A Real Time Research Project/ Societal Related Project Report

On

SMART HOME AUTOMATION USING ALEXA AND GOOGLE ASSISTANT

Submitted in fulfillment of the requirements for the award of the

Bachelor of Technology

In

Department of Electronics and Communication Engineering

By

G. Vasantha	**21241A06E9**
M. Nanditha	**21241A04G6**
M.Uday Kiran	**21241A04G9**

Under the Esteemed guidance of

Ms. A. Lavanya

Assistant Professor

Department of Electronics and communication Engineering

GOKARAJU RANGARAJU INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

Bachupally, Kukatpally, Hyderabad, Telangana, India, 500090

2023-2024

GOKARAJU RANGARAJU

INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

CERTIFICATE

This is to certify that the Real Time Research Project/ Societal Related Project entitled "**ESP32 Smart Home Automation Using Alexa and Google Assistant** " is submitted by **G.Vasantha (21241A04E9) , M. Nanditha (21241A04G6) , M. Uday Kiran (21241A04G9)** in fulfillment of the award of a degree in BACHELOR OF TECHNOLOGY in Electronics and Communication Engineering during the academic year **2023-2024.**

INTERNAL GUIDE	HEAD OF THE DEPARTMENT
Ms. A. Lavanya	**Dr. Ch.Usha Kumari**
Asst.Professor	**Professor**

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

Many people helped us directly and indirectly to complete our project successfully. We would like to take this opportunity to thank one and all. First, we wish to express our deep gratitude to our internal guide **Ms. A. Lavanya**, **Asst.Professor** , Department of ECE for her support in the completion of our project report. We wish to express our honest and sincere thanks to **Mov. Pavan and Y. Sudharshan Reddy** for coordinating in conducting the project reviews, **Dr.Ch. Usha Kumari, HOD,** Department of ECE for providing resources, and to the principal **Dr. J. Praveen** for providing the facilities to complete our Real Time Research Project/ Societal Related Project. We would like to thank all our faculty and friends for their help and constructive criticism during the project completion phase. Finally, we are very much indebted to our parents for their moral support and encouragement to achieve goals.

G. Vasantha 21241A04E9
M. Nanditha 21241A04G6
M.Uday Kiran 21241A04G9

DECLARATION

We hereby declare that the Real Time Research Project/ Societal Related Project entitled "**Smart Home Automation Using Alexa and Google Assistant**" is the work done during the period from **2023-2024** and is submitted in the fulfillment of the requirements for the award of the degree of Bachelor of Technology in Electronics and Communication Engineering from **Gokaraju Rangaraju Institute of Engineering and Technology (Autonomous under Jawaharlal Nehru Technology University, Hyderabad).** The results embodied in this project have not been submitted to any other university or Institution for the award of any degree or diploma.

G. Vasantha	**21241A04E9**
M. Nanditha	**21241A04G6**
M. Uday Kiran	**21241A04G9**

ABSTRACT

The way we engage with our living spaces has changed dramatically as a result of the incorporation of voice-activated virtual assistants, like Google Assistant and Alexa, into smart home systems. This project showcases a cutting-edge smart home automation system built on the ESP32 that is easily connected to Google Assistant and Alexa. The technology improves accessibility and convenience by enabling users to remotely manage a variety of household appliances, lights, and sensors using basic voice commands. The goal of this project is to use the capabilities of the ESP32 microcontroller technology to develop a smooth and effective voice-activated device control system. The project uses Firebase and Kodular cloud applications, which are flexible platforms that connect ESP32 to Google Assistant and Alexa while guaranteeing compatibility and interoperability, to accomplish this integration.

By combining the capabilities of voice-controlled virtual assistants with ESP32 technology, this project demonstrates the potential to create an intuitive and intelligent automation solution that enhances comfort, efficiency, and security in modern households. Overall, this project demonstrates the potential of combining voice-controlled virtual assistants with smart home technology to create an intuitive and intelligent automation solution that enhances convenience, comfort, and efficiency in modern households.

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW OF THE PROJECT

Home automation is the process of automating appliances and other components of the house so that they may be controlled remotely, from a computer, or even from a phone. the concept of smart homes has revolutionized the way we interact with our living spaces. Automation technologies integrated with voice assistants like Alexa and Google Assistant have made it possible to control various home devices and systems effortlessly using voice commands.

This Internet of Things project builds a smart home using an ESP32, Firebase, Kodular, and Google Assistant. Alexa and Google Assistant can operate domestic appliances. Additionally, we can use the Kodular Companion App to control the relay from anywhere in the world. W may still use the manual switches to operate the relay module in the case that the internet is down.

The system is built around the ESP32 microcontroller, a powerful yet low-cost solution known for its adaptability and IoT features. The ESP32 will act as the main hub for establishing Wi-Fi connections between different household appliances, including fans, lighting, and other gadgets, and the internet. These appliances will be managed by a relay module, which will enable the ESP32 to turn them on and off in response to commands from the cloud and local interactions. Alexa and Google Assistant integration will enable users to control their smart home devices using natural language voice commands.

1.2 OBJECTIVE

Enhance homeowners' convenience, energy efficiency, and security by allowing voice commands, seamless integration with well-known voice assistants like Alexa and Google Assistant, and remote access and control of home appliances via a mobile app. With the use of cloud-based services like Firebase, the project aims to highlight how IoT can be used practically in everyday situations for managing and syncing data in real-time across various devices.to use Firebase for cloud-based storage and synchronization, ensuring real-time updates and remote access so that customers can monitor and control their devices from anywhere. This allows users to monitor and control their smart home devices with natural language commands. need to use Firebase as a backend service for command processing, synchronization between many platforms and devices, and real-time data storage. With Kodular, develop a user-friendly mobile application that functions as an easy-to-use interface for scheduling, managing, and monitoring linked devices. It should also have manual control options, status alerts, and scheduling features.

CHAPTER 2
SYSTEM REQUIRMENTS

As we all know IOT is a major factor in our future way of life. Our garage door, garden lighting, watering, feeding, and other features will all be managed and observed remotely, from a distance of kilometers away from the final destination. In this instance, our means of communication will be the internet. Thus, we may monitor or control our belongings from anywhere in the globe by connecting them to the internet.

In this project, an ESP32 and an Android app will be used to operate the household appliances that are linked to the relay system.

HARDWARE COMPONENTS:

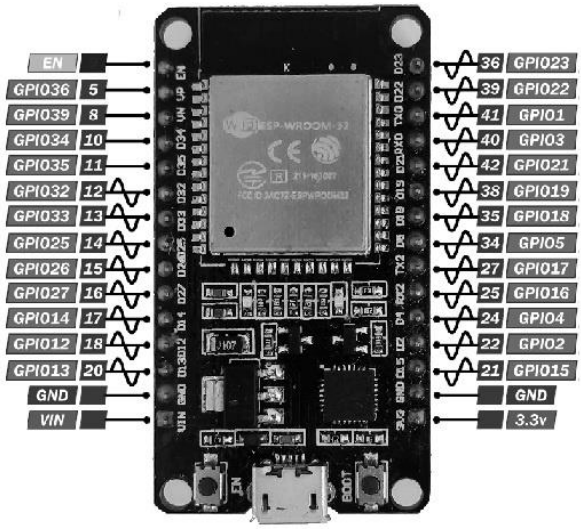
- ESP32 devkit Wroom 32
- 1 channel relay
- Bulb holder
- Bulb
- 230 v AC supply
- USB Cable

SOFTWARE COMPONENTS:

- Firebase
- Kodular
- Arduino IDE
- Kodular Companion Mobile App

2.1 HARDWARE COMPONENTS

ESP32 MICROCONTROLLER



ESP 32 Fig.2.1

With integrated Wi-Fi and Bluetooth, the ESP32 is a potent microcontroller from Espress if Systems that can be used in a wide range of applications. It is extensively utilized in wearable electronics, home automation, Internet of Things (IoT) devices, and other embedded systems.

