1: Introduction

1.1 Statement of the problem

48.4% of the total number of people with special needs in the Palestinian Territories are those with movement disabilities, so the idea came when we see this group in our society suffering from many things, so can we provide a safe and easy environment for them from this category and specifically paraplegia? because the traditional wheelchairs don't provide that and there is an urgent need to find innovative solutions to help them.

1.2 Objectives

The project aims to help people with special needs from category of paraplegia in terms of safety and controlling the surrounding area like the stair and the electrical appliances, so we hope that the project will help them live a normal life and ease it to them as much as possible in dealings and using, and this will open up facilities for them to enable them to integrate into the labor market and other aspects.

1.3 Importance of work

Traditional chairs don't fulfill the purpose and it shows a heavy dependence on people, so our system will make their life easier and more secure, and it will reduce dependence on others significantly. It will enable them to handle their home and outdoor environment.

1.4 Organization of the report

In this report, first you will see an introduction to know the problem and why we need to use the project in our community, then we started to talk about our journey while working on it, starting with the constraints which we faced, the earlier courses that we relied on it at work, and the components and codes that we used. Then you will see the literature review and the methodology we followed, finally you will see the results the discussion followed by conclusion.

Chapter 2: Constraints, Standards/ Codes and Earlier course work

2.1 Constraint

According to our lack of experience in mechanical field, we got help in designing the seat of the chair, and another one is the constant need for Wi-Fi so that our project based on the use of Wi-Fi. Finally, the GPS, this technology consumes a lot of battery capacity, in addition to the presence of engines and this is the reason why we changed the battery and put a battery with a higher capacity instead of the old one and we need to charge it continuously.

2.2 Standards/Codes

After searched and comparing, we used ESP32 as a microcontroller because it supports the features which we used like WIFI feature and has a good number of cores.

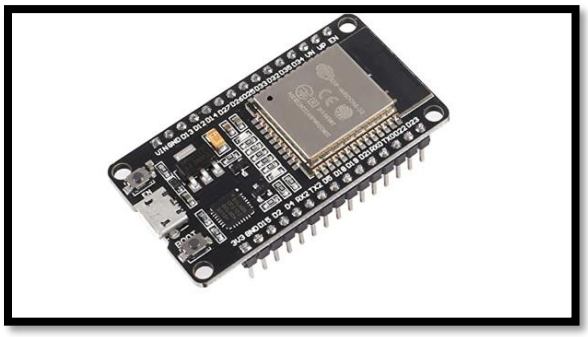
SPECS/BOARD	ESP32	ESP8266	ARDUINO UNO
Number of Cores	2	1	1
Architecture	32 Bit	32 Bit	8 Bit
CPU Frequency	160 MHz	80 MHz	16 MHz
WiFi	YES	YES	NO
BLUETOOTH	YES	NO	NO
RAM	512 KB	160 KB	2 KB
FLASH	16 MB	16 MB	32 KB
GPIO PINS	36	17	14
Busses	SPI, I2C, UART, I2S, CAN	SPI, I2C, UART, I2S	SPI, I2C, UART
ADC Pins	18	1	6
DAC Pins	2	0	0

THE DIFFRENCES BETWEEN ESP32 AND OTHER CONTRLLERS ٢ FIGURE

And we used Arduino IDE for coding.

2.2.1 The hardware components

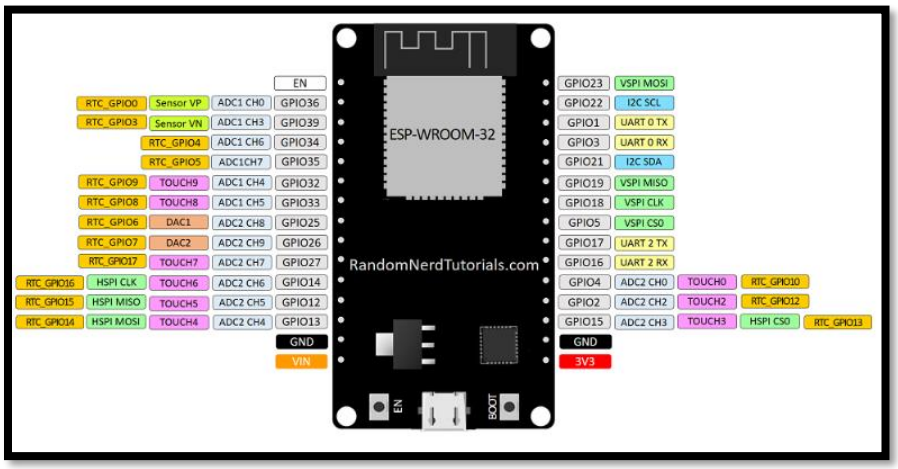
1- ESP32 WROOM with 30 pins



ESP32 WROOM۳۲ FIGURE

This board is used with 2.4 GHz dual-mode Wi-Fi and Bluetooth chips by TSMC 40nm low power technology, power and RF properties best, which is safe, reliable, and scalable to a variety of applications.

This is Pinout guide:



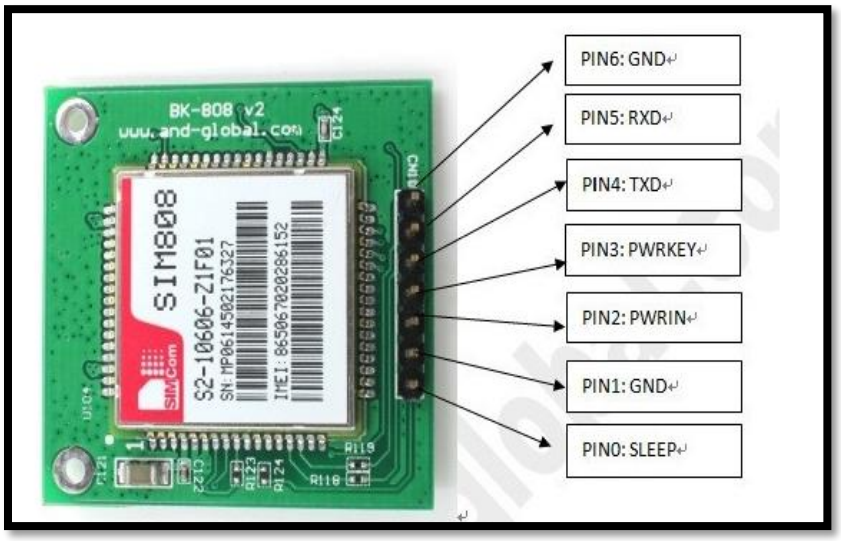
PINOUT OF ESP32٤ FIGURE

2-SIM808 GSM/GPRS/GPS Module:

We used it to receive GPS data, send and receive SMS messages Make and receive phone calls

SIM808 FIGURE

And this is pins definition:



PINS DEFINITION OF SIM808 FIGURE

3-L298N DC Motor Driver Module:

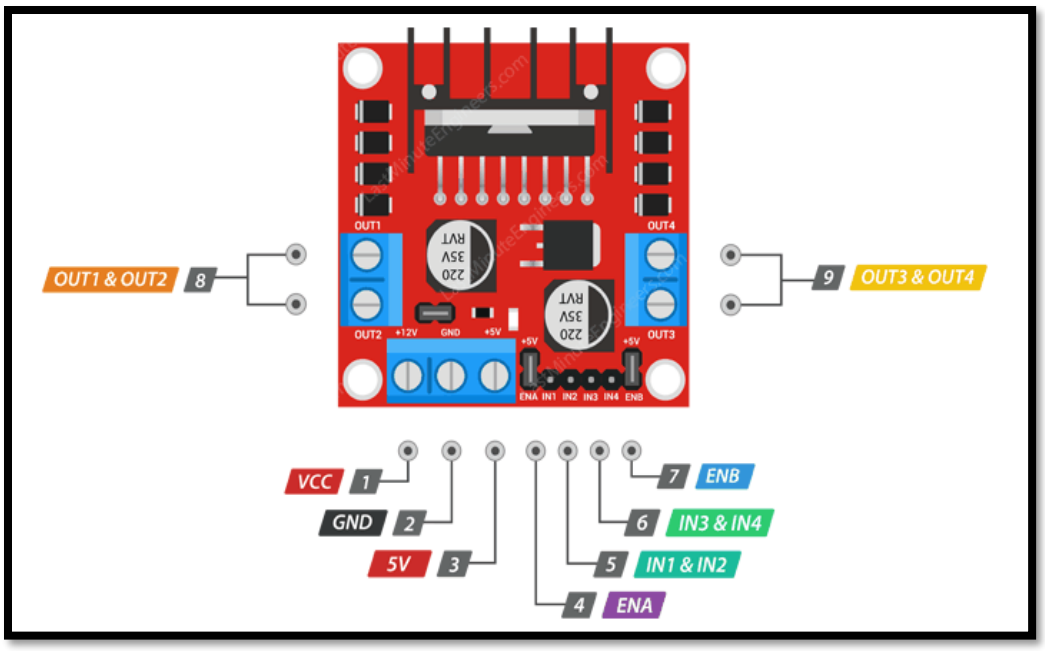


FIGURE 7 L298N DC MOTOR DRIVER

Pin 8 and pin 9 are outputs' channels for motor A and motor b.

Each channel of the module can deliver up to 2A to the DC motor. However, the amount of current supplied to the motor depends on the power supply of the system.

Pins 5 and 6 for the direction, the spinning direction of the motor can be controlled by applying logic HIGH (5V) or logic LOW (Ground) to these inputs.

Input 1	Input 2	Direction
0	0	OFF
0	1	Backward
1	0	Forward
1	1	OFF

Pins 4 and 7 for controlling the speed: Pulling these pins HIGH will cause the motors to spin, while pulling it LOW will stop them. But we controlled the speed by Pulse Width Modulation (PWM).

The module usually comes with a jumper on these pins. We removed the jumper to control the speed of the motors programmatically, and then connected them to the PWM-enabled pins on the esp32 module.

4- 5V CHANNEL RELAY MODULE ACTIVE HIGH/LOW

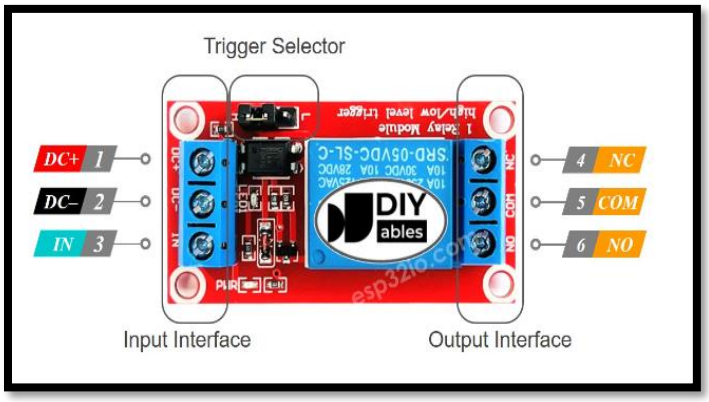


FIGURE 8 5V CHANNEL RELAY MODULE

The relay is a programmable switch that can be used to control ON/OFF electrical devices.

