A PROJECT REPORT PHASE II

VOICE-ACTIVATED SMART BLINDS USING MACHINE LEARNING

Project report submitted in partial fulfillment of the Requirements for the Award of the Degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

By

P VISHNU – SNC18CS025

ABHISHEK N – SNC17CS001

SREEHARI K – SNC18CS032

SALMATH S P – SNC18CS027

ADITH VINOD – SNC18CS002

ANASWARA RAJAN – LSNC18CS035

Under the Guidance of

Prof. NIMISHA M K

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SREE NARAYANA GURU COLLEGE OF ENGINEERING AND TECHNOLOGY, PAYYANUR

AFFILIATED TO A P J ABDUL KALAM TECHNOLOGICAL UNIVERSITY, KERALA

2021-2022

SREE NARAYANA GURU COLLEGE OF ENGINEERING AND TECHNOLOGY, PAYYANUR

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project report entitled **VOICE-ACTIVATED SMART BLINDS USING MACHINE LEARNING** submitted by **SREEHARI K (SNC18CS032)**, in the partial fulfillment for the award of the Degree of Bachelor of Technology in Computer Science and Engineering to **A P J ABDUL KALAM TECHNOLOGICAL UNIVERSITY, KERALA,** is a record of bonafied work carried out under my guidance and supervision.

PROJECT GUIDE

Prof. NIMISHA M K

Assistant Professor

Dept of Computer Science and Engineering

SNGCET, Payyanur

HEAD OF THE DEPARTMENT

Prof. SUNDER V

Assistant Professor

Dept of Computer Science and Engineering

SNGCET, Payyanur

DECLARATION

I,**SREEHARI K (SNC18CS032)** hereby declare that the dissertation entitled **VOICE-ACTIVATED SMART BLINDS USING MACHINE LEARNING** , submitted for the B.Tech Degree is my original work and the dissertation has not formed the basis for the award of any degree, associate ship, fellowship or any other similar titles.

SREEHARI K PAYYANUR

DATED SIGNATURE 07/06/2022

ACKNOWLEDGEMENT

Any mission never concludes without cordial co-operation from surroundings. At the very outset, I would like to give the first honors to God who gave the wisdom and knowledge for the success of this presentation. First, I thank our Management **SREE BAKTHI SAMVARDHINI YOGAM TALAP , KANNUR** , for having me provided with all the facilities in the college campus. Next, I thank our Principal, **Dr. SURESH PARETH** , for having me provided with all the facilities required for the success of this presentation. My sincere thanks to, **Prof. SUNDER V**, Head of the Department of CSE, SNGCET, Payyanur for his encouragement throughout this presentation. I take this opportunity to thank my guide **Prof. NIMISHA M K** , Assistant Professor, Department of CSE, SNGCET, Payyanur, for her encouragement throughout the presentation. I am expressing thankfulness to all the teaching and non-teaching staff of the Department of CSE, SNGCET, Payyanur , for their valuable guidance , help and co-operation and continued encouragement in each and every step of this project presentation. Last but not the least I thank my parents, friends and all my well-wishers who had supported me directly and indirectly during the project presentation.

Thanking you,
SREEHARI K

INDEX

TITLE **PAGE. NO**

LIST OF FIGURES

LIST OF SHORT FORMS

AC	Adaptive Controller
IOTF	IOT Framework
IOT	Internet Of Things
SRP	Speech Recognition Platform
VS CODE	Visual studio code
API	Application programe interface
CGI	Commom gateway interface
CSS	Cascading stylesheet
HTML	Hypertext markup language
PHP	Hypertext preprocessor

ABSTRACT

The emergence of the Internet of Things concept has provided a great vision for the technological future, intending to enable the extraction and comprehension of information from the environment around us, making use of the interaction and cooperation between several technological devices. The example of Smart Homes, in particular, aims to integrate voice activated smart blinds into households,institution,hospitals , enabling the automation of tasks previously performed by users, to simplify their daily lives and create a more comfortable environment. However,This device works beyond the users uncomfortable situations, being necessary reprogramming of the device to follow the new behaviors. Taking this problem into account, this article presents the design and end-to-end implementation of a voice-activated smart home blind for intelligent controller,deployed in a real environment and validated in an experimental setup of motorized blinds. The architecture of the proposed solution integrates evolvable intelligence with the use of an Online Learning framework, enabling it to automatically adapt to the user’s habits and behavioral patterns. The results obtained from the various evaluation tests provide a validation of the operation and usefulness of the developed system.Voice controlled smart blind based the continous updations of sensor readings.

CHAPTER 1
PROJECT DESCRIPTION

1.1 Module Description

1.1.1 Speech recognition platform

The smart home controller must work without an Internet connection Communication protocols and Websockets, with built-in support for HA .The Rhasspy allows configurig, programming, and testing the voice assistant . Speech to Text Text to Speech . And Wake Word detection . Transforming audio data into JavaScript Object Notation (JSON) events .The Various algorithms and computation techniques are used to recognize speech into text and improve the accuracy of transcription.

1.1.1.2 Voice Biometrics

This is a rapidly evolving field, also largely driven by AI, which can now be used to enable workflows and collaboration. Biometrics takes many forms other modes, such as retina scans or fingerprinting, have been in use for years but voice has some distinct benefits that align well with a touchless workplace.

1.1.1.3 Speech to text

Like biometrics, this is an application, but among speech recognition use cases, it has great value for collaboration when used with any voice-enabled endpoint that has a keypad. Speech to text (STT) is one of many AI use cases that have become good enough now for everyday use in the workplace, and it's an ideal application for office- or home-based work.

1.1.2 IOT framework:

For the IOT framework HA platform was used . Ability to track control and automate Smart device . HA's Lovelace dashboard can used . 'area' and 'position' are sent from Rhaspy as Slots in the JSON event . Smart blinds are installed and the position they’re required to change to. To change the default dashboard, create a new file ui-lovelace.yaml in your configuration directory and add the following section to your configuration.yaml and restart Home Assistant.Multiple Dashboards we can define multiple dashboards in Home Assistant. Each dashboard can be added to the sidebar. This makes it possible to create separate control dashboards for each individual part of your house. You can manage your dashboards via the user interface. Go to Configuration ->

Dashboards. Here you can see all defined dashboards and create new ones. Using YAML for the default dashboard. To change the default dashboard, create a new file ui-lovelace.yaml in your configuration directory and add the following section to your configuration.yaml and restart Home Assistant: lovelace: mode: yaml YAML is a good way to start this file is to copy and paste the “Raw configuration” from the UI so your manual configuration starts the same as your existing UI.Click Overview in your sidebar. Click the three dots menu (top-right) and click on Edit Dashboard. Click the three dots menu again and click on Raw configuration editor. There you see the configuration for your current dashboard. Copy that into the <config>/ui-lovelace.yaml file. Once you take control of your UI via YAML, the Home Assistant interface for modifying it won’t be available anymore and new entities will not automatically be added to your UI.When you make changes to ui-lovelace.yaml, you don’t have to restart Home Assistant or refresh the page. Just hit the refresh button in the menu at the top of the UI. To revert back to using the UI to edit your dashboard, remove the lovelace section from your configuration.yaml and copy the contents of your ui-lovelace.yaml into the raw configuration section of Home Assistant and restart.