The Espress if Systems-developed ESP32 is a very flexible microcontroller that is well-known for having integrated Bluetooth and Wi-Fi. It is extensively utilized in wearable electronics, home automation systems, Internet of Things devices, and numerous other embedded applications.

General Specifications

- Microcontroller: Dual-core Tensilica LX6 microprocessor
- Clock Frequency: Up to 240 MHz
- Wi-Fi: 802.11 b/g/n (2.4 GHz) v
- Bluetooth: Bluetooth v4.2 BR/EDR and BLE v
- Memory: 520 KB SRAM, up to 16 MB external SPI flash
- Interfaces: UART, SPI, I2C, I2S, ADC, DAC
- GPIO: 36 GPIO pins Operating Voltage: 3.3V

The pin configurations used in the project:

PIN	PIN NO
GND	GND
Vin	Vin
RELAY	13

PIN CONFIGURATION Table no 2.1

RELAY MODULE

RELAY Fig 2.2

A 1-channel relay module is a convenient and simple way to control high-voltage devices like Controlling lights,fans,and home appliances remotely with a low-voltage microcontroller like the ESP32. It acts as a switch that can isolate the control circuitry from the high-voltage load, providing a safe and reliable means to automate or remotely control various applications.

One of the main advantages of using a relay board is that it allows multiple circuits to be controlled using a single control signal. This can simplify the wiring and control logic for a system, reducing the overall complexity and cost. Another advantage of using a relay board is that it provides electrical isolation between the control signal and the circuits being controlled. This isolation helps to protect the control signal from electrical noise or interference, which can be particularly important in industrial or harsh environments. Relay consist of Com, NC

(normally closed), NO (normally open).

Normally Close Relays

A normally closed relay, also known as an NC relay, is a type of electromechanical relay that is normally closed when it is not energized. This means that the contacts of the relay are connected when the relay coil is not powered. When a control signal is applied to the coil of the NC relay, it creates a magnetic field that causes the contacts to open, breaking the circuit.

NC relays are commonly used in applications where it is important to ensure that a circuit remains closed when power is not being supplied. For example, they may be used in alarm systems to sound an alarm when a circuit is broken or in safety systems to activate an alarm or shut down a system when a specific signal is received. NC relays come in various sizes and designs, with different contact ratings, coil voltages, and switching characteristics.

Normally Open Relays

When used as a switch in electrical and electronic circuits, a normally open (NO) relay is a crucial component. An NO relay's contacts are open when it is in its default state, which stops current flow. A magnetic field is created within the relay by an electromagnet, which pulls the armature and closes the contacts when electricity is applied to the coil. By doing this, current can flow through and power the attached load, which could be a motor, light, or other device. NO relays are widely used in a variety of applications, from basic industrial machinery to complicated home automation systems. They provide a dependable way to operate high-power circuits using low-power signals.

Specifications:

- 1-channel relay module features a SPDT (Single Pole Double Throw)
- Load Voltage:

  AC: 250V @ 10A (maximum)

  DC: 30V @ 10A (maximum)
- Power Consumption: Around 70 mA at 5V when the relay is activated
- Pinout:

  VCC: Power supply pin (3.3V or 5V)

  GND: Ground

  IN: Control signal from the microcontroller

POWER SUPPLY

POWER SUPPLY Fig.2.3

To operate a bulb, the bulb holder and switch play critical roles as integral components. The bulb holder provides the physical interface to securely house and connect the bulb to the electrical circuitry. It ensures proper electrical contact and safety when interfacing with the relay module controlled by the ESP32 microcontroller. The switch, typically integrated into the bulb holder or as a separate component, acts as a manual override mechanism for toggling the bulb's power state locally.

To control a bulb, the proper provision of AC supply at 230V is pivotal. This voltage level is essential for powering the bulb and ensuring its reliable operation within the electrical system. Supplying AC (Alternating Current) power at 230V involves connecting the electrical equipment or appliances to a mains power source that provides a sinusoidal voltage waveform with an effective voltage of 230 volts.

JUMPING WIRES:

Jumper wires are a fundamental component in the world of electronics. They are a type of wire used to connect various components on a circuit board, breadboard, or other electronic projects. Jumper wires are usually made of a thin, flexible, insulated wire with connectors at both ends. One of the most common uses of jumper wires is in breadboards, which are used to prototype and test electronic circuits.

JUMPING WIRES Fig 2.4

3.2 SOFTWARE DESCRIPITION

3.2.1 FIREBASE CONSOLE

Firebase is a backend for mobile and web app development created by Google. It offers numerous services that simplify the process of creating, enhancing, and expanding your applications. You can create and release apps for iOS, Android, and the Web with the assistance of Firebase's comprehensive documentation and cross-platform SDKs.

ADVANTAGES OF FIREBASE:

Integration Ease: Because Firebase is created by Google, it is simple to link with other Google services, giving you the ability to include some strong features in your application.

Realtime Database: A NoSQL cloud database that enables real-time data syncing and storing across users.

Authentication: A system that supports phone auth, social login providers like Facebook, Twitter, and Google, and email and password authentication using solely client-side code.

cloud Storage: An object storage solution designed for Google scale that is strong, easy to use, and reasonably priced.

Cloud Functions: A serverless framework that enables you to launch backend programs on demand in response to HTTPS requests and Firebase capabilities.

Machine Learning: Use Firebase's ML Kit, which offers APIs for text recognition, image labeling, and other machine learning functions, to include machine learning capabilities into your projects.

Remote Configuration: Without requiring users to download an update, your app may be dynamically configured and customized for various user segments.

KODULAR

Kodular is a platform that allows you to create Android apps without having to write any code. It uses a visual block-based interface, where you drag and drop blocks to build the functionality of your app. This makes it a great option for beginners or those who don't have any coding experience.

Here are the few key things about kodular:

Components: Kodular provides a large selection of components that are useful for creating apps. These comprise connectivity (Bluetooth, Wi-Fi), media (camera, music player), sensors (accelerometer, GPS), buttons, labels, and graphics from the user interface, and many more.

Extensions: By utilizing extensions, users can improve the functionality of their programs. Extensions are extra parts made by developers or the community that offer new features and functionalities not found in the main platform.

Real-time Testing: Kodular allows for real-time testing of the apps using the Kodular Companion app. This app lets users see the changes they make to their project on their mobile device instantly, facilitating rapid prototyping and debugging. Kodular also has a robust community and support system, including forums, documentation, and tutorials that help users get started and troubleshoot issues.

ARDUINO

Arduino is an open-source electronics platform based on easy-to-use hardware and software. It is intended for anyone creating interactive projects, including designers, engineers, and students as well as enthusiasts. Microcontrollers known as Arduino boards are able to take inputs, such as light from a sensor, a user's finger on a button, or a message from Twitter, and convert them into outputs, such as starting a motor, turning on an LED, or posting content to the internet.