We used it to control the lamp and the fan because it required when we want to connect high voltage devices to the microcontroller.

5-MPU6050

inexpensive 6-axis Motion Tracking chip that combines a 3-axis gyroscope, 3-axis accelerometer, and a Digital Motion Processor (DMP) all in a small 4mm x 4mm package.

It can measure angular momentum or rotation along all the three axes, the static acceleration due to gravity, as well as dynamic acceleration resulting from motion, shock, or vibration.

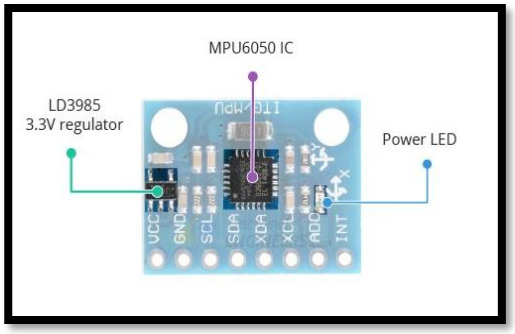


FIGURE 9 MPU6050

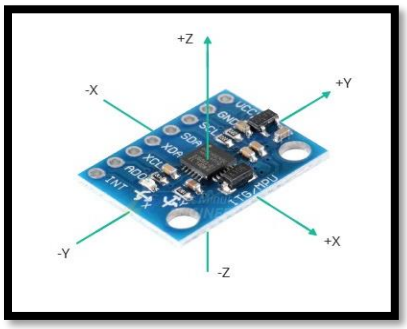
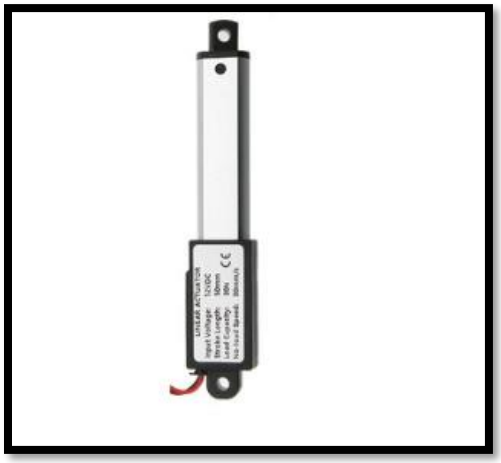


FIGURE 10 MPU6050 AXIS

Measuring Acceleration

The MPU6050 can measure acceleration using its on-chip accelerometer with four programmable full-scale ranges of ±2g, ±4g, ±8g and ±16g.

6-12V DC Linear Actuator:



12V DC LINEAR ACTUATOR ١١ FIGURE

It is a 12 V actuator 50 mm stroke 8mm/S with 60N Force, we used this linear actuator for controlling the seat according to this design:

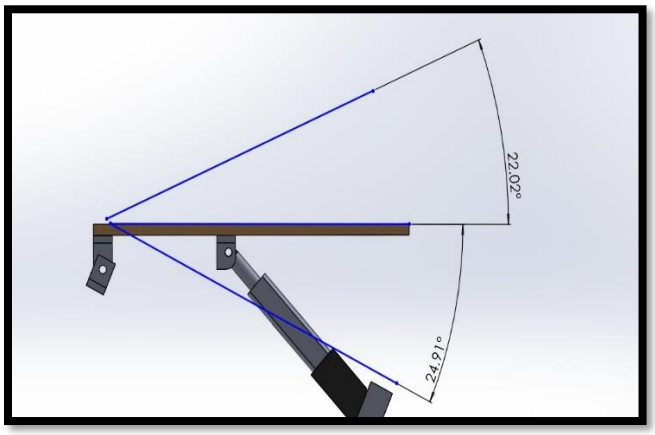
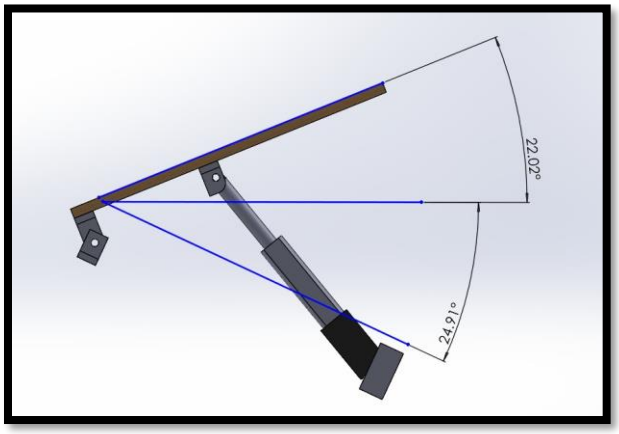
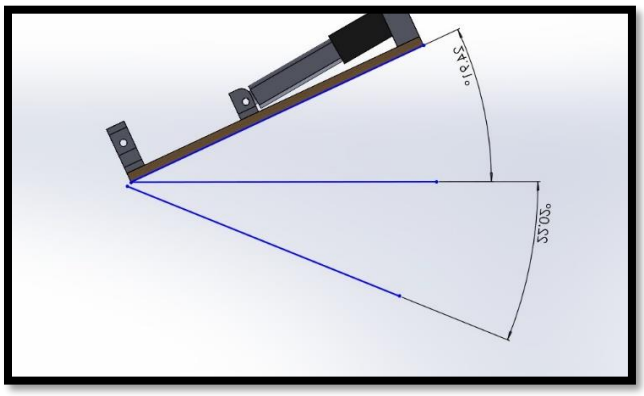


FIGURE 12 DC LINEAR ACTUATOR IN NORMAL STATE



DC LINEAR ACTUATOR WHEN GOING DOWN THE STAIRS ١٣ FIGURE



DC LINEAR ACTUATOR WHEN GOING UP THE STAIRS ١٤ FIGURE

2.3 Earlier course work

Working on our project depends on some courses we learned within the Computer Engineering program such as:

1-Micocontroller:

The microcontroller provides basic information about understanding the PIC Microcontroller and how to program hardware components.

2- Electronic circuits

This course has mainly contributed to helping us deal with electrical circuits and related connections.

3-Wirless:

Wireless is a term used to define telecommunication and data transmission without wires. This course helped us to control things using WIFI mechanism.

4-Arduino Course:

Arduino consists of both a physical programmable circuit board (often referred to as a microcontroller) and a piece of software, or IDE (Integrated Development Environment) that runs on your computer, used to write and upload computer code to the physical board.

App inventor course:

MIT App Inventor is a web application integrated development environment originally provided by Google, we learnt from it how to build a mobile application.

Chapter 3: Literature review

The idea came when we see this group in our society suffering from many things, and specially using the stairs in the home or outside, so we need to provide them integrated environment of ease of handling the surrounding area and the safety.

There are new types of wheelchairs which it can dealing with stairs through an elastic belt that has a strong cohesion with the ground and allows movement in the same time, they have appeared in recent years as mechanical graduation projects or companies that were able to manufacture them. For example, Dragon chair which was named "Stair climber" it helped people using the stair, but there are still signs of dependence on others (it required someone else to help them), and it is not safe relatively.

FIGURE 15 STAIR CLIMBER MODEL

STAIR CLIMBER ١٦ FIGURE

Then it appeared improved chair with the same idea (using the belt) but it controlled by himself without need someone else, and there is a controlled joy steak to balance the seat during using the stair, and this is much better than the previous.

FIGURE 17 STAIR CLIMBER

FIGURE 18 STAIR CLIMBER

From this point we started thinking how to improve a system for them to live normal life far from being dependent on others, and it includes as many devices as they use.

Chapter 4: Methodology

4.1 Methodology of the work:

We aim in our project to help paraplegics so that they can use all the facilities around them easily, and without relying on others, so after we noticed the people who are round us are suffering from problems in their life, we came with the idea of a system includes chair and the surrounding devices that is safe and easy to use by a person who suffers from paraplegia, so after we decided the idea, we searched for similar projects and information in the same field. Then we searched about the components that we used in our project individually to understand how they works and how to use and link them effectively, we searched and chose the best options to suit the needs of our project.

So, in our project, we used a system includes chair that the person who suffers from paraplegia use it most of his time, the stair which we find it in home and all the initiations, the house devices like lamp and fan, and the mobile which we used it as a controller in the project.

We used app inventor to design a mobile application works using WI-FI technology, we chose app inventor because it is easy to learn and it support Wi-Fi technology which we need.

We used WI-FI in controlling the system because we can add devices to control effectively and easily and it serves longer distances than other technologies such as Bluetooth.