1.1.3 Smart blinds:

Position controller and IOT interface . Universal Asynchronous Receiver Transmitter (UART) . Aduino Nano ,a rotary encoder and a three-relay module . The IOT interface is composed of a Node MCU and four sensors . ESPH home platform was used to program the node MCU .This research aims to implement an IoT stick that will view the image of opportunity, autonomy, and certainty. The proposed smart stick is planned with an impediment identification module, a global positioning system (GPS), pit and flight of stairs detection, water detection, and a global system for mobile communication (GSM) to perform their daily activities quickly. The impediment identification module utilizes an ultrasonic sensor alongside a water level sensor to distinguish the obstructions that insinuate recognizing the obstacles and identifying the obstructions pattern. An Arduino ATmega328 is used to advise the weakened people about the barriers and sends notifications using an earphone and a buzzer. The current location of the blind person is located using GPS and GSM modules. The stick activates an alert system in case of loss. Several test cases prove that the functionalities introduced with the stick are performing correctly. Such a stick will be a blessing for blind people having a positive impact on science and technology.

1.1.4 Adaptive controller:

For interpreting the behavioural patterns . Implemented and different ML model is also created . Undergo training period without sending predictions to the user . Pediction refer to the position the blind will be in next timestamp . Recognise of one set of smart blind fail to communicate.When designing adaptive control systems, special consideration is necessary of convergence and robustness issues. Lyapunov stability is typically used to derive control adaptation laws and show .Self-tuning of subsequently fixed linear controllers during the implementation phase for one operating point;Self-tuning of subsequently fixed robust controllers during the implementation phase for whole range of operating points; Self-tuning of fixed controllers on request if the process behaviour changes due to ageing, drift, wear, etc.; Adaptive control of linear controllers for nonlinear or time-varying processes; Adaptive control or self-tuning control of nonlinear controllers for nonlinear processes; Adaptive control or self-tuning control of multivariable controllers for multivariable processes (MIMO systems); Usually these methods adapt the controllers to both the process statics and dynamics. In special cases the adaptation can be limited to the static behavior alone, leading to adaptive control based on characteristic curves for the steady-states or to extremum value control, optimizing the steady state. Hence, there are several ways to apply adaptive control algorithms.A particularly successful application of adaptive control has been adaptive flight control.[7][8] This body of work has focused on guaranteeing stability of a model reference adaptive control scheme using Lyapunov arguments. Several successful flight-test demonstrations have been conducted, including fault tolerant adaptive control

1.2 System Flow Diagram

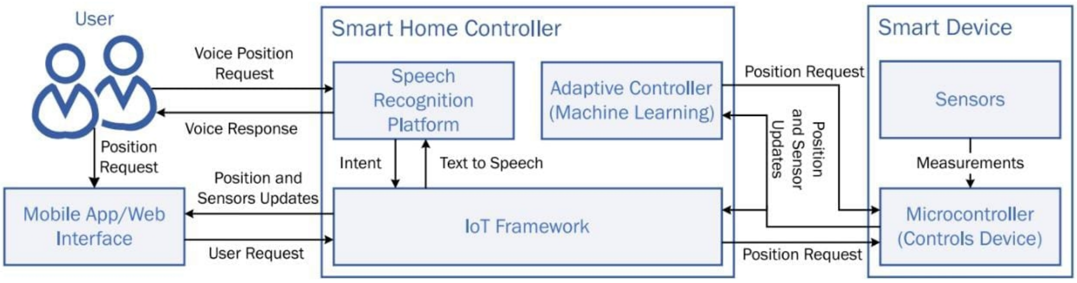


Fig1.2 : Smart blinds architecture overview

CHAPTER 2
CODING DETAILS

2.1 DEVELOPMENT TOOLS AND PROGRAMMING LANGUAGES :

2.1.1 C++

C++ is one of the world's most popular programming languages.And it can be found in today's operating systems, Graphical User Interfaces, and embedded systems.it is an object-oriented programming language which gives a clear structure to programs and allows code to be reused, lowering development costs.It is portable and can be used to develop applications that can be adapted to multiple platforms.Itis fun and easy to learn.As C++ is close to C# and Java, it makes it easy for programmers to switch to C++ or vice versa.

2.1.1 HTML

The HyperText Markup Language or HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such asCascading Style Sheets(CSS) and scripting languages such as JavaScript.Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documentsby denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by *tags*, written using angle brackets. Tags such as `<img />` and`<input />` directly introduce content into the page. Other tags such as `<p>`surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags but use them to interpret the content of the page.

2.1.3 JAVA SCRIPT

JavaScript often abbreviated JS, is a programming language that is one of the core technologies of the World Wide Web, alongside HTML and CSS. As of 2022, 98% of websites use JavaScript on the client side for web page behavior often incorporating third-party libraries.All major web browsers have a dedicated JavaScript engine to execute the code on users' devices.JavaScript is a high-level, often just-in-time compiled language that conforms to the ECMAScript standard. It has dynamic typing, prototype-based object-orientation, and first-class functions. It is multi-paradigm, supportingevent-driven,functional,and imperative programming styles. It has application programming interfaces (APIs) for working with text, dates,regular expressions, standarddata structures, and the Document Object Model(DOM).

2.1.3 PHP

PHP is a general-purpose scripting language geared toward web development. It was originally created by Danish-Canadian programmer Rasmus Lerdorf in 1994.The PHPreference implementationis now produced by The PHP Group.PHP originally stood for Personal Home Page,but it now stands for therecursive initialism PHP: Hypertext Preprocessor.PHP code is usually processed on a web serverby a PHP interpreter implemented as a module, a daemon or as a Common Gateway Interface(CGI) executable. On a web server, the result of the interpreted and executed PHP code – which may be any type of data, such as generated HTML or binaryimage data– would form the whole or part of an HTTP response. Various web template systems, web content management systems, and web frameworks exist which can be employed to orchestrate or facilitate the generation of that response.

2.1.4 CSS

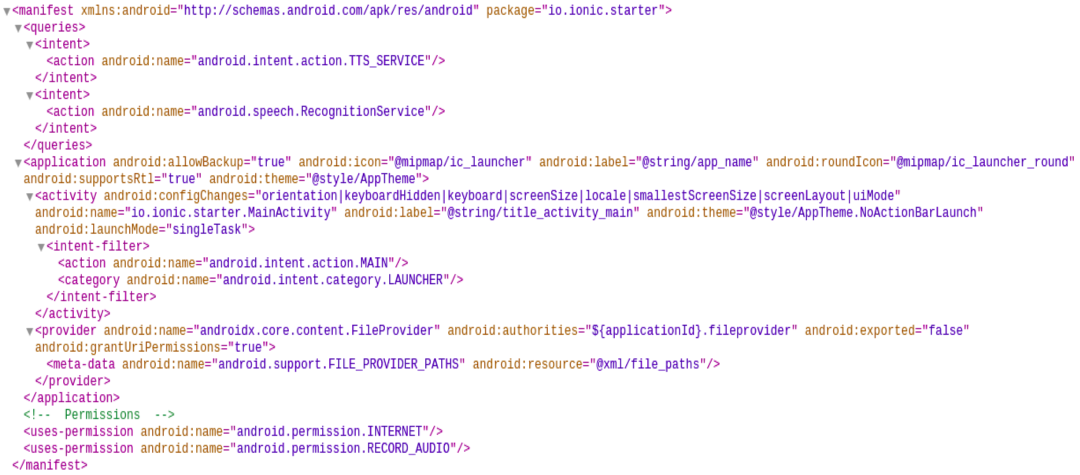
Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML or XML (including XML dialects such as SVG, MathML or XHTML).CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.CSS is designed to enable the separation of presentation and content, including layout,colors, and fonts.This separation can improve content accessibility; provide more flexibility and control in the specification of presentation characteristics; enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, which reduces complexity and repetition in the structural content; and enable the .css file to be cachedto improve the page load speed between the pages that share the file and its formatting.