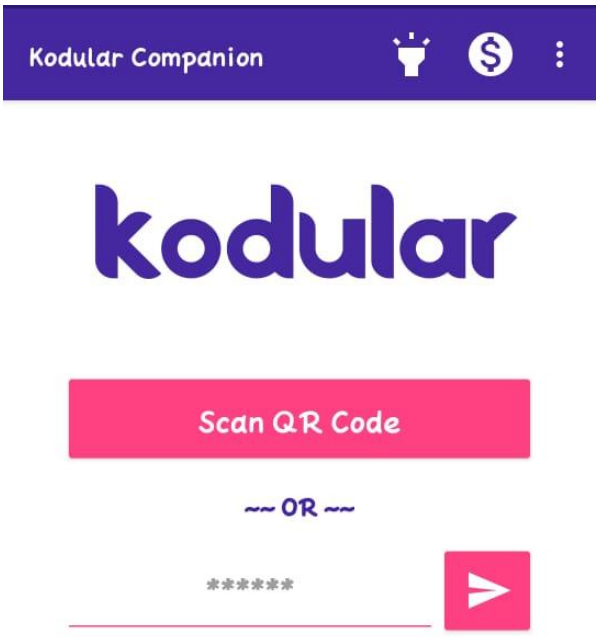
The code writing, compilation, and uploading process for the Arduino board is done using the Arduino IDE, or Integrated Development Environment. Along with pre-installed examples, it supports a condensed form of C++.

Arduino can communicate with the outside environment through the use of sensors and actuators. While actuators carry out activities (such as moving a motor, turning on a light, or producing music), sensors pick up changes in the environment (such as temperature, light, or motion).

KODULAR COMPANION MOBILE APP

Providing an easy-to-use interface for remote control and monitoring of household appliances. Users may design and develop unique mobile apps that easily interface with cloud services like Firebase with Kodular, a well-liked framework for developing Android applications without requiring deep programming experience. Here's a thorough analysis of the Kodular Companion app's function in this project.

This app can be integrated with Firebase Realtime Database, enabling real-time updates and control. When a user interacts with the app (e.g., pressing a button to turn on a lamp), the app updates the Firebase database, which the ESP32 at home monitors and responds to.The app can provide real-time status updates of home devices. For instance, it can display whether the lamp is currently on or off by reading data from Firebase. This feedback loop ensures users are always aware of the current state of their devices.



APP PREFACE Fig 2.5

CHAPTER 3
METHODOLOGY

Creating a smart home automation project using Kodular, Firebase, and an ESP32 microcontroller is a great way to integrate mobile app development with IoT. Below are the steps and necessary code to set up this project

3.1 CREATE A FIREBASE PROJECT

To create a firebase project here are the following steps:

- Step 1: Go to firebase console

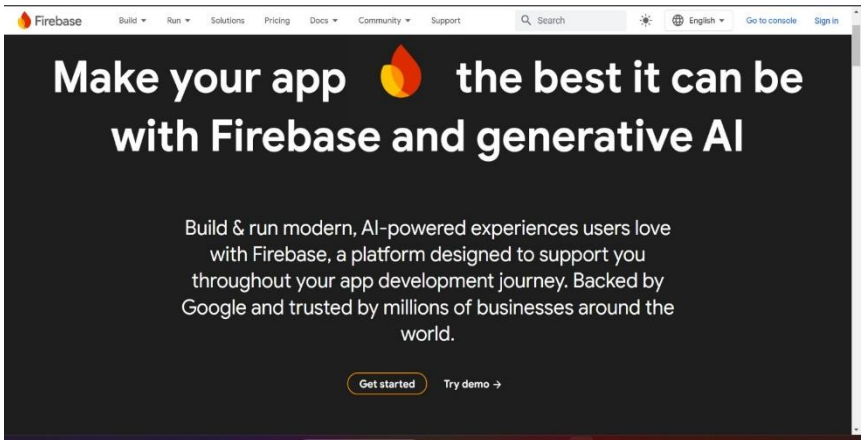


Fig 3.1

- Step 2: Sign into firebase using google account

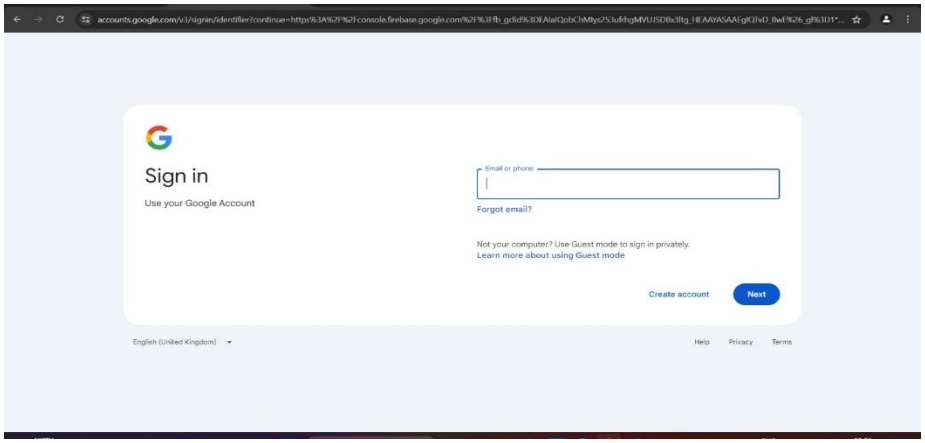


Fig 3.2

❖ Step 3: Create project

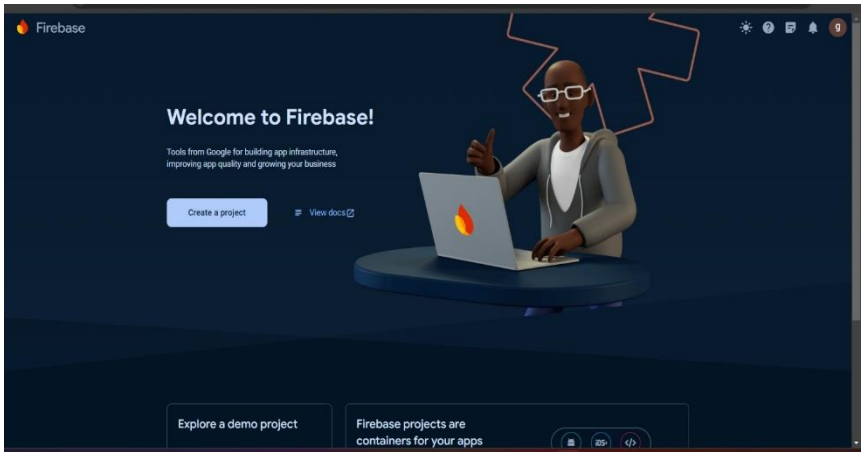


Fig 3.3

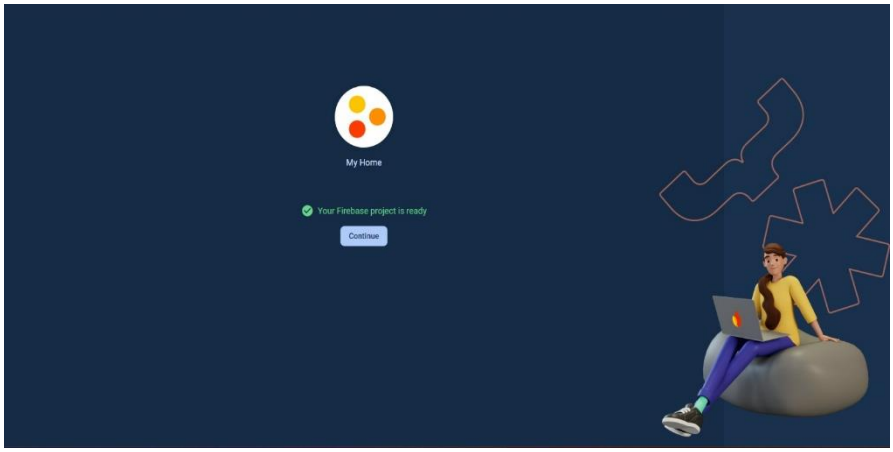


Fig 3.4

❖ Step 4: Create a Real time database

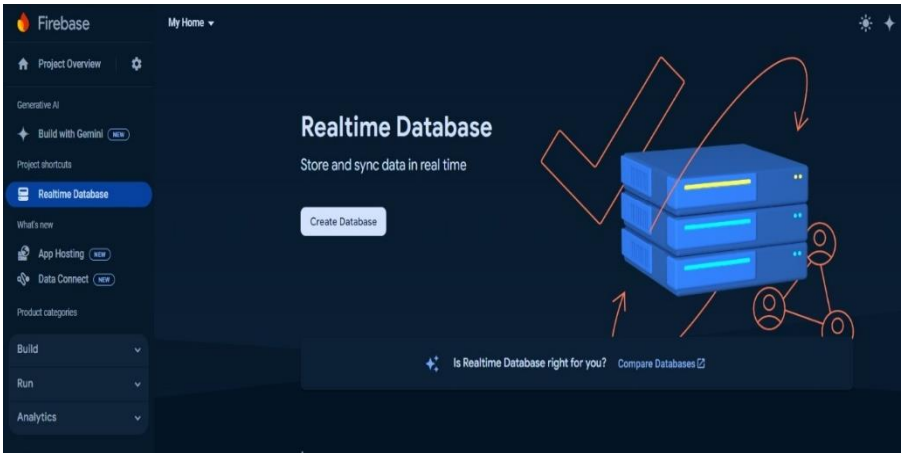


Fig 3.5

❖ Step 5: setup database in test mode

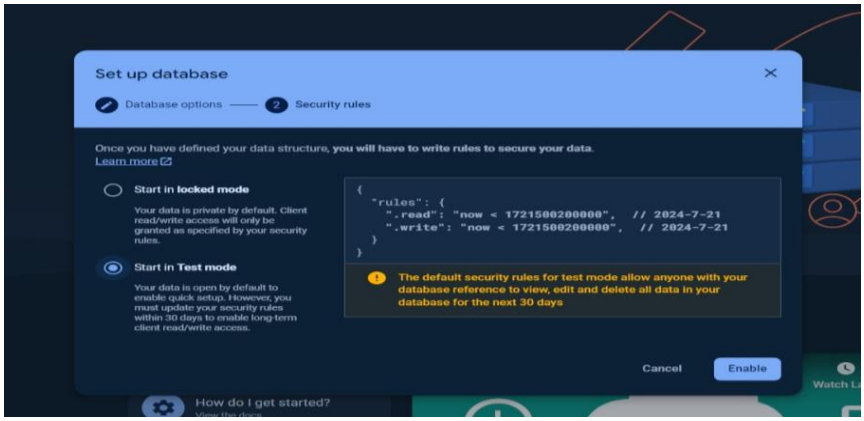


Fig 3.6

3.2 CREATING APP USING KODULAR

To get started with Kodular, you can visit their website and sign up for a free account. Once registered, you can start creating new projects, adding components, designing the user interface, and defining the app's logic using the block editor.

Here are the steps to create app:

❖ Step 1: open kodular.io and create app.

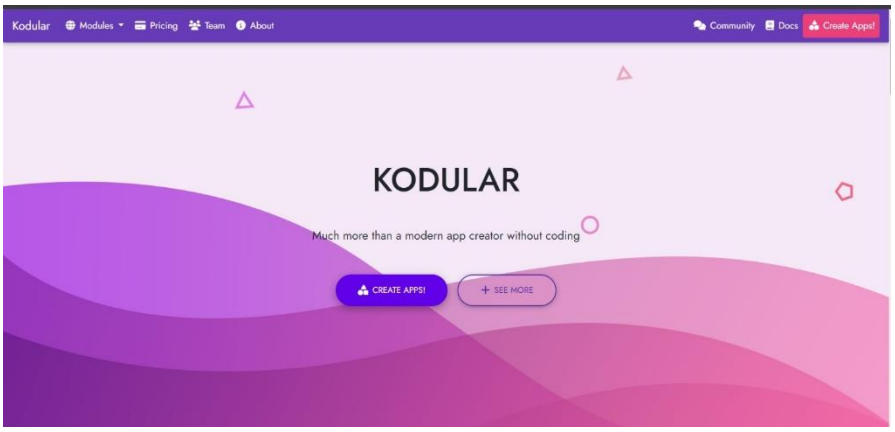


Fig 3.7

❖ Step 2: sign up or create kodular account

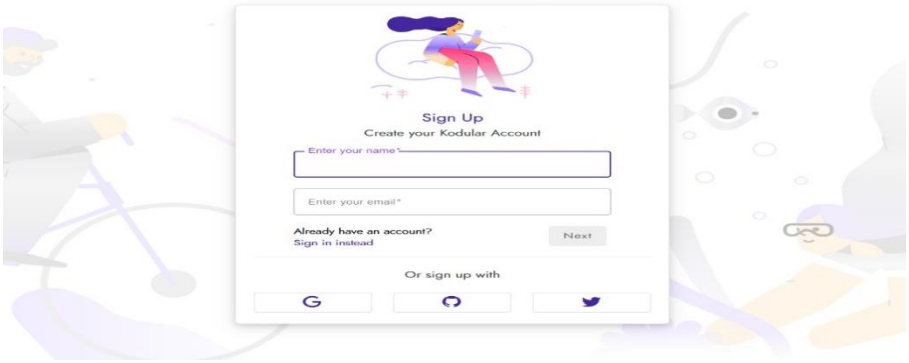
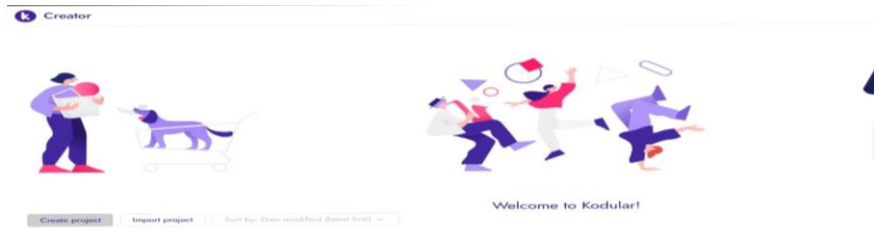