4.2 Methodology of the system:

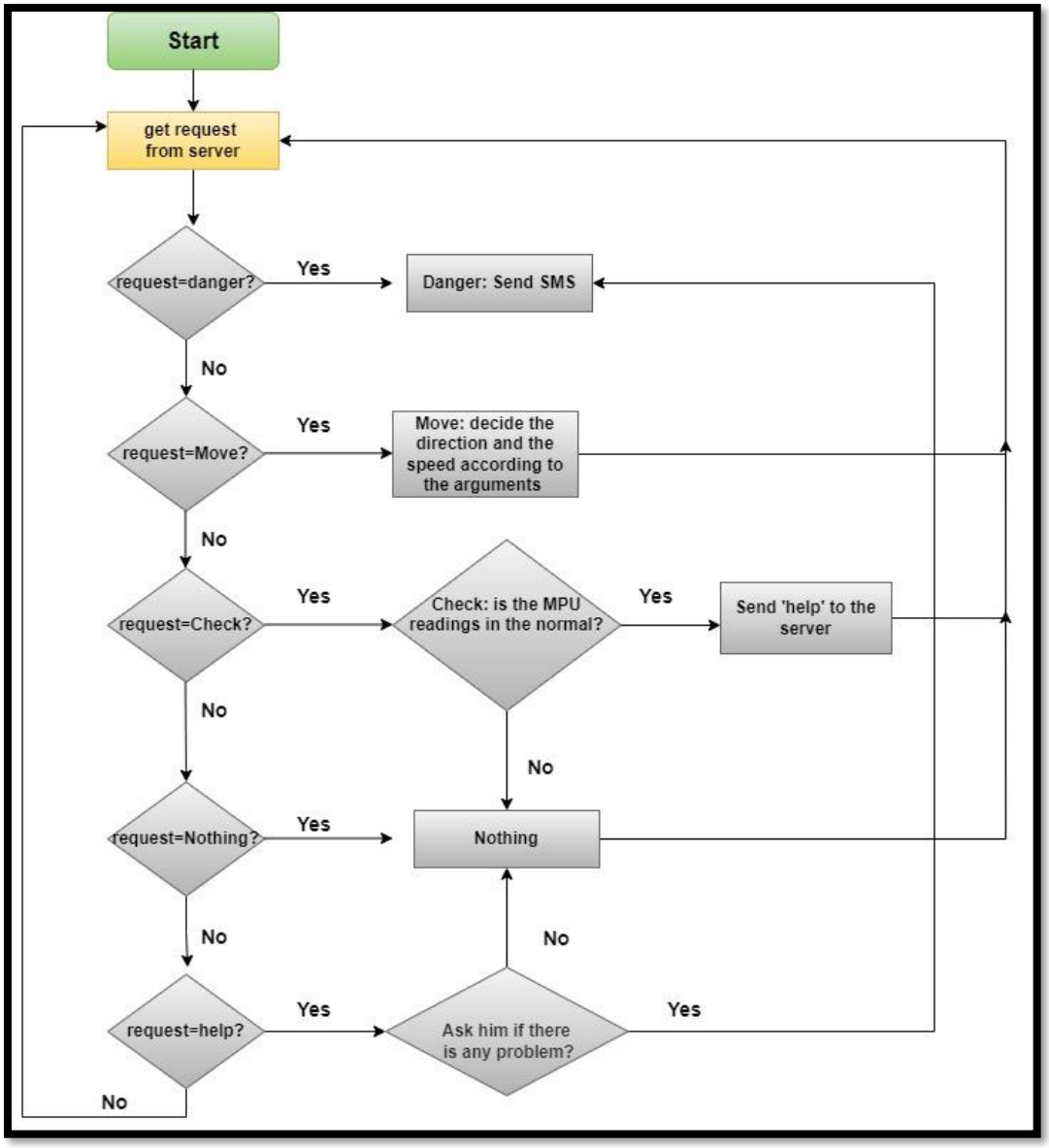


FIGURE 19 DIAGRAM SHOWING HOW THE SYSTEM WORKS

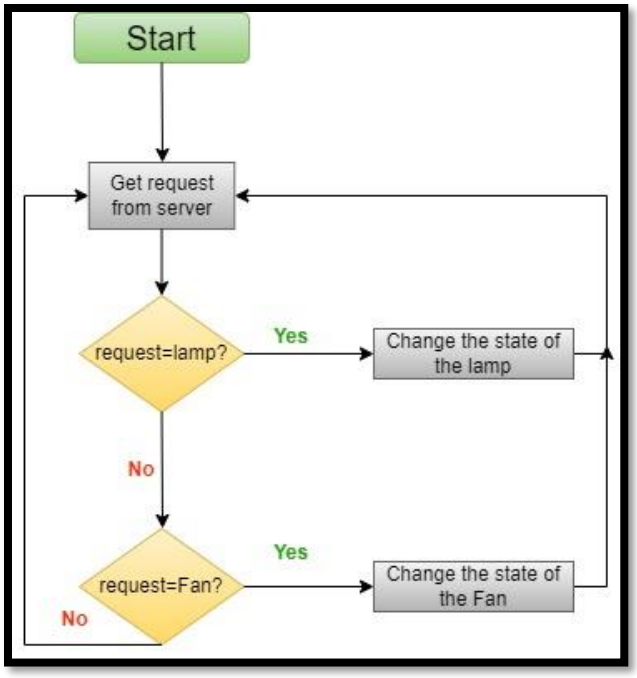


FIGURE 20 DIAGRAM SHOWING HOW THE HOUSE WORKS

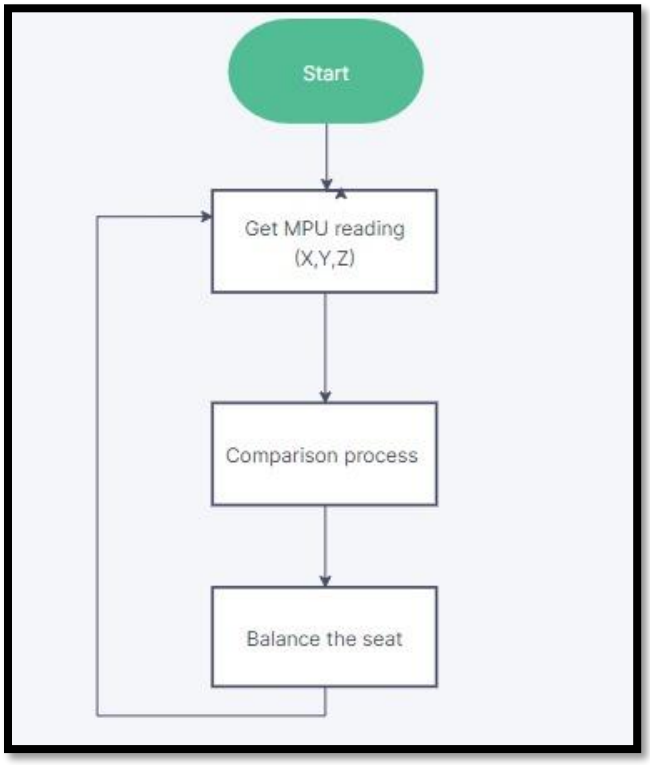


FIGURE 21 DIAGRAM SHOWING HOW THE MPU6050 WORKS

4.3 Connecting components:

The chair: contains esp32 microcontroller which is connected with all the following, MPU6050 sensor to keep reading x y z axes, it returns I2C data, **2** L298N DC Motor Driver Module to control the two motors which is used to move the chair, and the linear motor which is used to balance the seat, GSM module which is connected with the esp32 serially using serial communications.

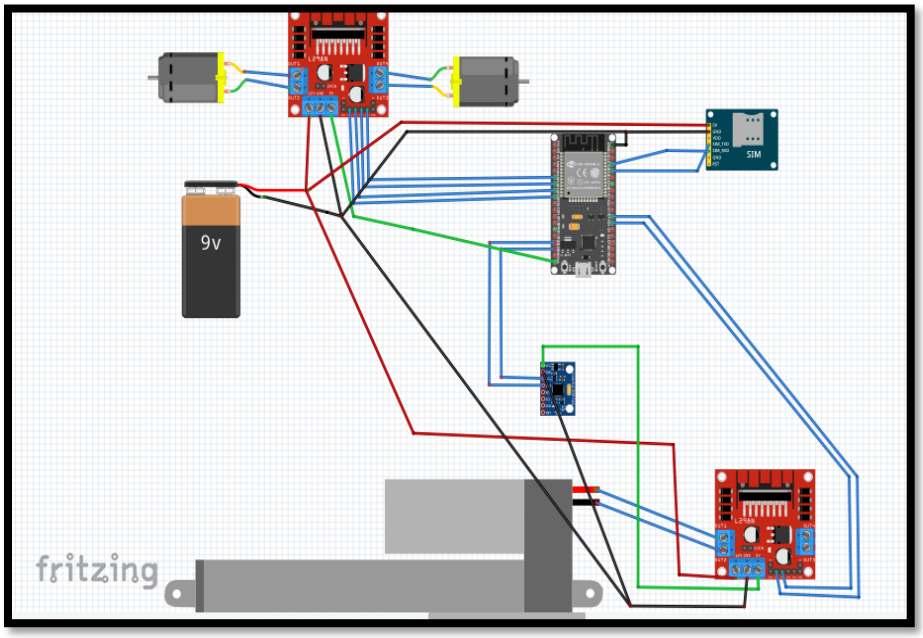


FIGURE 22 CONNECTING COMPONENTS DIAGRAM

The home: contains esp32 with 2 relays and led strip and fan.

1- 12VDC LED STRIP

FIGURE 23 12VDC LED STRIP

2- **12V DC COOLING FAN**

FIGURE 24 12V DC COOLING FAN

Chapter 5: Results and Discussion

In the end, after our journey in this project, the project worked as expected and planned. It works correctly and realizes the ideas; we tested all the features that we put on the system, and we were able to design an ideal and more advanced chair than the previous chairs and reduce dependence on others significantly.

The system does the following:

First: The application which we created through App inventor:

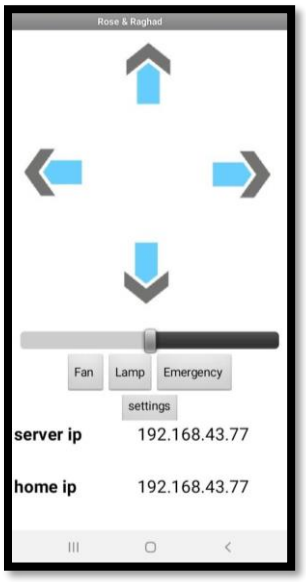


FIGURE 25 THE APPLICATION

It contains:

1- 4 Buttons for the movement of the chair in all directions.
2- Slider to control the speed for the chair.
3- Button for the emergency state to send SMS message to one of his family.
4- Buttons to control a Fan and Lamp.
5- Setting Button to choose the Ip addresses.

Second: The chair is able to climb up and down the stairs using the model that we brought (it has a belt with strong proportion of cohesion with the stairs).

The chairs' seat can balance while climbing the stair, this point provide safety while using it, it Keep the person in a state parallel to the straight floor. And most importantly, it is automatic using the sensor MPU6050.

Third: If there are abnormal readings From the MPU6050, Notification appears in the application with sound asking the person if there is a problem.

There are 3 possibilities for his/her response (yes there is a problem, No I'm in fine situation, or No response).

If he/she pressed yes or No response, A SMS message will be sent to one of his family with his/her GPS via google maps.

If he/she pressed "No, I'm in fine situation", it's fine, nothing will happen.