2.1.5 VS CODE

Visual Studio Code, also commonly referred to as VS Code,is a source-code editor made by Microsoft for Windows,Linux and macOS.Features include support for debugging,syntax highlighting,intelligent code completion,snippets,code refactoring, and embedded Git. Users can change the theme,keyboard shortcuts, preferences, and install extensions that add additional functionality.In the Stack Overflow 2021 Developer Survey, Visual Studio Code was ranked the most popular developer environment tool, with 70% of 82,000 respondents reporting that they use it.

2.2 CODING DETAILS OF SMART BLINDS SYSTEM

2.2.1 DEPLOYING VOICE-ACTIVATED SMART BLINDS TO APP INTERFACE

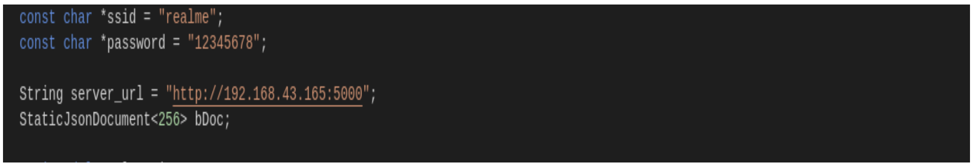


```
<manifest xmlns:android="http://schemas.android.com/apk/res/android" package="io.ionic.starter">
  <queries>
    <intent>
      <action android:name="android.intent.action.TTS_SERVICE"/>
    </intent>
    <intent>
      <action android:name="android.speech.RecognitionService"/>
    </intent>
  </queries>
  <application android:allowBackup="true" android:icon="@mipmap/ic_launcher" android:label="@string/app_name" android:roundIcon="@mipmap/ic_launcher_round"
  android:supportsRtl="true" android:theme="@style/AppTheme">
    <activity android:configChanges="orientation|keyboardHidden|keyboard|screenSize|locale|smallestScreenSize|screenLayout|uiMode"
    android:name="io.ionic.starter.MainActivity" android:label="@string/title_activity_main" android:theme="@style/AppTheme.NoActionBarLaunch"
    android:launchMode="singleTask">
      <intent-filter>
        <action android:name="android.intent.action.MAIN"/>
        <category android:name="android.intent.category.LAUNCHER"/>
      </intent-filter>
    </activity>
    <provider android:name="androidx.core.content.FileProvider" android:authorities="${applicationId}.fileprovider" android:exported="false"
    android:grantUriPermissions="true">
      <meta-data android:name="android.support.FILE_PROVIDER_PATHS" android:resource="@xml/file_paths"/>
    </provider>
  </application>
  <!--  Permissions  -->
  <uses-permission android:name="android.permission.INTERNET"/>
  <uses-permission android:name="android.permission.RECORD_AUDIO"/>
</manifest>
```

Fig2.2.1: Fig:Integrating voice-activated smart blinds with front page

To deploy the system to app,HTML code is used.An manifest tag is used that includes the source as the voice-activated smart blinds API.It can be deployed to any website by performing these API call.

2.2.2 IMPORTING LIBRARIES AND PACKAGES



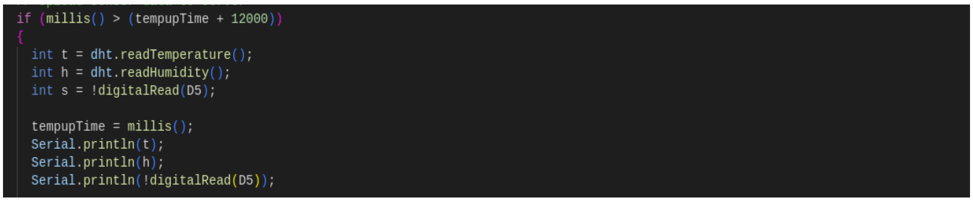
```
const char *ssid = "realme";
const char *password = "12345678";

String server_url = "http://192.168.43.165:5000";
StaticJsonDocument<256> bDoc;
```

Fig 2.2.2:Importing WifiClient library

Wificlient library is free software; you can redistribute it and/or modify it under the terms of the gnu lesser general public license as published by the free software foundation; either version 2.1 of the license, or (at your option) any later version. this library is distributed in the hope that it will be useful, but without any warranty; without even the implied warranty of merchantability or fitness for a particular purpose. See the gnu lesser general public license for more details.

2.2.3 COLLECTING THE SENSOR READINGS



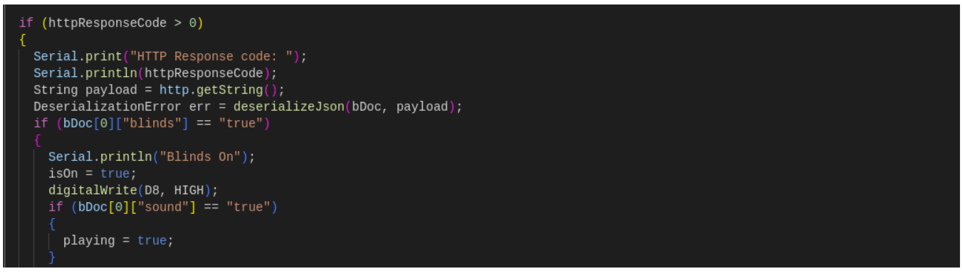
```
if (millis() > (tempupTime + 12000))
{
  int t = dht.readTemperature();
  int h = dht.readHumidity();
  int s = !digitalRead(D5);

  tempupTime = millis();
  Serial.println(t);
  Serial.println(h);
  Serial.println(!digitalRead(D5));
```

Fig 2.2.3:Upload sensor data to server

The sensor readings is constantly reads every 10 seconds.These input are uploaded to the server.

2.2.4 PREDICTING THE PRESENCE OF FIRE



```
if (httpResponseCode > 0)
{
  Serial.print("HTTP Response code: ");
  Serial.println(httpResponseCode);
  String payload = http.getString();
  DeserializationError err = deserializeJson(bDoc, payload);
  if (bDoc[0]["blinds"] == "true")
  {
    Serial.println("Blinds On");
    isOn = true;
    digitalWrite(D8, HIGH);
    if (bDoc[0]["sound"] == "true")
    {
      playing = true;
    }
```

Fig 2.2.4:System predict the presence of fire

The model analyses the sensor readings and predicts the presence of fire.Else case fire is not detected and blinds and alarm in off state.