Fig 3.8

❖ Step 3: create new project and configure your project by adding app name



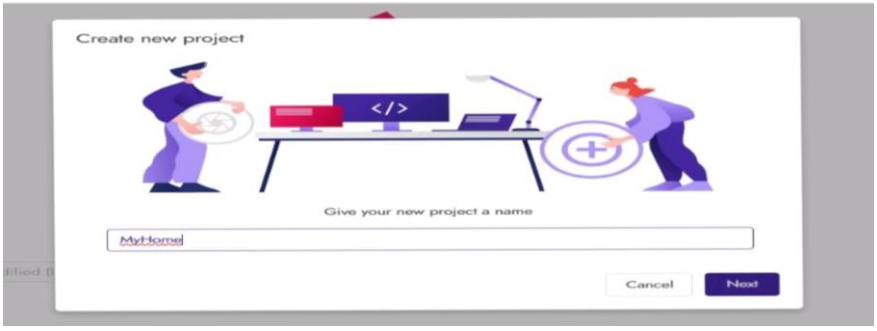


Fig 3.9

❖ Step 4: design the app interface using buttons, labels, and other components

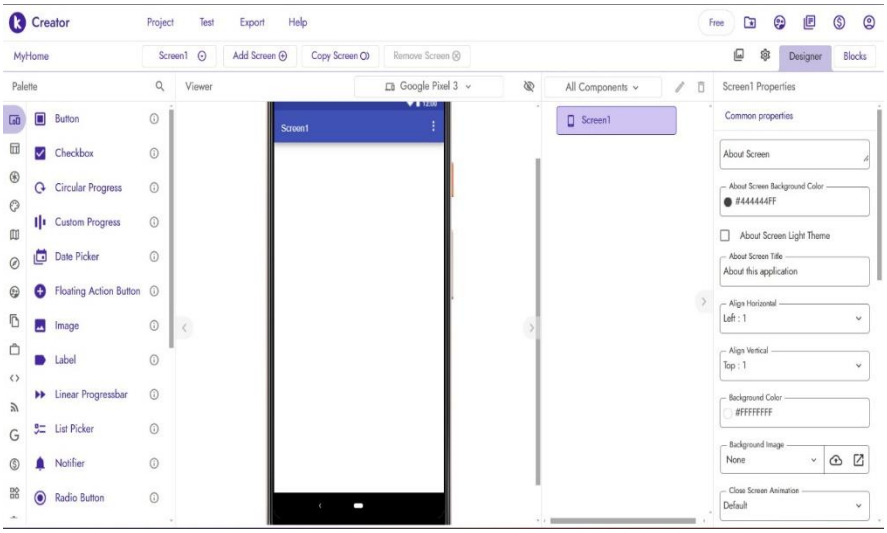


Fig 3.10

❖ Step 5: Add blocks to control ESP32 and send commands to firebase

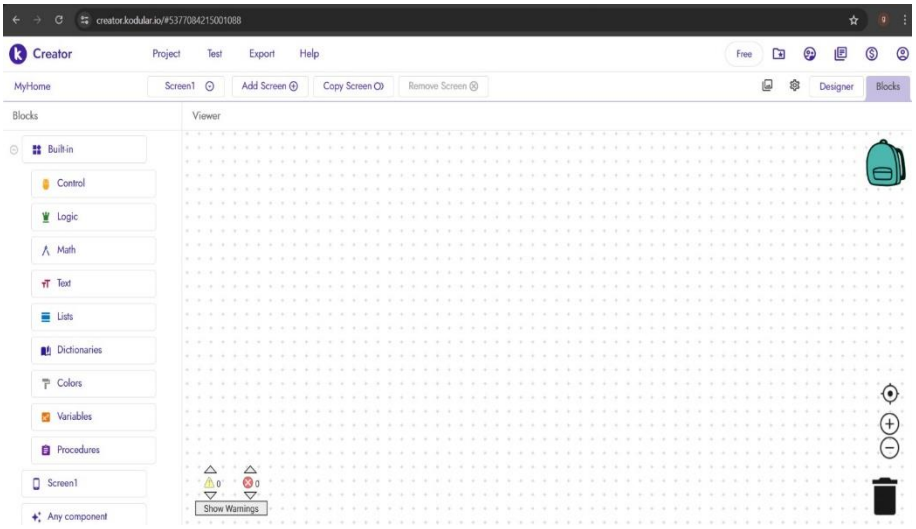


Fig 3.11

❖ Step 6: to test the app install kodular companion app through play store on your android device.

3.3 SET UP ARDUINO

The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code,a message area, a text console, a toolbar with buttons for common functions, and a series of menus. It connects to the Arduino hardware toupload programs and

communicate with them.

Step 1: Install Arduino ide

Step 2: Add ESP32 board to Arduino ide

Step 3: Install libraries from board manager

Required libraries:

- ❖ Wi-Fi Library
- ❖ Firebase ESP32 library

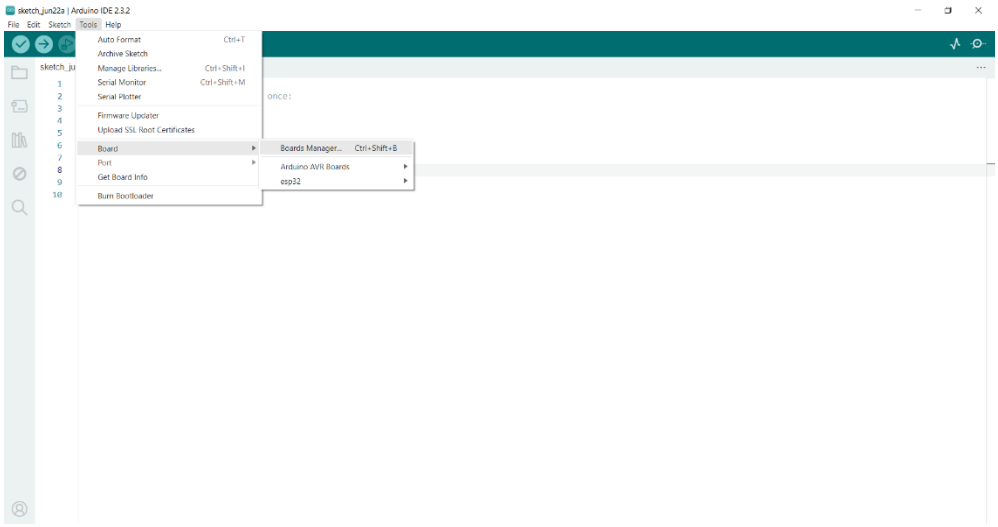
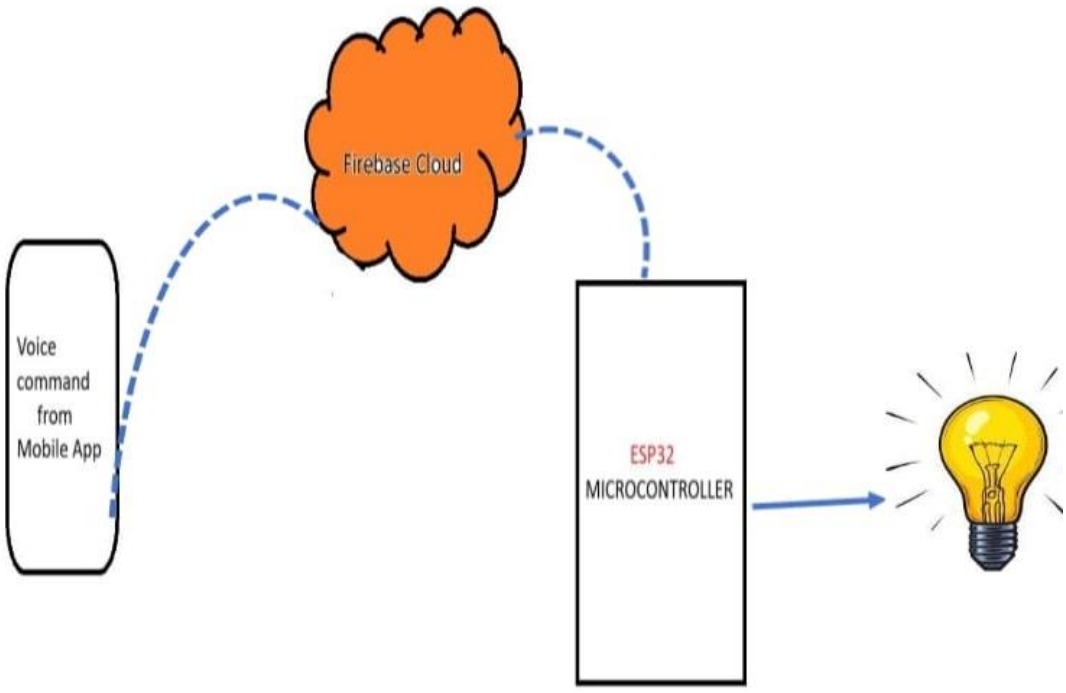


Fig 3.11

CHAPTER 4
WORKING AND IMPLEMENTAION

4.1 WORKING



SYSTEM FLOW Fig.4.1

The working of the smart home automation project involving ESP32, Firebase, and voice assistants revolves around a systematic interaction between hardware components and cloud-based services. Initially, the ESP32 microcontroller establishes a connection to a designated WiFi network, enabling it to communicate with Firebase Realtime Database through the FirebaseESP32 library. This connection serves as the backbone for retrieving and updating data stored in Firebase, specifically monitoring changes in the "IOTLAB/Alexa/Lamp_Cmd" node. Voice commands issued via Alexa or Google Assistant are processed through their respective platforms, which in turn update Firebase with commands to turn the lamp on or off. The ESP32 continuously checks Firebase for these commands; upon detection of a command (either '1' for ON or '0' for OFF), it triggers the corresponding action by controlling a relay connected to the lamp. This relay effectively switches the AC power supply to the lamp,

thereby illuminating or extinguishing it as per the user's voice command. This seamless interaction between cloud-based voice control, Firebase data management, and ESP32 hardware integration demonstrates a robust implementation of smart home technology, offering users convenient and intuitive control over their home lighting system from anywhere with internet connectivity.