FIGURE 26 THE SYSTEM

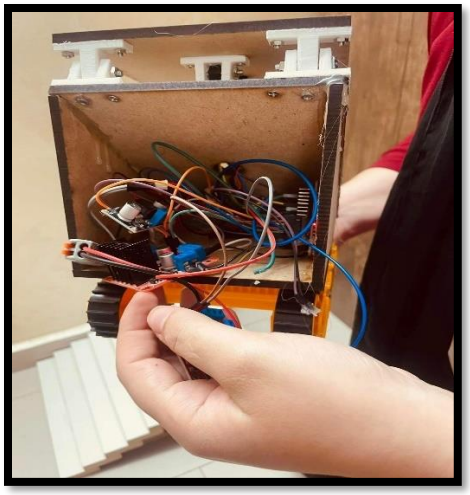


FIGURE 27 THE CHAIR MODEL1

FIGURE 28 THE CHAIR MODEL2

FIGURE 29 THE CHAIR MODEL3

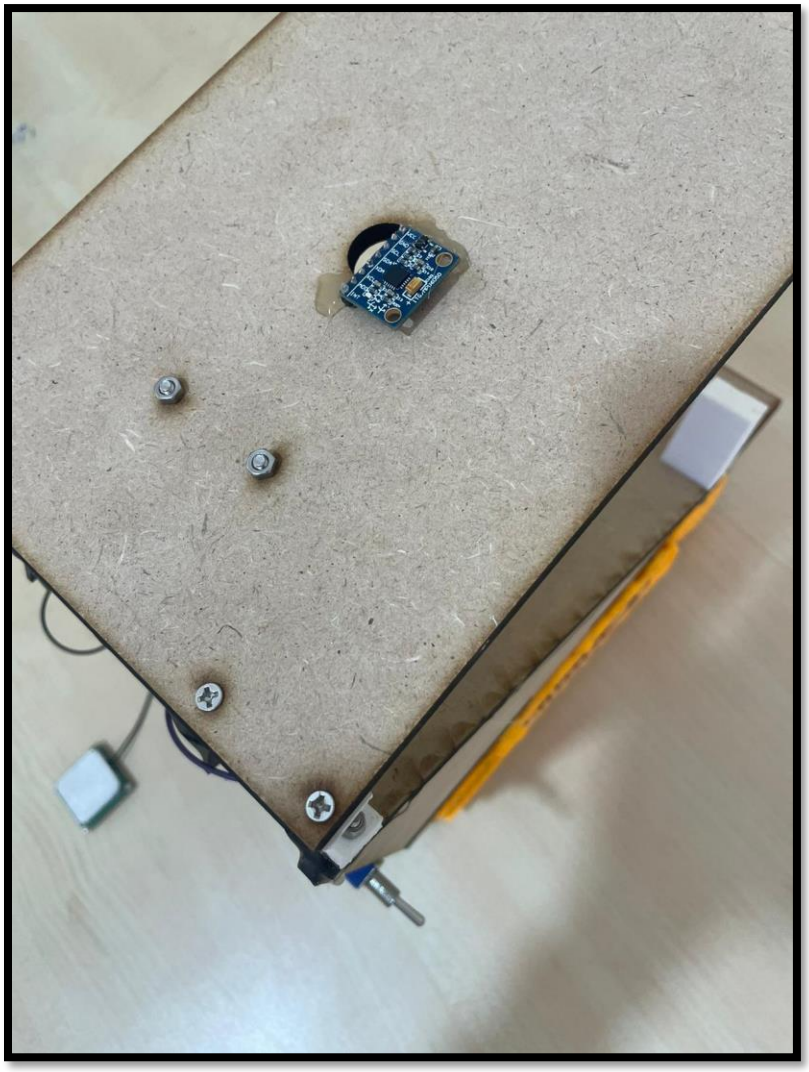


FIGURE 30 THE MPU6050

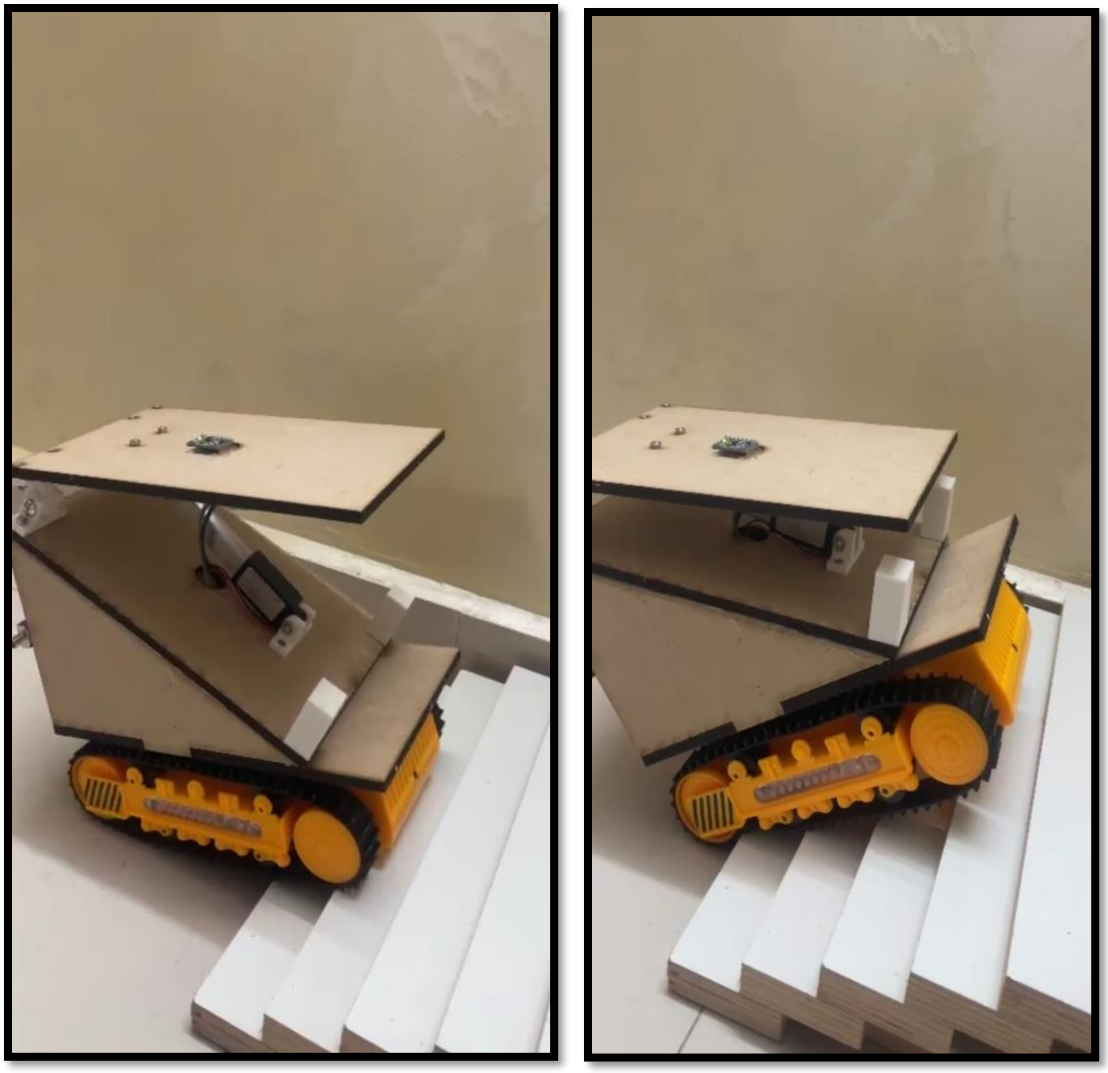


FIGURE 31 THE CHAIR WHILE CLIMBING THE STAIRS

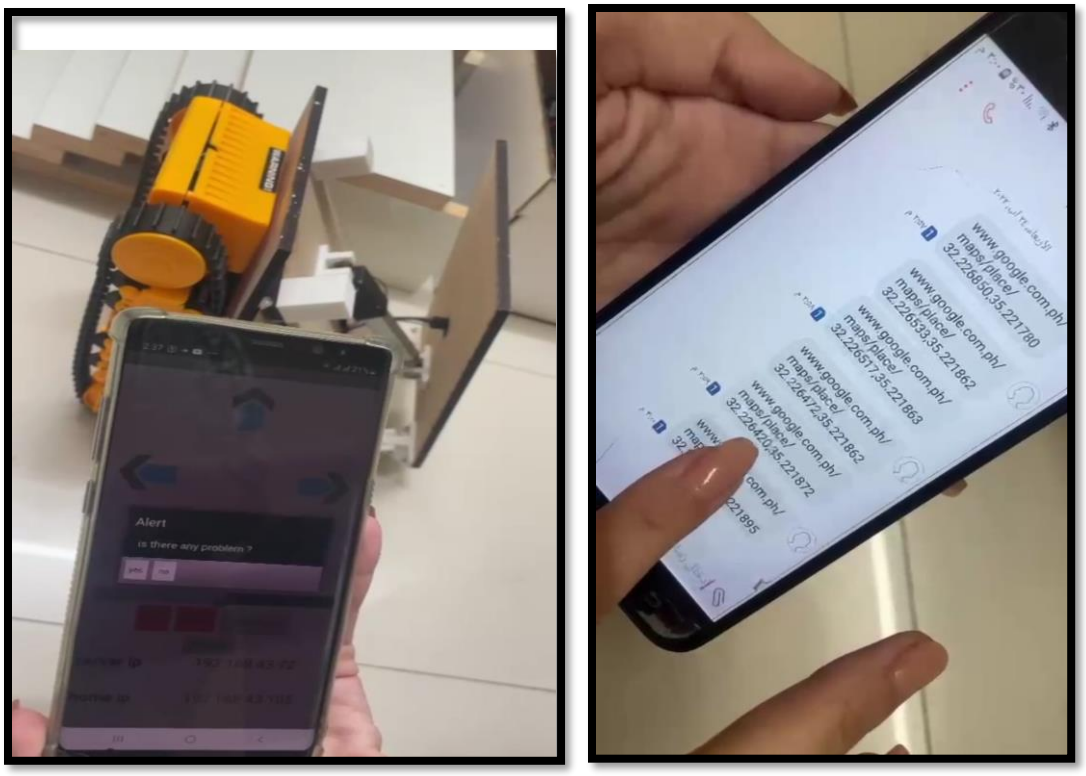


FIGURE 32 THE NOTIFICATION WHEN THE CHAIR FELL

Chapter 6: Conclusion

The Chair of hope system is an integrated system for people who suffering from paraplegia and enables them to live a normal life as a normal person without relying on others, it is a system that combines many devices that save time and effort for the person, such as controlling the house and climbing the stairs, and it has safety features such as contacting the person with his family and sending location if the chair fell.

we learned new things during our work, such as new hardware components that we haven't treat with them before.

There is a future work for our project, we were limited by the semester time. But we planned to improve the idea, to include treating with more devices such as the bathroom without anyone's help and more.