CHAPTER 3
SYSTEM TESTING

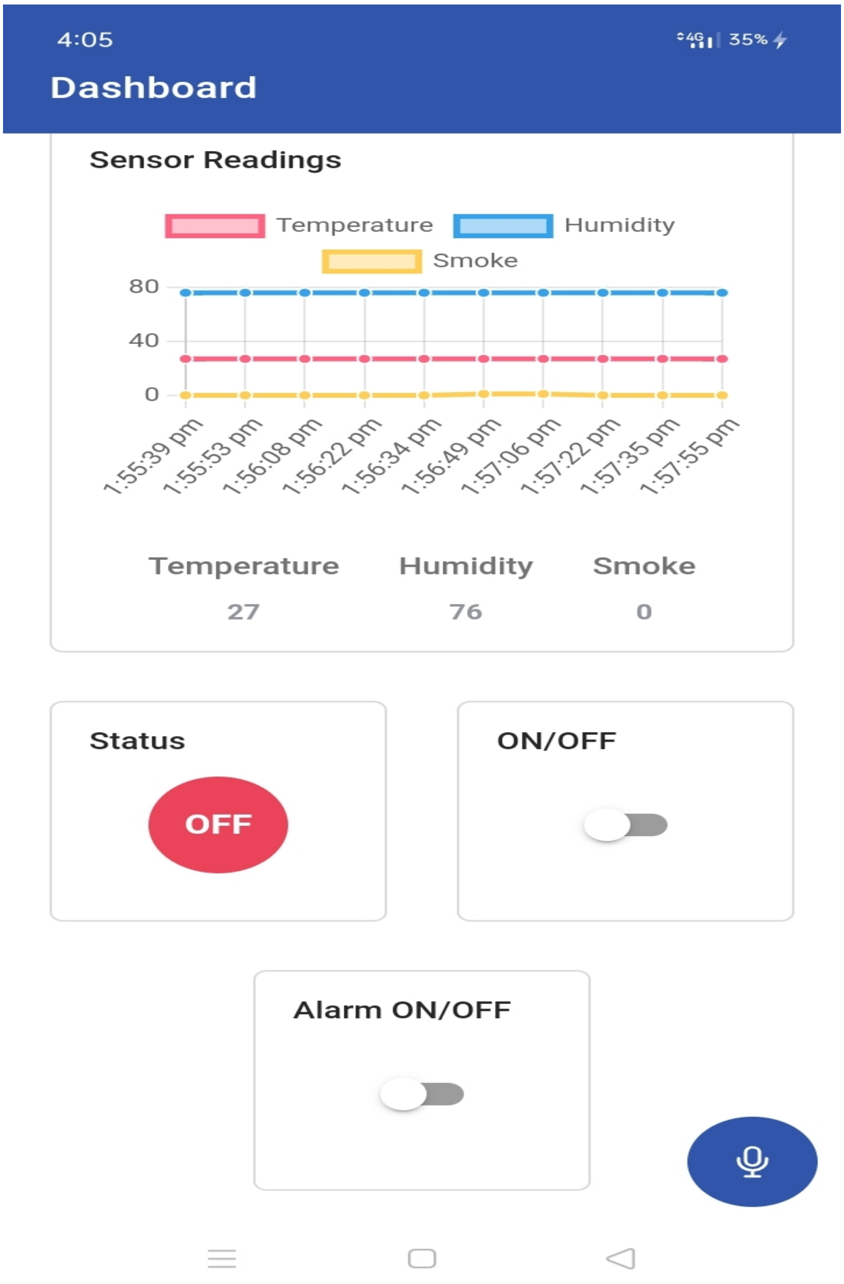
Test plan is necessary for any project to plan the testing phase and decide the scope of the project. Test plan involves collecting design specifications about the project, wiring test cases, executing them manually or automatically using automated testing tools. Testing any application is highly important. Test plan is a method of documenting the test cases, specification plans and other basic level details about how the application works. Test Activities for this project includes various testing like:

- Black Box testing: In this project, sample test cases are written and manual testing is done to check the functionality of the application.
- White Box testing: Once the application meets the user requirements and functionalities according to the test cases, its internal logic are completely tested to ensure that the application does not have any logical errors or issues.
- Unit Testing: I have tested all the modules of the application individually by running as a test program.
- Integration testing: After testing the modules individually, tested them by integrating all the sub modules, modules into one application.
- System Testing: It refers to checking whether the system in which the application is built meets the necessary requirements like software support. For example: In this project, I have checked whether the device in which the application developed is compatible with the software (Android Studio).
- End to End Testing: Tested the complete environment of application by connecting the device with different machines, installing as an APK file, with the database and in local network.

CHAPTER 4

SYSTEM IMPLEMENTATION

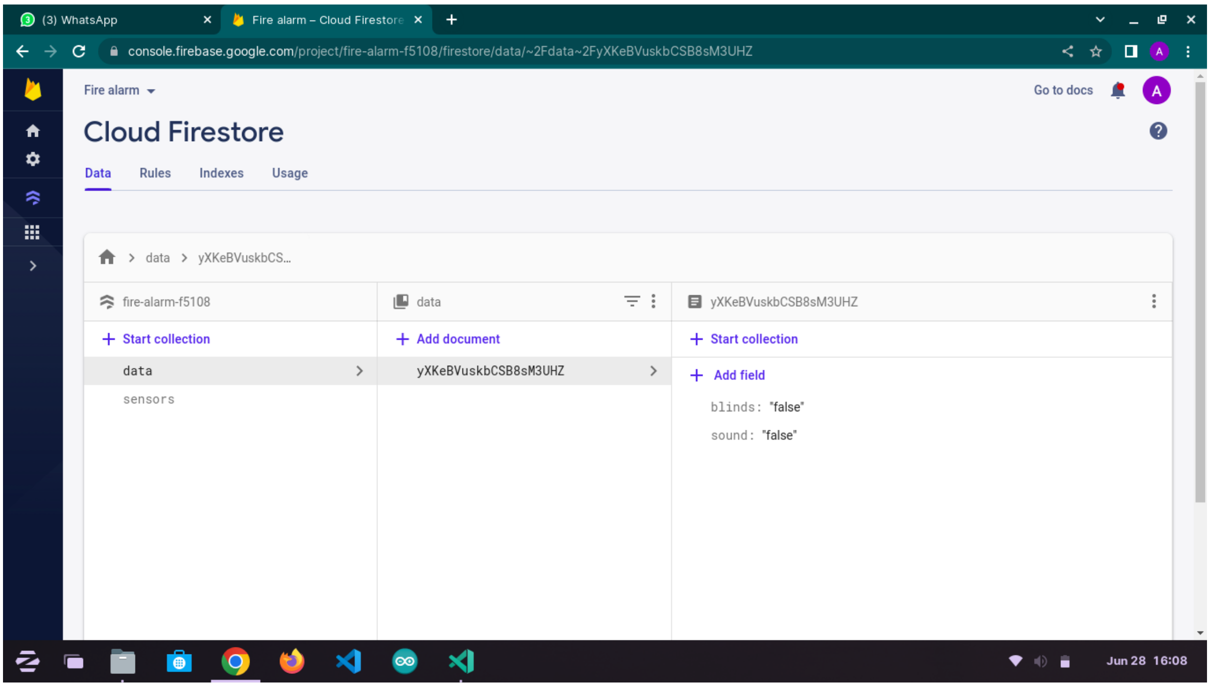
4.1 SCREENSHOT OF VOICE ACTIVATED FORNT PAGE