ARDUINO CODE

```
#include <WiFi.h>
#include "FirebaseESP32.h"
#define FIREBASE_HOST "alexa1-f0e05-default-rtdb.firebaseio.com" //Do not include https:// in FIREBASE_HOST
#define FIREBASE_AUTH "VtbTRRnoNHpFc0HYWrE41S2YSpRPMGcVWbFsrTqN"
#define WIFI_SSID "wifi_name"
#define WIFI_PASSWORD "password"
//Define Firebase Data object
FirebaseData firebaseData;
#define ledPin 13
float value = 0.0;
void setup() {
  // put your setup code here, to run once:
pinMode(ledPin,OUTPUT);
Serial.begin(115200);
WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
Serial.print("Connecting to Wi-Fi");
  while (WiFi.status() != WL_CONNECTED)
  {
    Serial.print(".");
    delay(300);
  }
  Serial.println();
  Serial.print("Connected with IP: ");
```

```
  Serial.println(WiFi.localIP());
  Serial.println();
  Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
  Firebase.reconnectWiFi(true);
}
void loop() {
  // put your main code here, to run repeatedly:
String ledCmdFull;
String ledCmd;
int finalCmd;
Firebase.getString(firebaseData,"IOTLAB/Alexa/Lamp_Cmd",ledCmdFull);
ledCmd = ledCmdFull.substring(2,ledCmdFull.length()-2);
finalCmd = ledCmd.toInt();
Serial.println(finalCmd);
if(finalCmd == 1)
{
  digitalWrite(ledPin,HIGH);
  Serial.println("LED ON");
}
else
{
  digitalWrite(ledPin,LOW);
  Serial.println("LED OFF");
 }
}
```

4.2 IMPLEMENTATION

A seamless hardware-software interface must be developed in order to manage and monitor home equipment remotely when utilizing an ESP32 microcontroller with Firebase and Kodular for smart home automation. A dependable backend platform called Firebase is used to store and synchronize real-time data between the ESP32 and the Kodular app. An easy-to-use user interface for the smart home system can be designed using Kodular, a visual programming environment for Android application development. Firebase sends real-time updates to the Kodular app based on sensor data obtained from the ESP32. In order to perform things like turning on lights or changing thermostat settings, users can also send commands to Firebase through the app, which the ESP32 then retrieves. By guaranteeing real-time connectivity, this integration provides scalable and efficient smart home automation. Here are few steps in implementation.

INTEGRATING FIREBASE WITH KODULAR.

Step1: Open the project that you have created then open Realtime Database and copy the URL.

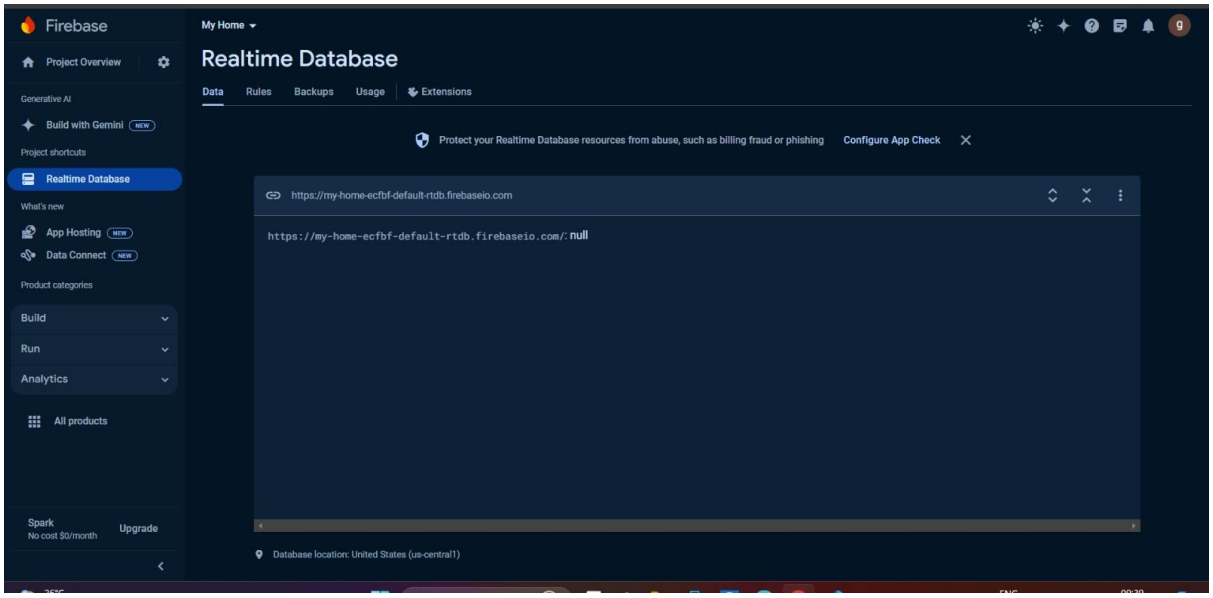


Fig.4.2

Step2: Open the Project settings from the project overview and copy the Database secrets from service accounts.

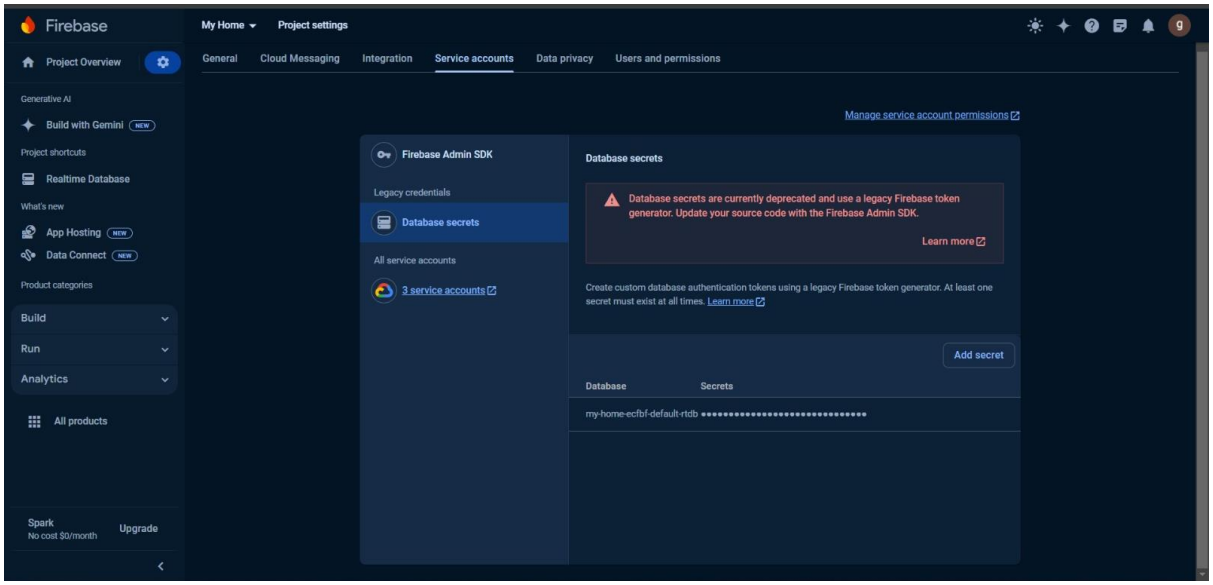


Fig. 4.3

Step3: Go to the projects in kodular creater and arrange the blocks as shown to design a user interface for the mobile app.