Chapter 7: References

1- Tutorials, R. N. (2021). ESP32 Pinout Reference: Which GPIO pins should you use? línea]. Available: https://randomnerdtutorials. com/esp32-pinout-reference-gpios.

2- Bharavi, U., & Sukesh, R. M. (2017, May). Design and development of GSM and GPS tracking module. In 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT) (pp. 283-288). IEEE.

3- Behera, P. K., & Gupta, A. (2018). Novel design of stair climbing wheelchair. Journal of Mechanical Science and Technology, 32(10), 4903-4908.

4- https://randomnerdtutorials.com/esp32-mpu-6050-accelerometer-gyroscope-arduino

5- https://www.instructables.com/Vehicle-Tracking-With-ESP32-GPS-SIM808-No-Monthly-/

6- الجهاز المركزي للإحصاء الفلسطيني
https://www.pcbs.gov.ps/postar.aspx?lang=ar&ItemID=4122

7- https://esp32io.com/tutorials/esp32-relay

Code for the home:

```
40 void setup(void) {
41   pinMode(led, OUTPUT);
42   pinMode(fan_pin, OUTPUT);
43   pinMode(lamp_pin, OUTPUT);
44   digitalWrite(led, 0);
45   Serial.begin(115200);
46   WiFi.mode(WIFI_STA);
47   WiFi.begin(ssid, password);
48   Serial.println("");
49
```

This is to configure the pins for the fan and the lamp, set the mode for the Wi-Fi "station mode".

```
51   while (WiFi.status() != WL_CONNECTED) {
52     delay(500);
53     Serial.print(".");
54   }
55   Serial.println("");
56   Serial.print("Connected to ");
57   Serial.println(ssid);
58   Serial.print("IP address: ");
59   Serial.println(WiFi.localIP());
60
61   if (MDNS.begin("esp32")) {
62     Serial.println("MDNS responder started");
63   }
64
```

Waiting for connection.

```
64
65   server.on("/", handleRoot);
66   server.on("/lamp", lamp);
67   server.on("/fan", fan);
68
69   server.on("/inline", []() {
70     server.send(200, "text/plain", "this works as well");
71   });
72
```

When getting a request from the server we checked if the request was lamp or fan, each one of them has a function.

```
void loop(void) {
  server.handleClient();
  delay(2);//allow the cpu to switch to other tasks
}

void lamp(void) {
  if (lampFlag == 1) {
    server.send ( 200 , "text/plain", "lampon");
    Serial.println("lampon");
    digitalWrite(lamp_pin, lampFlag);
    lampFlag = 0;

    return;
  }
  if (lampFlag == 0) {
    server.send ( 200 , "text/plain", "lampoff");
    Serial.println("lampoff");
    digitalWrite(lamp_pin, lampFlag);
    lampFlag = 1;
    return;
  }
}
```

We have a loop and we called handleClient inside it, it detects new requests, it will automatically execute the right functions that we specified in the setup.

Function lamp (): This function will work when lamp button has pressed, if we want to turn on the light, the server will send http state Ok and text " lampOn " to the client, the same thing when we want to turn it off.

```
void fan(void) {

  if (fanFlag == 1) {
    server.send ( 200 , "text/plain", "fanon");
    Serial.println("fanon");
    digitalWrite(fan_pin, fanFlag);
    fanFlag = 0;
    return;
  }
  if (fanFlag == 0) {
    server.send ( 200 , "text/plain", "fanoff");
    Serial.println("fanoff");
    digitalWrite(fan_pin, fanFlag);
    fanFlag = 1;
    return;
  }
}
```

Function fan (): This function will work when fan button is pressed, the same idea of the lamp.

Code For the Chair:

```
#include <Adafruit_MPU6050.h>
#include <Adafruit_Sensor.h>
#include <Wire.h>

Adafruit_MPU6050 mpu;
#include <HardwareSerial.h>
HardwareSerial GSMSerial(2);

#include <WiFi.h>
#include <WiFiClient.h>
#include <WebServer.h>
#include <ESPmDNS.h>

//SSID and Password to your ESP Access Point
const char* ssid = "Mohammed";
const char* password = "12345678";
String targetPhone="0594082583";

String command;             //String to store app command state.
int speedCar = 150; // 0 to 255
int speed_low = 60;
WebServer server(80);

/* Defining motor pins */
const int RMotor1 = 14;
const int RMotor2 = 15;
const int LMotor1 = 13;
const int LMotor2 = 12;
```

```
const int b1 = 26;
const int b2 = 27;

int checkIndex;
String date_time = "";
String LONG = "";
String LAT = "";
String dataIn = "";

long timer = 0;

bool check_flag = 0;
bool danger_flag = 0;
```

```
void setup() {

  pinMode(b1, OUTPUT);
  pinMode(b2, OUTPUT);

  Serial.begin(115200);
  GSMSerial.begin(9600, SERIAL_8N1, 16, 17); // the GPRS/GSM baud rate
  GSMSerial.setRxBufferSize(1024);
  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, password);
  Serial.println("");

  // Wait for connection
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.print("Connected to ");
  Serial.println(ssid);
  Serial.print("IP address: ");
  Serial.println(WiFi.localIP());

  if (MDNS.begin("esp32")) {
    Serial.println("MDNS responder started");
  }
```

This is the code which is inside setup, contains initializing serial communications, starting GSM shield, and setting the Wi-Fi mode as station, then starting Wi-Fi connection.

```
  server.on ( "/check", check );
  server.on ( "/danger", danger );
  server.on ( "/nothing", nothing );
  server.on ( "/move", Move );
  server.onNotFound ( HTTP_handleRoot );
  server.begin();
  initMotors();
  //init_mpu6050();
  ini_SIM808();
  if (GSMSerial.available() > 0) {

    dataIn = GSMSerial.readString();
    Serial.println("----------------------------dataIn--------------------------------");
    Serial.println(dataIn);
    Serial.println("-----------------------------end-------");
  }
}


```

When getting a request from the server, it will execute a special function depending on what followed by / ,then we initiate the motors, and sim808.

If there is a data was sent to the serial from GSM it will stored in String dataIn.

```

void loop() {
  server.handleClient();

  if (Serial.available() > 0) {
    Serial.read();

    SendLocation();
  }
}

```

handleClient Function: it detects new requests; it will automatically execute the right functions that we specified in the setup.

And send the location for the person if there is a data sent to the serial.

```
113
114 void Move(void) { //IP/Move?state=F
115
116   command = server.arg("State");
117   Serial.println(command);
118   if (command == "F") goForword();
119   else if (command == "B") goBack();
120   else if (command == "L") goLeft();
121   else if (command == "R") goRight();
122   else if (command == "0") speedCar = 100;
123   else if (command == "1") speedCar = 120;
124   else if (command == "2") speedCar = 140;
125   else if (command == "3") speedCar = 160;
126   else if (command == "4") speedCar = 180;
127   else if (command == "5") speedCar = 200;
128   else if (command == "6") speedCar = 215;
129   else if (command == "7") speedCar = 230;
130   else if (command == "8") speedCar = 240;
131   else if (command == "9") speedCar = 255;
132   else if (command == "S") stopRobot();
133
134   server.send ( 200 );
135   delay(1);
136 }
```

Function Move: it will execute when getting a request from the server as "//Ip/Move?state='charchter'".

This charchter could be B to go back, L to go left, R to go right, F to go Forward, and the numbers from 0 to 9 to control the speed.

Then it will send HTTP response OK (200) to the server.

```
137 void check(void) {
138
139
140   if (check_flag == 1) {
141     server.send ( 200, "text/plain", "help" );
142     Serial.println("check");
143     check_flag = 0;
144     return;
145   }
146
147   server.send ( 200);
148   return;
149
150 }
```

Function Check: if the chair tilts in danger position it will send help to the server.

```
152
153 void danger(void) {
154   check_flag = 0;
155   server.send ( 200 );
156   Serial.println("danger");
157   SendLocation();
158   return;
159 }
160
```

Function Danger: this Function called when the user pressed emergency button, it will send the location as SMS.

```
162
163
164 void nothing(void) {
165   check_flag = 0;
166   server.send ( 200 );
167   Serial.println("nothing");
168   return;
169 }
170
171
```

When the user pressed 'No, I'm in fine situation' there is nothing will happen.

```
175 void goForword() {
176
177   ledcWrite(3, 0);
178   ledcWrite(4, speedCar);
179   ledcWrite(5, speedCar);
180   ledcWrite(6, 0);
181 }
182 void goBack() {
183   ledcWrite(3, speedCar);
184   ledcWrite(4, 0);
185   ledcWrite(5, 0);
186   ledcWrite(6, speedCar);
187
188 }
189 void goRight() {
190
191   ledcWrite(3, 0);
192   ledcWrite(4, speedCar);
193   ledcWrite(5, 0);
194   ledcWrite(6, speedCar);
195 }
196 void goLeft() {
197
198   ledcWrite(3, speedCar);
199   ledcWrite(4, 0);
200   ledcWrite(5, speedCar);
201   ledcWrite(6, 0);
202 }
```

Movements' directions: This ledcWrite function is used to generate the signal with a specified duty cycle value. The first argument to this function is a channel number and the second value is the required duty cycle. The resolution= 8 bits so the speed of the chair between 0,255.

```
206 void stopRobot() {
207
208   ledcWrite(3, 0);
209   ledcWrite(4, 0);
210   ledcWrite(5, 0);
211   ledcWrite(6, 0);
212 }
213
214
215 void initMotors()
216 {
217   /* Configuring motor PWM functionalities to generate the signal */
218   /* ledcSetup(Channel, Frequency, Resolution) */
219   ledcSetup(3, 2000, 8); /* 2000 hz PWM, 8-bit resolution and range from 0 to 255 */
220   ledcSetup(4, 2000, 8); /* 2000 hz PWM, 8-bit resolution and range from 0 to 255 */
221   ledcSetup(5, 2000, 8); /* 2000 hz PWM, 8-bit resolution and range from 0 to 255 */
222   ledcSetup(6, 2000, 8); /* 2000 hz PWM, 8-bit resolution and range from 0 to 255 */
223
224   /* Attaching the channel to the GPIO to be controlled */
225   /* ledcAttachPin(GPIO, Channel) */
226   ledcAttachPin(RMotor1, 3);
227   ledcAttachPin(RMotor2, 4);
228   ledcAttachPin(LMotor1, 5);
229   ledcAttachPin(LMotor2, 6);
230
231 }
```

ledcSetup: This function is used to set the channel number, frequency, and resolution of the output signal. We should pass three arguments as an input to this function, channel number, frequency, and the resolution.