Temperature	Humidity	Smoke
27	76	0

Fig 4.1: Voice activated smart blinds's front page

4.2 SCREENSHOT OF CLOUD FIRESTORE DATA COLLECTION



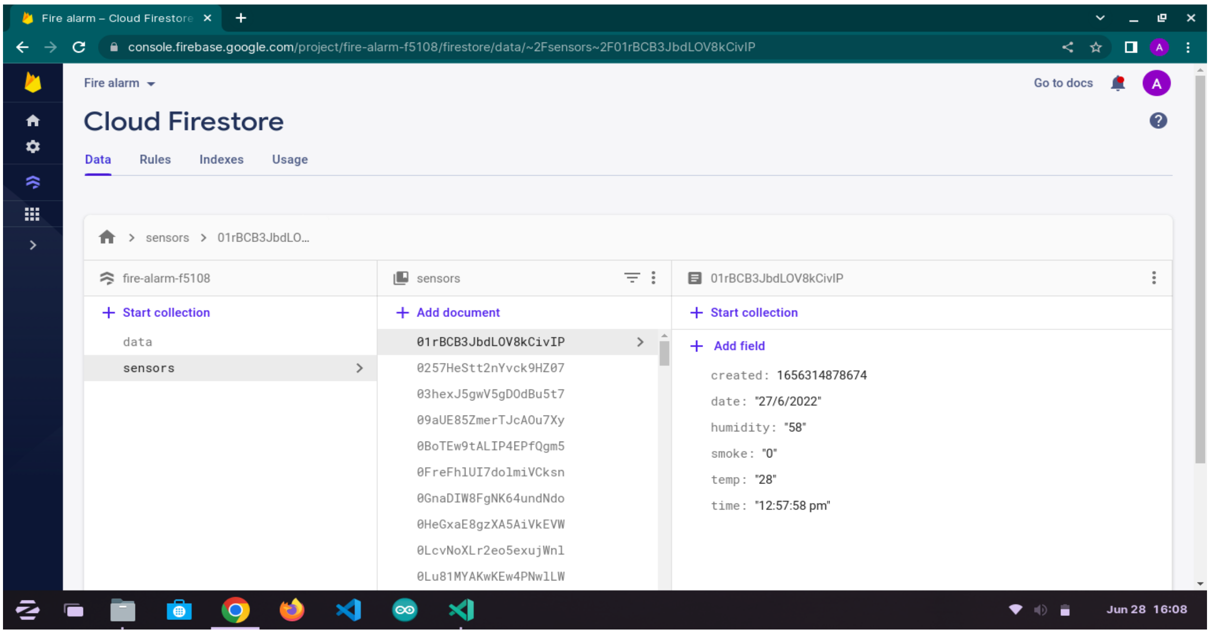
```
data
sensors

yXKeBVuskbCSB8sM3UHZ

blinds: "false"
sound: "false"
```

Fig 4.2 : Cloud firestore data collection

4.3 SCREENSHOT OF CLOUD FIRESTORE SENSORS COLLECTION



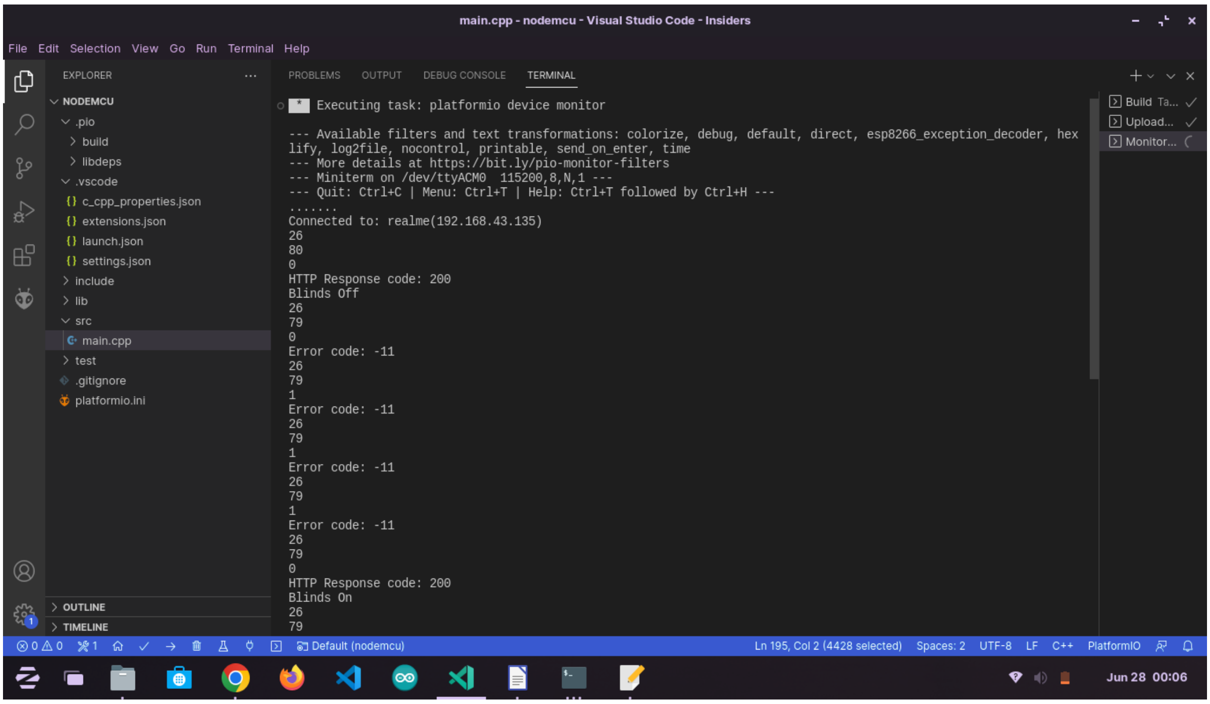
```
data
sensors

01rBCB3JbdLOV8kCivIP
0257HeStt2nYvck9HZ07
03hexJ5gwV5gDOdBu5t7
09aUE85ZmerTJcAOu7Xy
0BoTEw9tALIP4EPfQgm5
0FreFhlUI7dolmiVCksn
0GnaDIW8FgNK64undNdo
0HeGxaE8gzXA5AiVkEVW
0LcvNoXLr2eo5exujWnl
0Lu81MYAKwKEw4PNwlLW

created: 1656314878674
date: "27/6/2022"
humidity: "58"
smoke: "0"
temp: "28"
time: "12:57:58 pm"
```

fig 4.3 : Cloud firestore sensors collection

4.4 SCREENSHOT OF PRINTING RESULTS



```
* Executing task: platformio device monitor

--- Available filters and text transformations: colorize, debug, default, direct, esp8266_exception_decoder, hex
lify, log2file, nocontrol, printable, send_on_enter, time
--- More details at https://bit.ly/pio-monitor-filters
--- Miniterm on /dev/ttyACM0  115200,8,N,1 ---
--- Quit: Ctrl+C | Menu: Ctrl+T | Help: Ctrl+T followed by Ctrl+H ---
.......
Connected to: realme(192.168.43.135)
26
80
0
HTTP Response code: 200
Blinds Off
26
79
0
Error code: -11
26
79
1
Error code: -11
26
79
1
Error code: -11
26
79
1
Error code: -11
26
79
0
HTTP Response code: 200
Blinds On
26
79
```