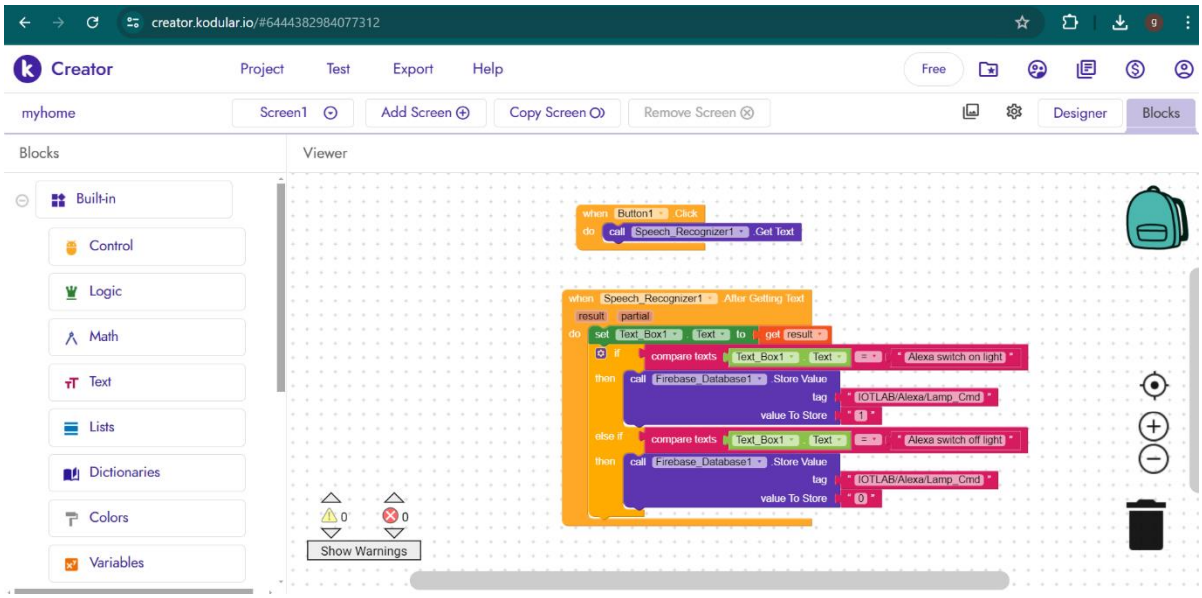


Fig.4.4

Step4: Add Firebase Authentications to the kodular project and make sure to empty the project bucket

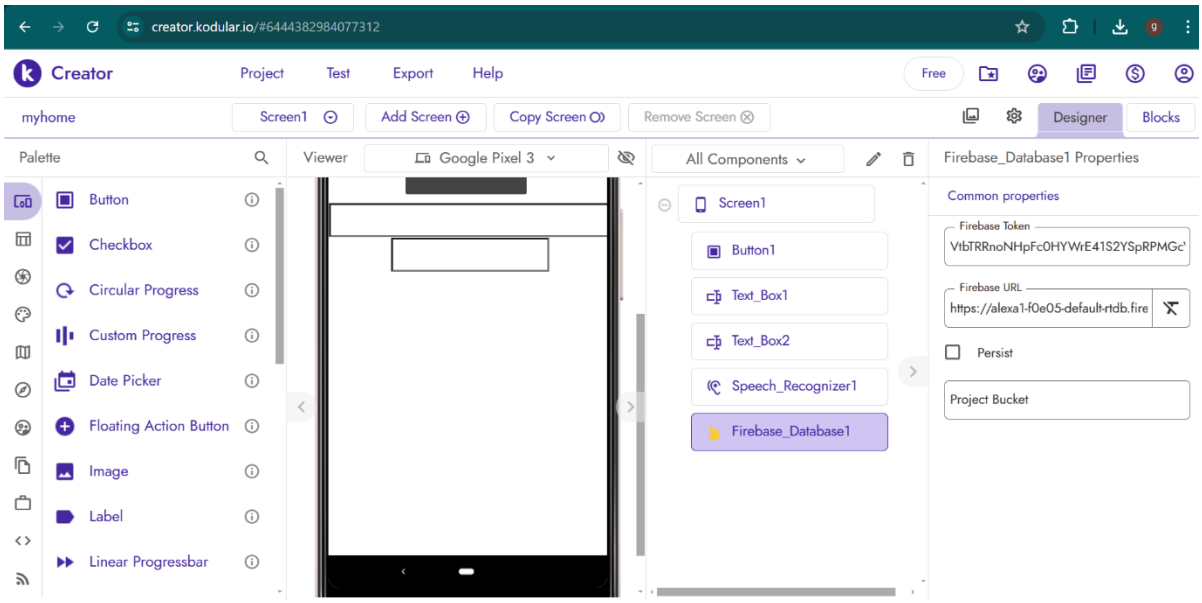


Fig 4.5

Step5: Open the Arduino and write the Arduino code to interface Firebase and ESP32.

Step6: Change the Firebase Credentials in the code. Before uploading the code make sure all the connections are connected properly.

Step7: To test the project open the Kodular Companion Mobile application then scan the QR code.

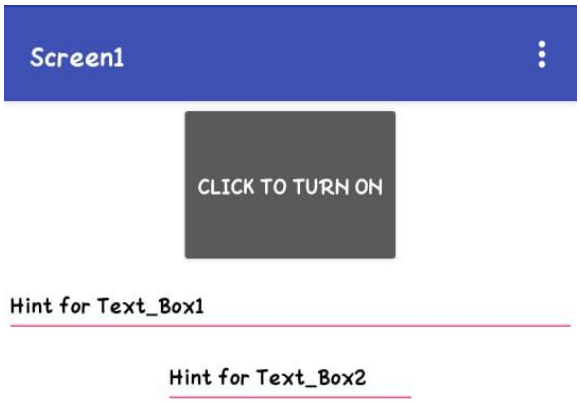


Fig 4.6

Step8: Kodular provides export options to generate the APK file, which can then be installed on Android devices or published to the Google Play Store.