ledcAttachPin: This attaches any GPIO pins with a channel. This function accepts two arguments. One is the GPIO pin on which we want to get the OUTPUT of the signal and the second argument is the channel which we produce the signal.

```
233 void ini_SIM808() {
234   GSMSerial.print("AT+CGNSPWR=1\r");
235   delay(100);
236   GSMSerial.print("AT+CGNSSEQ=RMC\r");
237   delay(100);
238   GSMSerial.print("AT+CLIP=1\r");
239   delay(100);
240   GSMSerial.print("AT+CMGF=1\r");
241   delay(100);
242   GSMSerial.print("AT+CNMI=1,2,0,0,0");
243   GSMSerial.flush();
244 }
```

This function to initiate the sim808 to work, we used AT commands.

```
247
248 void split(String dataIn) {
249   int indexOfComma[10];
250   dataIn = dataIn.substring(dataIn.indexOf("+CGNSINF: 1,1,"));
251   for (int i = 0; i < 6; i++) {
252     indexOfComma[i] = dataIn.indexOf(",");
253     dataIn[indexOfComma[i]] = '#';
254   }
255
256
257
258   date_time = dataIn.substring(indexOfComma[1], indexOfComma[2]);
259   LAT = dataIn.substring(indexOfComma[2], indexOfComma[3]);
260   LONG = dataIn.substring(indexOfComma[3], indexOfComma[4]);
261   date_time.remove(0, 1);
262   LAT.remove(0, 1);  // DELETE THE #
263   LONG.remove(0, 1);
264
265 }
```

This function to split the data from the serial which comes from the GPS.

```
268 void SendLocation()
269 {
270   GSMSerial.println("AT+CGNSINF\r");
271   delay(1000);
272   if (GSMSerial.available() >=0) {
273     dataIn = GSMSerial.readString();
274     Serial.println("----------------------------dataIn--------------------------------");
275     Serial.println(dataIn);
276     Serial.println("----------------------------end---------------------------------");
277     checkIndex = dataIn.indexOf("+CGNSINF: 1,1,");
278     if ( checkIndex >= 0) {
279
280       split(dataIn);
281       Serial.print("DATE TIME :");
282       Serial.println(date_time);
283
284       Serial.print("LAT :");
285       Serial.println(LAT);
286
287       Serial.print("LONG:");
288       Serial.println(LONG);
289       int LATInt = 0;
290       int LONGInt = 0;
291       LATInt = LAT.toInt();
292       LONGInt = LONG.toInt();
```

This function to send the location via GPS, dataIn is the data from the GSM serial, after we got it, we split it using the previous function, then we convert the LAT and LONG to integer.

```
293     if (LATInt > 31 && LATInt < 34 && LONGInt > 32 && LONGInt < 37 ) {
294
295       Serial.println("Sending SMS");
296       //*************** send SMS GOOGLE MAPS LINK ******************
297
298       GSMSerial.print("AT+CMGF=1\r");    //Because we want to send the SMS in text mode
299       delay(100);
300       GSMSerial.print("AT+CMGS=\""+targetPhone+"\"\r");    //Start accepting the text for the message
301       delay(100);
302
303
304       GSMSerial.print("www.google.com.ph/maps/place/" + LAT + "," + LONG + "   ");
305       delay(100);
306       GSMSerial.write(0x1A);  //Equivalent to sending Ctrl+Z
307       delay(10000);
308       Serial.println("done sending");
309
310     }
```

If the range of the readings in reasonable position, it will send the position using GSM to the target phone.

```
311     else
312     {
313       Serial.println("unreasonable position");
314     }
315   }
316
317   else
318   {
319     Serial.println("No signal or check wiring");
320   }
321 }
322 else(Serial.println("Modem didn't respond!"));
```

```
325 void init_mpu6050() {
326   // Try to initialize!
327   if (!mpu.begin()) {
328     Serial.println("Failed to find MPU6050 chip");
329     while (1) {
330       delay(10);
331     }
332   }
333   Serial.println("MPU6050 Found!");
334
335   mpu.setAccelerometerRange(MPU6050_RANGE_16_G);
336   Serial.print("Accelerometer range set to: ");
337   switch (mpu.getAccelerometerRange()) {
338     case MPU6050_RANGE_2_G:
339       Serial.println("+-2G");
340       break;
341     case MPU6050_RANGE_4_G:
342       Serial.println("+-4G");
343       break;
344     case MPU6050_RANGE_8_G:
345       Serial.println("+-8G");
346       break;
347     case MPU6050_RANGE_16_G:
348       Serial.println("+-16G");
349       break;
350   }
```

```
351   mpu.setGyroRange(MPU6050_RANGE_2000_DEG);
352   Serial.print("Gyro range set to: ");
353   switch (mpu.getGyroRange()) {
354     case MPU6050_RANGE_250_DEG:
355       Serial.println("+- 250 deg/s");
356       break;
357     case MPU6050_RANGE_500_DEG:
358       Serial.println("+- 500 deg/s");
359       break;
360     case MPU6050_RANGE_1000_DEG:
361       Serial.println("+- 1000 deg/s");
362       break;
363     case MPU6050_RANGE_2000_DEG:
364       Serial.println("+- 2000 deg/s");
365       break;
366   }
```

```
368   mpu.setFilterBandwidth(MPU6050_BAND_260_HZ);
369   Serial.print("Filter bandwidth set to: ");
370   switch (mpu.getFilterBandwidth()) {
371     case MPU6050_BAND_260_HZ:
372       Serial.println("260 Hz");
373       break;
374     case MPU6050_BAND_184_HZ:
375       Serial.println("184 Hz");
376       break;
377     case MPU6050_BAND_94_HZ:
378       Serial.println("94 Hz");
379       break;
380     case MPU6050_BAND_44_HZ:
381       Serial.println("44 Hz");
382       break;
383     case MPU6050_BAND_21_HZ:
384       Serial.println("21 Hz");
385       break;
386     case MPU6050_BAND_10_HZ:
387       Serial.println("10 Hz");
388       break;
389     case MPU6050_BAND_5_HZ:
390       Serial.println("5 Hz");
391       break;
392   }
393 }
```

We initialized MPU6050, then we defined the parameters, Accelerometer range, Gyro range and the band width.

```
void get_mpu_readings() {
  /* Get new sensor events with the readings */
  sensors_event_t a, g, temp;
  mpu.getEvent(&a, &g, &temp);
  float x_deg;
  float y_deg;
  float z_deg;
  x_deg = a.acceleration.x;
  y_deg = a.acceleration.y;
  z_deg = a.acceleration.z;

  if (y_deg > 0.5) {
    blance_up();
  }
  if (y_deg < -0.5) {
    blance_down();
  }
  if (y_deg > -0.5 && (y_deg < 0.5 )) {
    blance_stop();
  }
```

This function is for MPU6050 readings, the value of Y degree is checked to know if the stair climbs up or down the stairs.

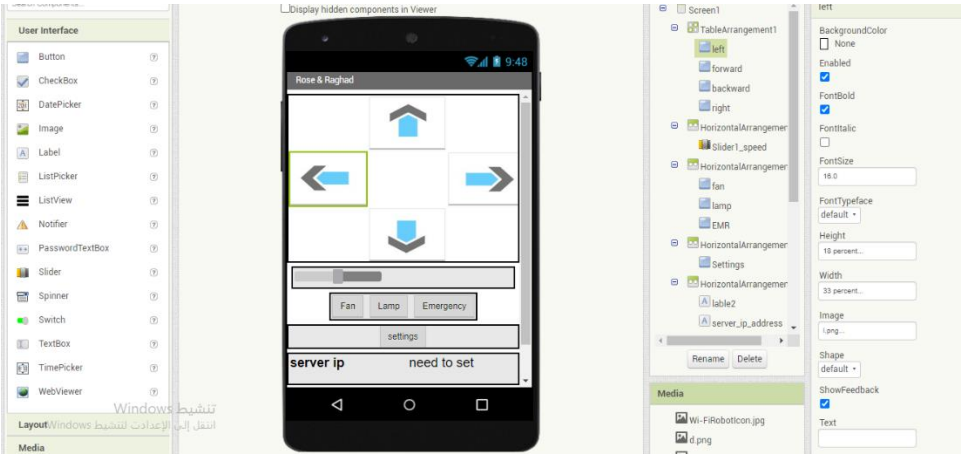
```
if (((x_deg > 6 || x_deg < -6)||(y_deg > 6 || y_deg < -6) )&& danger_flag == 0) {
  delay(100);
  if (((x_deg > 6 || x_deg < -6)||(y_deg > 6 || y_deg < -6) )&& danger_flag == 0) {
    delay(200);
        if (((x_deg > 6 || x_deg < -6)||(y_deg > 6 || y_deg < -6) )&& danger_flag == 0) {

  danger_flag = 1;
  check_flag = 1;

  Serial.print("Acceleration X: ");
  Serial.println(x_deg);
  Serial.print("Acceleration Y: ");
  Serial.println(y_deg);

}}}
if (x_deg < 5.5 && x_deg > -5.5 && y_deg < 5.5 && y_deg > -5.5) {
  check_flag = 0;
  danger_flag = 0;
```

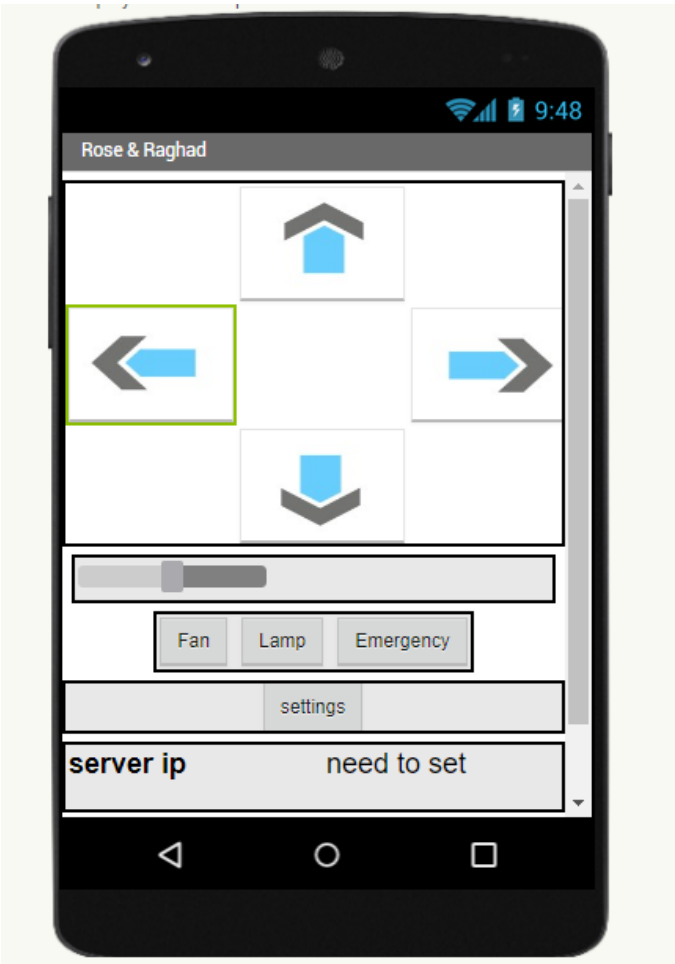
This part we used it to know if there is a fall in one of the four directions.