Fig 4.4 : Printing results

CHAPTER 5
CONCLUSION AND FUTURE WORKS

5.1 CONCLUSION

The emergence of these devices has driven Internet of Things’ vision of intelligent environments such as smart buildings and smart homes. Increasingly, consumers are bringing home intelligent devices such as televisions, light bulbs, fans, heating systems, or even blinds. To further simplify the integration of all these devices, smart home hubs have emerged, often integrated with a personal assistant.being now possible to carry out a large number of tasks without much effort, often at a distance of a simple voice control , or even through simple automations that do not require any input from the user. Many of these devices are still based on classical and low intelligence programming. Developing smart home solutions with devices like these becomes a complex process, with the software having to be customized for every different home and family, and having to be updated as the family’s lifestyle changes. The majority of users do not have the capabilities necessary to address the programming task, and the alternative of hiring professionals to perform such upgrades is quite expensive and inconvenient. As a hypothetical solution to this problem, it is proposed the use of Online Learning techniques, an area of ML, to create a smart home controller capable of adapting to the habits and preferences of the user and capable of evolving with the same. In this work, a smart home controller was used to control motorized home blinds, creating a functional and convenient smart blinds solution.

5.2 FUTURE SCOPE

In the future,we try to implement the system in hospitals,institutions,Home, industries for as better safety system like warning or presafety or precaution system. We need to modify buzzer sound as doesn't make noise in the situation of every time changing in fire detection,it's enable whenever the possibility of higher intensity of the fire. And must authenticate the system when the user using the App interface,need to store the user's voice for the future verification at the time of entering the interface.

CHAPTER 6
APPENDICES

6.1 SAMPLE CODE

6.1.1 CODE FOR SMART BLINDS SYSTEM

```
#include <Arduino.h>
#include <WiFiClient.h>
#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>
#include <ArduinoJson.h>
#include <Stepper.h>
#include <DHT.h>
#include <Adafruit_Sensor.h>
#define DHTPIN D4
#define DHTTYPE DHT11
const char *ssid = “realme”;
const char *password = “12345678”;
String server_url = “http://192.168.43.165:5000”;
StaticJsonDocument<256> bDoc;
unsigned long lastTime = 0;
unsigned long dataTime = 0;
unsigned long buzzTime = 0;
unsigned long tempupTime = 0;
unsigned long alarmTime = 0;
unsigned long timerDelay = 5000;
static bool playing;
static bool upload;
static bool isOn;
DHT dht(DHTPIN, DHTTYPE);
void setup()
```

```
{
 pinMode(D5, OUTPUT);
 pinMode(D6, OUTPUT);
 pinMode(D8, OUTPUT);
 pinMode(D5, INPUT);
Serial.begin(115200);
WiFi.begin(ssid, password);
Serial.print(“Connecting to: “);
Serial.println(ssid);
while (WiFi.status() != WL_CONNECTED)
{
delay(500);
Serial.print(“.”);
}
Serial.println(“”);
Serial.print(“Connected to: “);
Serial.println(WiFi.SSID() + ‘(‘ + WiFi.localIP().toString() + ‘)’);
}
void loop()
{
Buzzer
if (millis() > (buzzTime + 600))
{
if (playing)
{
buzzTime = millis();
digitalWrite(D6, !digitalRead(D6));
}
else
{
digitalWrite(D6, LOW);
}
}
if (millis() > (alarmTime + 10000))
{
alarmTime = millis();
int temp = dht.readTemperature();
int hum = dht.readHumidity();
if ((temp > 20) && (hum < 80) && !digitalRead(D5))
{
 WiFiClient client;
HTTPClient http;
String serverPath = server_url + “/data/yXKeBVuskbCSB8Sm3UHZ”;
http.begin(client, serverPath.c_str());
http.addHeader(“Content-Type”, “application/json”);
String Data = “{\”blinds\”:\”true\”,\”sound\”:\”true\”}”;
http.PUT(Data.c_str());
}
else
{
WiFiClient client;
HTTPClient http;
String serverPath = server_url + “/data/yXKeBVuskbCSB8Sm3UHZ”;
http.begin(client, serverPath.c_str());
http.addHeader(“Content-Type”, “application/json”);
String Data = “{\”blinds\”:\”false\”,\”sound\”:\”false\”}”;
http.PUT(Data.c_str());
}
lastTime = millis();
}
if (millis() > (tempupTime + 12000))
```

```
{
int t = dht.readTemperature();
int h = dht.readHumidity();
int s = !digitalRead(D5);
tempupTime = millis();
Serial.println(t);
Serial.println(h);
Serial.println(!digitalRead(D5));
if (WiFi.status() == WL_CONNECTED)
{
WiFiClient client;
HTTPClient http;
String serverPath = server_url + “/sensor”;
http.begin(client, serverPath.c_str());
http.addHeader(“Content-Type”, “application/json”);
String Data = “{\”smoke\”:\”” + String(s) + “\”,\”temp\”:\”” + String(t) + “\”,\”humidity\”:\”” + String(h) + “\”}”;
http.POST(Data.c_str());
}
}
if (millis() > (dataTime + 14000))
{
 dataTime = millis();
if (WiFi.status() == WL_CONNECTED)
{
WiFiClient client;
HTTPClient http;
String serverPath = server_url + “/data”;
http.begin(client, serverPath.c_str());
int httpResponseCode = http.GET();
if (httpResponseCode > 0)
{
```

```
Serial.print(“HTTP Response code: “);
Serial.println(httpResponseCode);
String payload = http.getString();
DeserializationError err = deserializeJson(bDoc, payload);
if (bDoc[0][“blinds”] == “true”)
{
Serial.println(“Blinds On”);
isOn = true;
digitalWrite(D8, HIGH);
if (bDoc[0][“sound”] == “true”)
{
playing = true;
}
else
{
playing = false;
}
}
else if (bDoc[0][“blinds”] == “false”)
{
Serial.println(“Blinds Off”);
playing = false;
isOn = false;
digitalWrite(D8, LOW);
}
else
{
Serial.println(“WiFi Disconnected.”);
}
if (err)
{
Serial.print(“Error: “);
Serial.println(err.c_str());
return;
}
}
else
{
Serial.print("Error code: ");
Serial.println(httpResponseCode);
 }
http.end();
}
else
{
Serial.println("WiFi Disconnected.");
}
lastTime = millis();
}
}
```

6.1.2 CODE FOR APP INTERFACE

6.1.2.1 HOME PAGES FOR DASHBOARD SENSOR READINGS

6.1.2.1.1 HOME.MODULE.TS

```
import { NgModule } from '@angular/core';
import { CommonModule } from '@angular/common';
import { IonicModule } from '@ionic/angular';
import { FormsModule } from '@angular/forms';
import { NgChartsModule } from 'ng2-charts';
import { HomePage } from './home.page';
import { HomePageRoutingModule } from './home-routing.module';
@NgModule({
```

```
imports: [
CommonModule,
FormsModule,
IonicModule,
HomePageRoutingModule,
NgChartsModule,
],
declarations: [HomePage]
})
export class HomePageModule {}
```