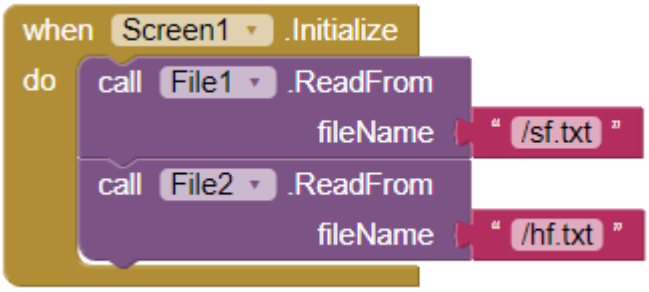
```
void blance_up() {
  digitalWrite(b1, 0);
  digitalWrite(b2, 1);
  //Serial.println("blance_up");

}
void blance_down() {
  digitalWrite(b1, 1);
  digitalWrite(b2, 0);
 // Serial.println("blance_down");
}
void blance_stop() {
  digitalWrite(b1, 0);
  digitalWrite(b2, 0);
  //Serial.println("blance_stop");
}
```

And these functions to balance the chair while climbing up or down the stair.

Our application using App inventor:



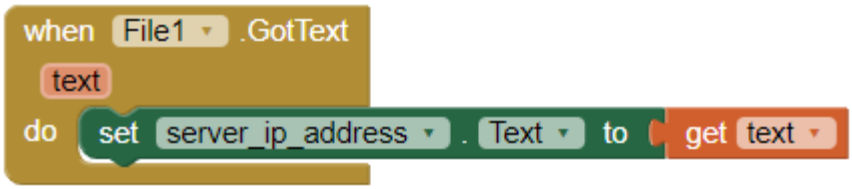


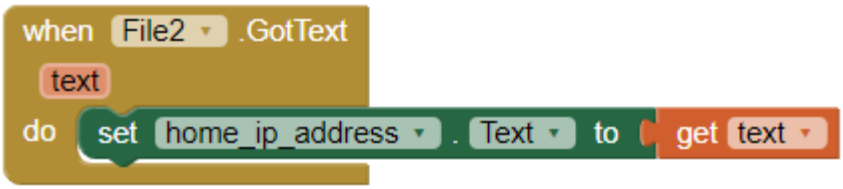


the system make call to file1 and , When screen1 initialized file 1 contains the ip address of server and file 2 , file2 kept in memory contains the ip address of home and they

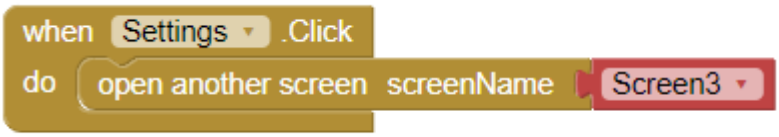
server file : Sf

home file : Hf



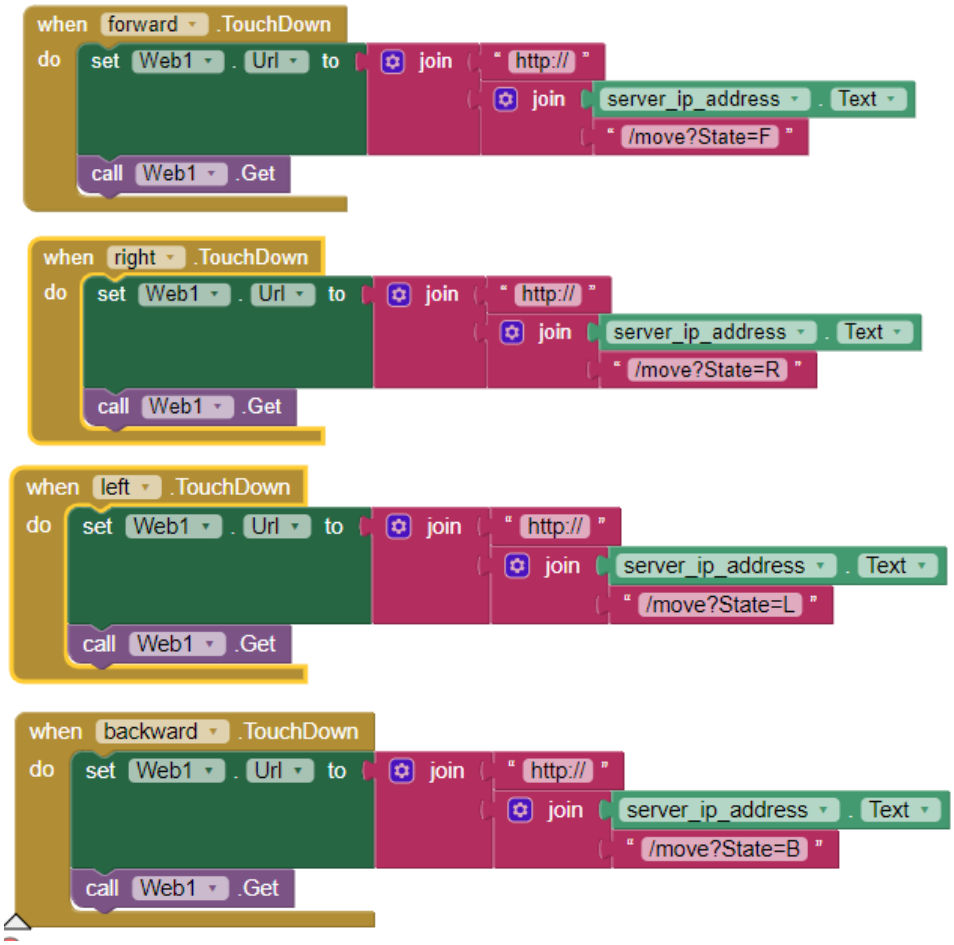


When the file1 or file2 contains text which is ip addresses then get the texts and put them so that , of server and home they appear in their own places in icons

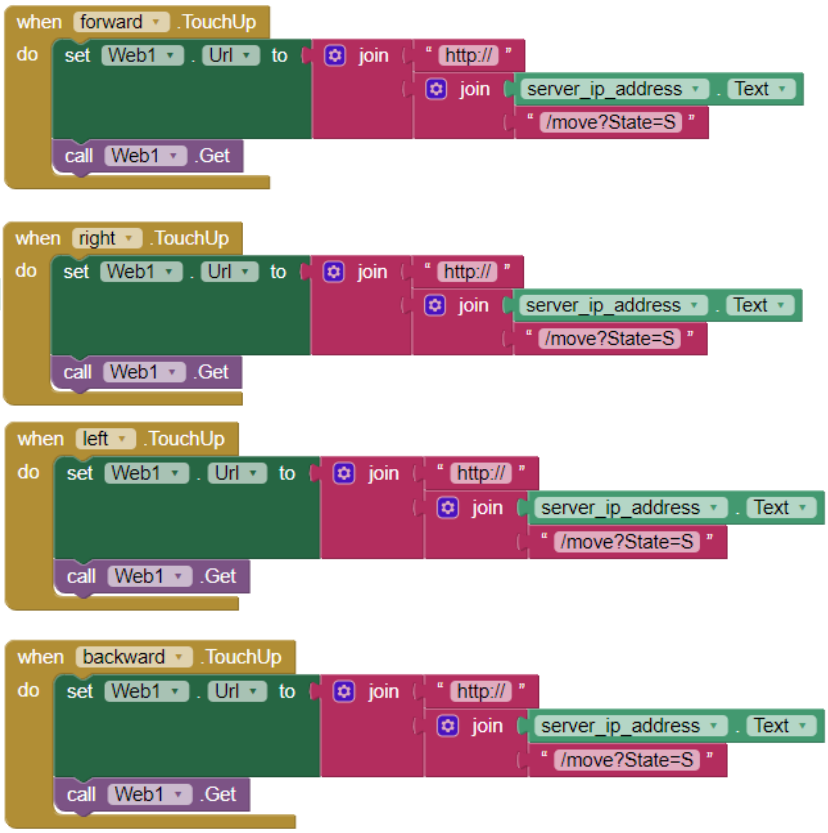


,it goes to screen 3 ,When we press the settings button which contains the ip addresses

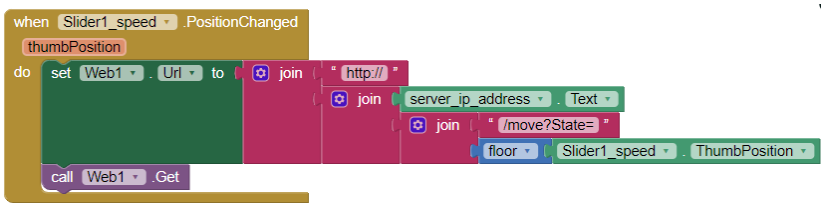
So that we can modify it



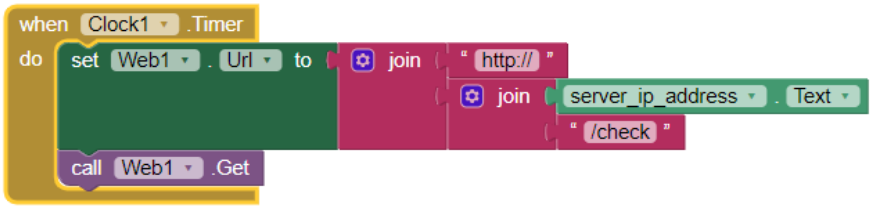
When we press the buttons :forward, backward, right, left, set the URL to (http://the ip address of server/move? State=F or B or R or L representing the directions or number between 0,9 representing the speed).