6.1.2.1.2 HOME.PAGE.HTML

```
{{c.label}}
{{c.data[c.data.length - 1]}}
Status {{ alarmStatus.blind == 'true' ? "ON" : "OFF" }} ON/OFF Alarm ON/OFF
```

6.1.2.1.3 HOME.PAGE.SCSS

```
.values {
display: flex;
justify-content: space-evenly;
.item {
text-align: center;
h2 {
font-size: 16px;
margin-top: 0;
}
p {
font-size: 14px;
font-weight: bold;
margin-bottom: 0;
margin-top: 0.8rem;
color: var(--ion-color-medium)
}
}
}
ion-card {
box-shadow: none;
border: 1px solid #dadada;
border-radius: 6px;
}
ion-card-title {
font-size: 16px !important;
}
.status {
display: flex;
align-items: center;
justify-content: center;
height: 100%;
span {
display: flex;
align-items: center;
justify-content: center;
height: 60px;
width: 60px;
border-radius: 30px;
font-weight: bold;
color: white;
font-size: 18px;
}
}
.buttons {
display: flex;
}
```

```
@media only screen and (max-width: 600px) {
.buttons {
ion-card {
width: 50%;
}
}
}
```

6.1.2.1.4 HOME.PAGE.SPEC.TS

```
import { ComponentFixture, TestBed, waitForAsync } from '@angular/core/testing';
import { IonicModule } from '@ionic/angular';
import { HomePage } from './home.page';
describe('HomePage', () => {
let component: HomePage;
let fixture: ComponentFixture<HomePage>;
beforeEach(waitForAsync(() => {
TestBed.configureTestingModule({
declarations: [ HomePage ],
imports: [IonicModule.forRoot()]
}).compileComponents();
fixture = TestBed.createComponent(HomePage);
component = fixture.componentInstance;
fixture.detectChanges();
}));
it('should create', () => {
expect(component).toBeTruthy();
});
});
```

6.1.2.1.5 HOME.PAGE.TS

```
import {
```

```
ChangeDetectorRef,
Component,
ElementRef,
ViewChild,
} from '@angular/core';
import { SpeechRecognition } from '@capacitor-community/speech-recognition';
import { ChartOptions } from 'chart.js';
import { threadId } from 'worker_threads';
import { SensorService } from '../services/sensor.service';
export interface SensorData {
date: number;
humidity: number;
light: number;
temp: number;
time: string;
}
export interface ChartItems {
data: Array<number>;
label: string;
}
@Component({
selector: 'app-home',
templateUrl: 'home.page.html',
styleUrls: ['home.page.scss'],
})
export class HomePage {
@ViewChild('lineCanvas') lineCanvas: ElementRef;
 chartData: Array<ChartItems> = [];
chartLabels: Array<string> = [];
alarmStatus: any;
isRec: boolean;
speechText: string = '';
```

```
chartOptions: ChartOptions = {
 responsive: true,
};
constructor(
private sensorService: SensorService,
private changeDetector: ChangeDetectorRef
) {
SpeechRecognition.requestPermission();
}
remove() {
this.chartLabels.shift();
this.chartData[0].data.shift();
this.chartData[1].data.shift();
this.chartData[2].data.shift();
this.chartData = [...this.chartData];
this.chartLabels = [...this.chartLabels];
}
async startRec() {
this.isRec = true;
const { available } = await SpeechRecognition.available();
if (available) {
SpeechRecognition.start({
popup: false,
partialResults: true,
language: 'en-US',
})
SpeechRecognition.addListener('partialResults', (data: any) => {
if(data && data.value.length > 0) {
this.speechText = data.value[0];
this.changeDetector.detectChanges();
if(data.value[0] == 'turn off sound') {
this.offSound(this.alarmStatus.id, this.alarmStatus.sound);
this.stopRec();
}else if(data.value[0] == 'turn off device') {
this.offSafety(this.alarmStatus.id, this.alarmStatus.blind);
this.stopRec();
}else if(data.value[0] == 'turn on device') {
this.offSafety(this.alarmStatus.id, this.alarmStatus.blind);
this.stopRec();
}
})
}
}
async stopRec() {
this.isRec = false;
await SpeechRecognition.stop();
}
getReading() {
this.sensorService.getTemp().subscribe({
next: (res) => {
if (res.length < 10 || this.chartData.length < 1) {
if (this.chartData.length < 1 || res.length < 1) {
this.chartLabels = Object.keys(res).map((a) => res[a].time);
this.chartData = [
{
data: Object.keys(res).map((a) => res[a].temp),
label: 'Temperature',
},
{
data: Object.keys(res).map((a) => res[a].humidity),
label: 'Humidity',
},
{
data: Object.keys(res).map((a) => res[a].light),
label: 'Light',
},
];
} else {
this.chartLabels = Object.keys(res).map((a) => res[a].time);
this.chartData = [
{
data: Object.keys(res).map((a) => res[a].temp),
label: 'Temperature',
},
{
data: Object.keys(res).map((a) => res[a].humidity),
label: 'Humidity',
},
{
data: Object.keys(res).map((a) => res[a].light),
label: 'Light',
},
];
}
} else {
console.log('third');
this.chartLabels.shift();
this.chartData[0].data.shift();
this.chartData[1].data.shift();
this.chartData[2].data.shift();
this.chartLabels.push(res[res.length - 1].time);
this.chartData[0].data.push(res[res.length - 1].temp);
this.chartData[1].data.push(res[res.length – 1].humidity);
this.chartData[2].data.push(res[res.length - 1].light);
this.chartData = [...this.chartData];
this.chartLabels = [...this.chartLabels];
```

```
}
},
error: (err) => {
console.log(err);
},
});
}
getAlarm() {
this.sensorService.getAlarm().subscribe({
next: (res) => {
this.alarmStatus = res[0];
console.log(this.alarmStatus);
},
error: (err) => {
console.log(err);
},
});
}
offSound(id: string, status: string) {
this.sensorService.offSound(id, status).then((res) => {
console.log(res);
});
}
offSafety(id: string, status: string) {
this.sensorService.offSafety(id, status).then((res) => {
console.log(res);
});
}
ngAfterViewInit(): void {
this.getReading();
this.getAlarm();
}
```

```
}
```

6.1.2.1.6 HOME-ROUTING.MODULE.TS

```
import { NgModule } from '@angular/core';
import { RouterModule, Routes } from '@angular/router';
import { HomePage } from './home.page';
const routes: Routes = [
{
path: '',
component: HomePage,
}
];
@NgModule({
imports: [RouterModule.forChild(routes)],
exports: [RouterModule]
})
export class HomePageRoutingModule {}
```

BIBLIOGRAPHY

[1] Leandro Filipe, Ricardo Silva Peres, R. Tavares, “Voice-Activated Smart Home Controller Using Machine Learning,”IEEEAccess·May 2021, Available: https://www.researchgate.net/publication/351255997

[2] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, ‘‘Internet of Things: A survey on enabling technologies, protocols, and applications,’’ IEEE Commun. Surveys Tuts., vol. 17, no. 4, pp. 2347–2376, 4thQuart.,2015.[Online].Available: https://ieeexplore.ieee.org/document/7123563/

[3] B. Qolomany, A. Al-Fuqaha, A. Gupta, D. Benhaddou, S. Alwajidi, J. Qadir, and A. C. Fong, ‘‘Leveraging machine learning and big data for smart buildings: A comprehensive survey,’’ IEEE Access, vol. 7,pp. 90316–90356, 2019.