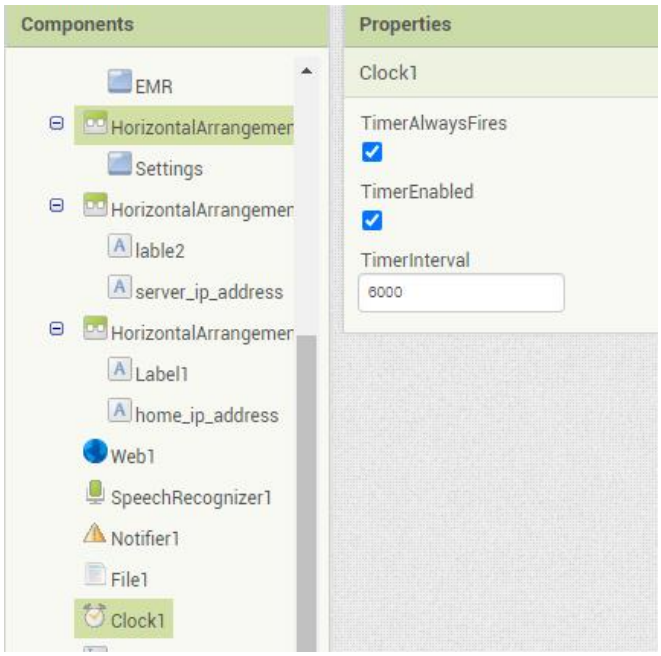
When the buttons touched up, then send URL, so stop the chair.

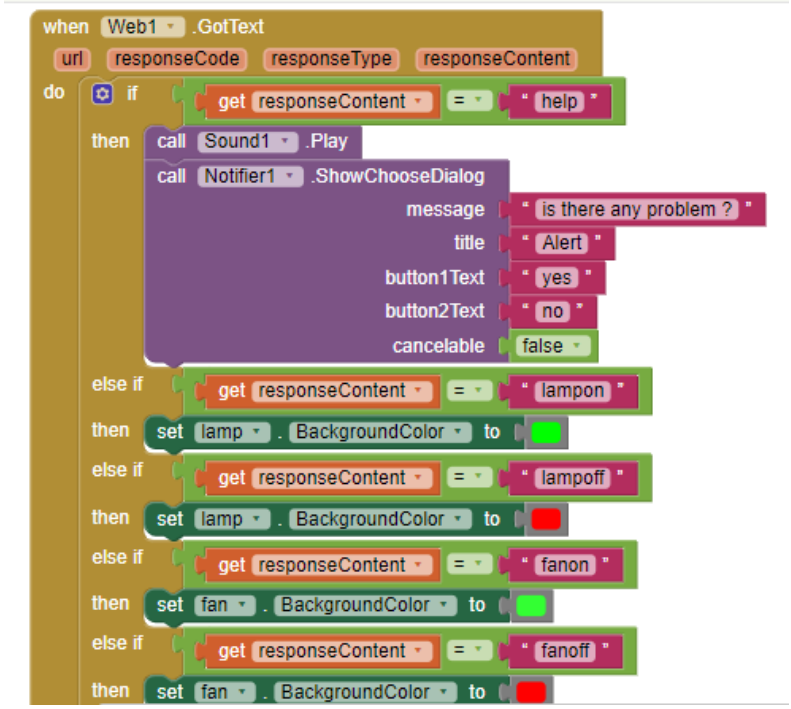


When we changed the silder position then the URLs' argument changes according to its thumb position(from 0 to 9)



There is a timer that send a 'check' request every 6 seconds to know if the chair in the normal.

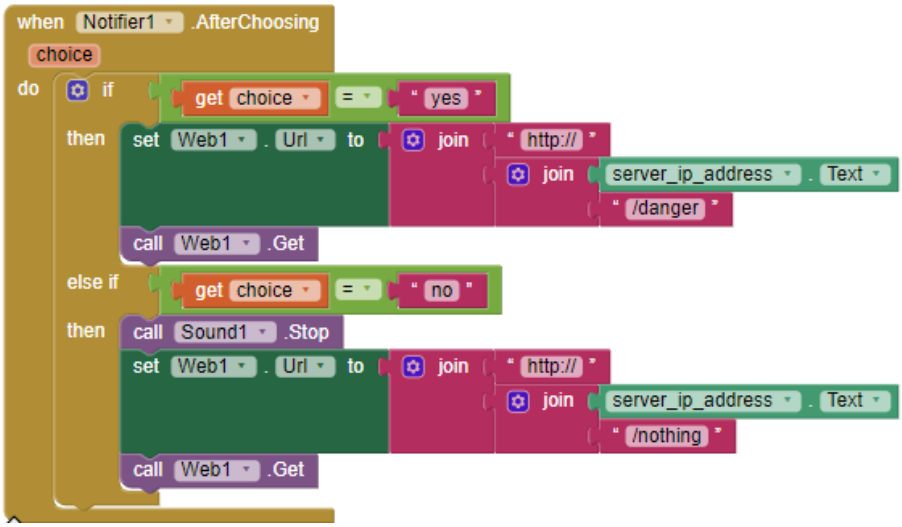




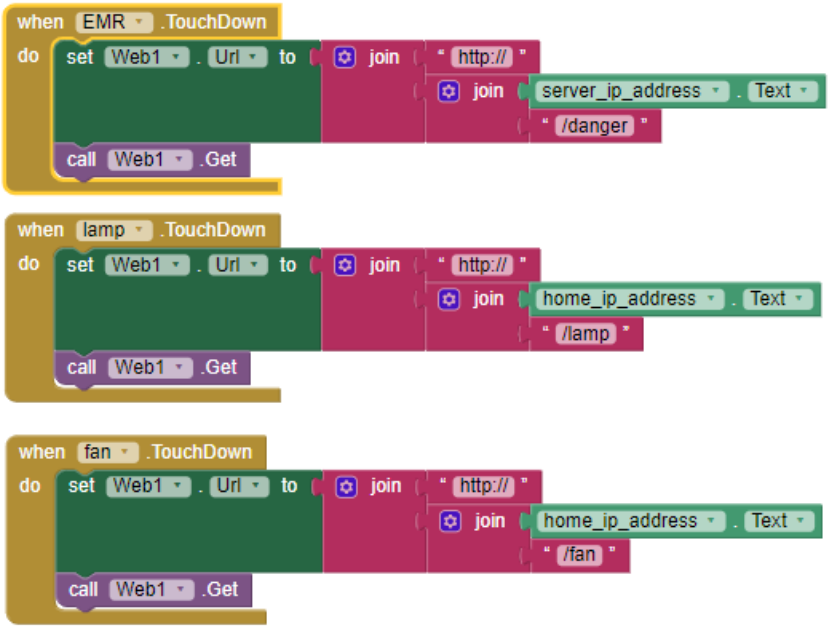
if web1 got a text from server of the chair then it checks the data , if its "help" it means the person in danger.

a notification appeared on the phone asks the person if he/she in danger.

And if the server got "lampon" then turn on the lamb and the same thing got the fan.

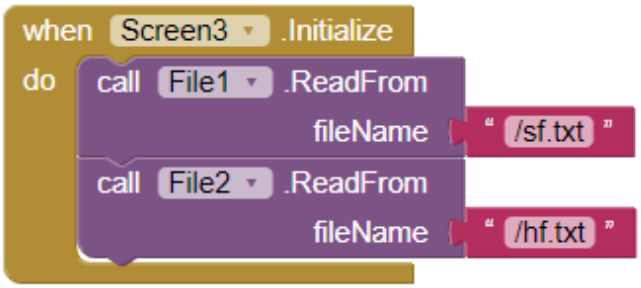


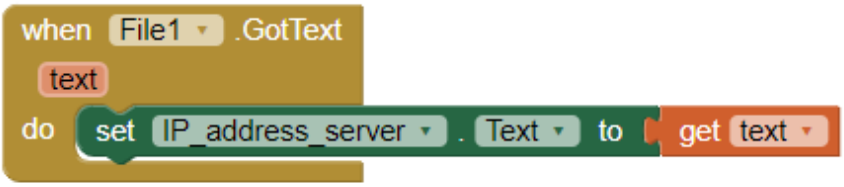
else if notifier 1 got yes, it means the person in danger nothing happened.

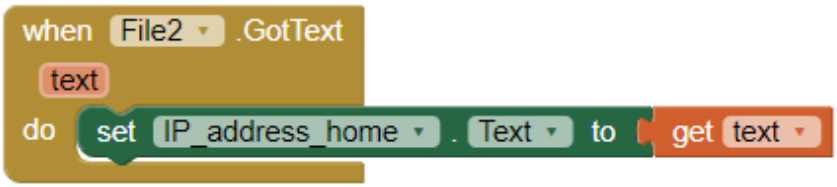


If we click emergency button in the application then send to web1 that the person in danger

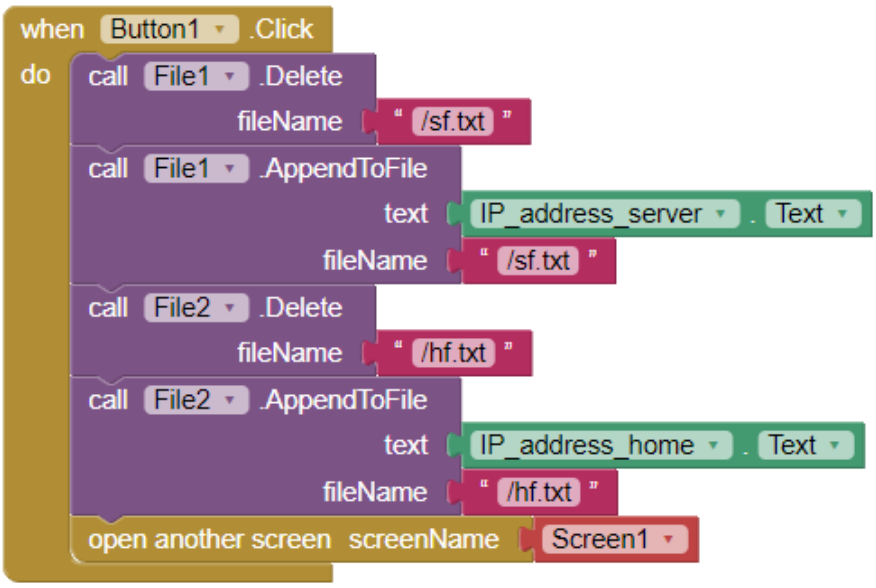
If we click lamp button in the application then send to web1 lamb and the same for fan







Screen3 it contains the settings (Ip addresses) ,we can update the Ip addresses from it.



here we can update and delete the Ip addresses in screen 3