[4] D. Liciotti, M. Bernardini, L. Romeo, and E. Frontoni, ‘‘A sequential deep learning application for recognising human activities in smart homes,’’ Neurocomputing, vol. 396, pp. 501–513, Jul. 2020.

[5] D. Schweizer, M. Zehnder, H. Wache, H.-F. Witschel, D. Zanatta, and M. Rodriguez, ‘‘Using consumer behavior data to reduce energy consumption in smart homes: Applying machine learning to save energy without lowering comfort of inhabitants,’’ in Proc. IEEE 14th Int. Conf. Mach. Learn. Appl. (ICMLA), Dec. 2015, pp. 1123–1129.

[6] M. C. Mozer, ‘‘The neural network house: An environment that adapts to its inhabitants,’’ in Proc. AAAI Spring Symp. Intell. Environ., Dec. 1998, pp. 110–114.

[7] M.-S. Pan and C.-J. Chen, ‘‘Intuitive control on electric devices by smartphones for smart home environments,’’ IEEE Sensors J., vol. 16, no. 11, pp.4281–4294,Jun.2016.[Online].Available: http://ieeexplore.ieee.org/document/7433919/

[8] A. Javed, H. Larijani, A. Ahmadinia, R. Emmanuel, M. Mannion, and D. Gibson, ‘‘Design and implementation of a cloud enabled random neural network-based decentralized smart controller with intelligent sensor nodes for HVAC,’’ IEEE Internet Things J., vol. 4, no. 2, pp. 393–403, Apr. 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7740096/

[9] J. Reyes-Campos, G. Alor-Hernández, I. Machorro-Cano, J. O. Olmedo-Aguirre, J. L. Sánchez-Cervantes, and L. Rodríguez-Mazahua, ‘‘Discovery of resident behavior patterns using machine learning techniques and IoT paradigm,’’ Mathematics, vol. 9, no. 3, p. 219, Jan. 2021. [Online]. Available: https://www.mdpi.com/2227-7390/9/3/219

[10] V. S. Babu, U. A. Kumar, R. Priyadharshini, K. Premkumar, and S. Nithin, ‘‘An intelligent controller for smart home,’’ in Proc. Int. Conf. Adv. Comput., Commun. Informat. (ICACCI), Sep. 2016, pp. 2654–2657. [Online]. Available: http://ieeexplore.ieee.org/document/7732459/

[11] S. Bajpai and D. Radha, ‘‘Smart phone as a controlling device for smart home using speech recognition,’’ in Proc. Int. Conf. Commun. Signal Process. (ICCSP),Apr.2019,pp.0701–0705. [Online]. Available:https://ieeexplore.ieee.org/document/8697923/

[12] G. M. Madhu and C. Vyjayanthi, ‘‘Implementation of cost effective smart home controller with Android application using node MCU and Internet of Things (IOT),’’ in Proc. 2nd Int. Conf. Power, Energy Environ., Towards SmartTechnol.(ICEPE),Jun.2018,pp.1–5.[Online]. Available:https://ieeexplore.ieee.org/document/8659128/

[13] A. F. Abbas and M. Z. Abdullah, ‘‘Design and implementation of tracking a user’s behavior in a smart home,’’ in Proc. IOP Conf. Ser., Mater. Sci.Eng., Feb.2021,vol.1094,no.1,Art.no.012008. [Online]. Available:https://iopscience.iop.org/article/10.1088/1757-899X/1094/1/012008

[14] A. D. Giorgio and L. Pimpinella, ‘‘An event driven smart home co troller enabling consumer economic saving and automated demand side management,’’ Appl. Energy, vol. 96, pp. 92–103, Aug. 2012, doi: 10.1016/j.apenergy.2012.02.024.

[15] A. Natani, A. Sharma, and T. Perumal, ‘‘Sequential neural networks for multi-resident activity recognition in ambient sensing smart homes,’’ Appl. Intell., Jan. 2021. [Online].Available: http://link.springer.com/10.1007/s10489-020-02134-z

[16] H. Alemdar, H. Ertan, O. D. Incel, and C. Ersoy, ‘‘ARAS human activity datasets in multiple homes with multiple residents,’’ in Proc. 7th Int. Conf. Pervasive Comput. Technol. Healthcare Workshops (PervasiveHealth), May 2013, pp. 232–235.

[17] M. Safyan, S. Sarwar, Z. U. Qayyum, M. Iqbal, S. Li, and M. Kashif, ‘‘Machine learning based activity learning for behavioral contexts in Internet of Things (IoT),’’ Program. Comput. Softw., vol. 46, no. 8, pp. 626–635, Dec. 2020.[Online].Available: http://link.springer.com/10.1134/S0361768820080204

[18] D. J. Cook, M. Youngblood, E. O. Heierman, K. Gopalratnam, S. Rao, A. Litvin, and F. Khawaja, ‘‘MavHome: An agent-based smart home,’’ in Proc. 1st IEEE Int. Conf. Pervas. Comput. Commun. (Per-Com), Mar. 2003, pp. 521–524. [Online]. Available: http://ieeexplore.ieee.org/document/1192783/

[19] A. Paulauskaite-Taraseviciene, N. Morkevicius, A. Janaviciute, A. Liutkevicius, A. Vrubliauskas, and E. Kazanavicius, ‘‘The usage of artificial neural networks for intelligent lighting control based on resident’s behavioural pattern,’’ Elektronika ir Elektrotechnika,vol. 21, no. 2, pp. 72–79, Apr. 2015. [Online]. Available: http://eejournal.ktu.lt/index.php/elt/article/view/8772

[20] I. B. P. P. Dinata and B. Hardian, ‘‘Predicting smart home lighting behavior from sensors and user input using very fast decision tree with kernel density estimation and improved Laplace correction,’’ in Proc. Int. Conf.Adv. Comput. Sci. Inf. Syst., Oct. 2014, pp. 171–175. [Online]. Available:http://ieeexplore.ieee.org/document/7065885/

[21] D. J. Cook, A. S. Crandall, B. L. Thomas, and N. C. Krishnan, ‘‘CASAS:A smart home in a box,’’ Computer, vol. 46, no. 7, pp. 62–69, Jul. 2013.[Online]. Available: http://ieeexplore.ieee.org/document/6313586/

[22] P. Domingos and G. Hulten, ‘‘Mining high-speed data streams,’’ in Proc.6th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD).New York, NY, USA: Association Computing Machinery, 2000, pp. 71–80, doi: 10.1145/347090.347107.

[23] F. Sakr, F. Bellotti, R. Berta, and A. De Gloria, ‘‘Machine learning on mainstream microcontrollers,’’ Sensors, vol. 20, no. 9, p. 2638, May 2020.[Online]. Available: https://www.mdpi.com/1424-8220/20/9/2638

[24] T. Chai and R. R. Draxler, ‘‘Root mean square error (RMSE) or mean absolute error (MAE)?– arguments against avoiding RMSE in the literature,’’Geoscientific Model Develop., vol. 7, no. 3, pp. 1247–1250, Jun. 2014.[Online]. Available: https://gmd.copernicus.org/articles/7/1247/2014/