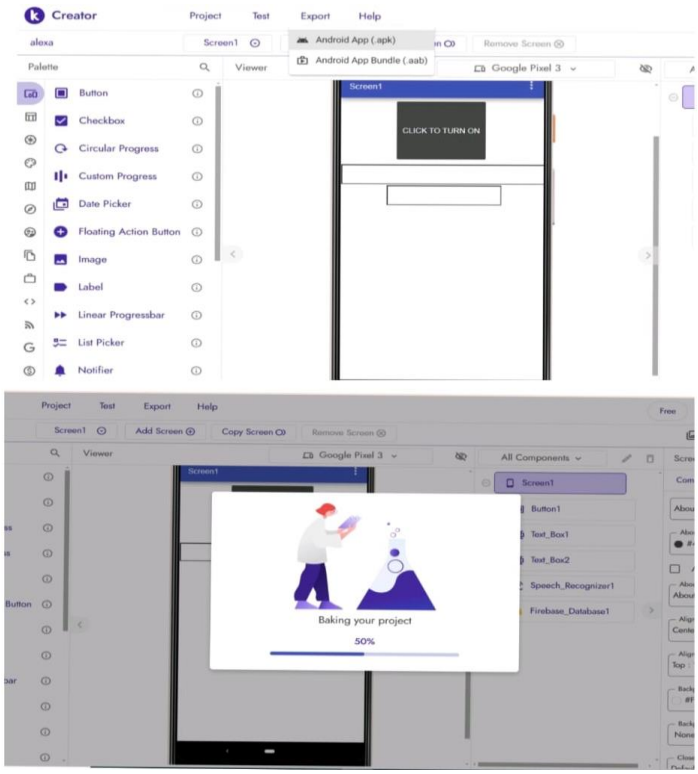


Fig 4.7

Step9: Give the voice commands and observe the switching action of the devices connected.

CHAPTER 5

APPLICATIONS

Convenience and Remote Accessibility

- **Control from Anywhere:** Users can turn lights on or off remotely using their mobile devices, providing flexibility and convenience.
- **Real-Time Updates:** The system provides instant feedback and updates, allowing users to see the current status of their lights in real-time.

Educational Value

- **Learning Opportunity:** This project provides a practical learning opportunity for understanding IoT, mobile app development, and cloud integration.
- **Skill Development:** Working on this project helps develop valuable skills in electronics, programming, and cloud computing.

Enhanced Security

- **Presence Simulation:** Users can turn lights on remotely to give the appearance of being home, deterring potential intruders.
- **Integration with Sensors:** The system can be integrated with motion detectors and other sensors to automate lighting based on presence, enhancing home security.

The cost-effectiveness

- **Low-Cost Components:** The system is affordable because to the relatively low cost of ESP32 microcontrollers and relay modules.
- **Open-Source Development:** Development expenses can be drastically lowered by making use of open-source tools like Firebase and Kodular.

An easy-to-use interface

- **Sensible Mobile App:** People of all ages can easily utilize the Kodular app thanks to its straightforward and user-friendly UI. Kodular makes app development accessible to anyone without programming skills by enabling drag-and-drop app creation.

Data Synchronization in Real Time

- **Seamless Integration:** The ESP32 and mobile app can synchronize data in real-time

thanks to Firebase, which guarantees a seamless and responsive user experience.

- **Reliability:** Firebase's cloud infrastructure guarantees high uptime and dependability, allowing the system to continue operating.

CHAPTER 6
CONCLUSION

To sum up, this smart home automation project successfully illustrates how to combine voice control technology, Internet of Things devices, and cloud-based services to improve the usability and convenience of home lighting systems. The project makes use of the ESP32 microcontroller's Wi-Fi connectivity and Firebase Realtime Database interface to allow real-time monitoring and management of a lamp using voice commands issued via Google Assistant and Alexa. An example of a useful application of IoT in daily life is the safe and dependable switching of the lamp's AC power source with the use of a single-channel relay module. This study demonstrates how cutting-edge technologies can be combined to provide smooth and approachable home automation solutions.

The study also emphasizes how crucial it is for devices and cloud services to communicate securely and effectively. In addition to making real-time command execution easier, the Firebase Realtime Database guarantees that the system may be expanded and modified to manage numerous devices in various places. This strong architecture serves as a versatile basis for more extensive smart home systems, allowing for future development and customization. This research lays the groundwork for more user-friendly and accessible home automation solutions by showcasing how voice assistants can be integrated with Internet of Things devices. This will ultimately contribute to the growing trend of smart homes and connected living spaces.

6.1 FUTURE SCOPE

This smart home automation project has a wide and bright future ahead of it, full of possibilities for improvements and more uses. Adding more smart devices to the mix and building a whole home automation ecosystem is one important direction. This could involve managing a variety of home appliances using a single platform, including air conditioners, security systems, and entertainment systems. Users can increase convenience and energy efficiency, as well as security and comfort, in their homes, by adding more devices to the system.

CHAPTER 7

LITERATURE SURVEY

Nikhil Rathod [1] et al, presented the architecture which is low cost and also they proposed flexible home

- Automation system which is using advanced versions of the Arduino microcontrollers. They concluded that using an Arduino is very easy to recognize with easy coding. They claim that implementing this kind of system we can make sure that the energy management can be completed It will augment the competence of this purpose. We manage the complete home domestic device over the internet. This will augment the reassure ability of humans and it will decrease the Human hard work.

International Journal for Multidisciplinary Research (IJFMR) [2] Niraj Patil1 , Anant Nimbalkar2 , Gaurav Pawar3 , Sulekha Shardul4

David Sheppard [3]; Nick Felker; John Schnalzel

- "Development of voice commands in digital signage for improved indoor navigation using Google Assistant SDK, IEEE sensors Applications symposium (SAS), IEEE

REFERENCES

- JSART - Volume 9 Issue 6 –JUNE 2023 ISSN [ONLINE]: 2395-1052
  Prof. Shubhangi Mahadik1, Tejashree Jain2, Mrunal Chavan
- International Research Journal of Modernization in Engineering Technology and Science Asst. Prof Mr. Ravi Jaiswal*1, Dhramvir Gond*2, Asst. Prof. Mr. Ankur Shukla*3, Satish Jaiswal*4
- Design and implementation of a Smart Home Automation System Ilesanmi Banjo Oluwafemi1,Oluwaseyi Olawale Bello2*,Tayo Dorcas Obasanya2

PROJECT PHOTOS

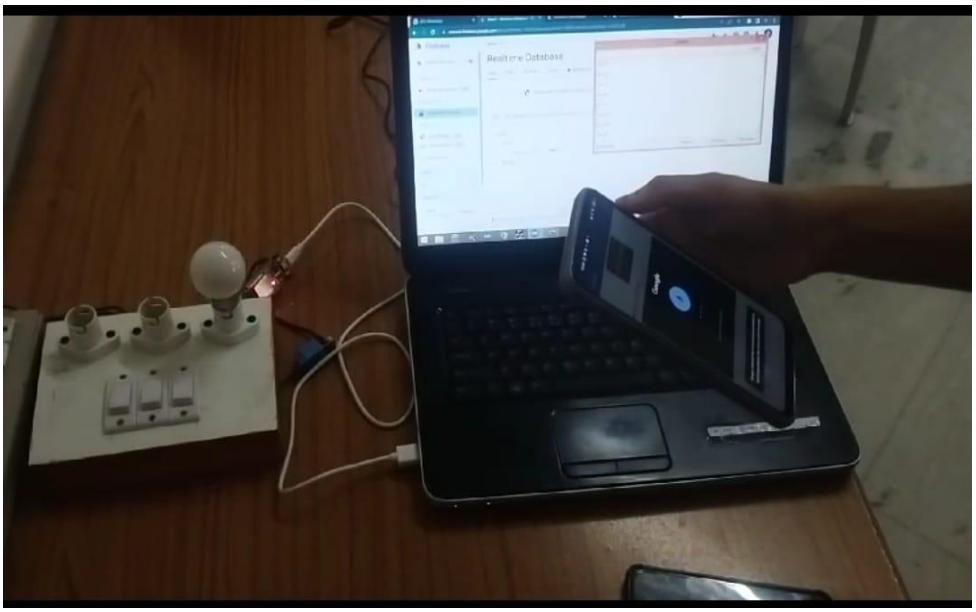


Fig (a) LIGHT OFF

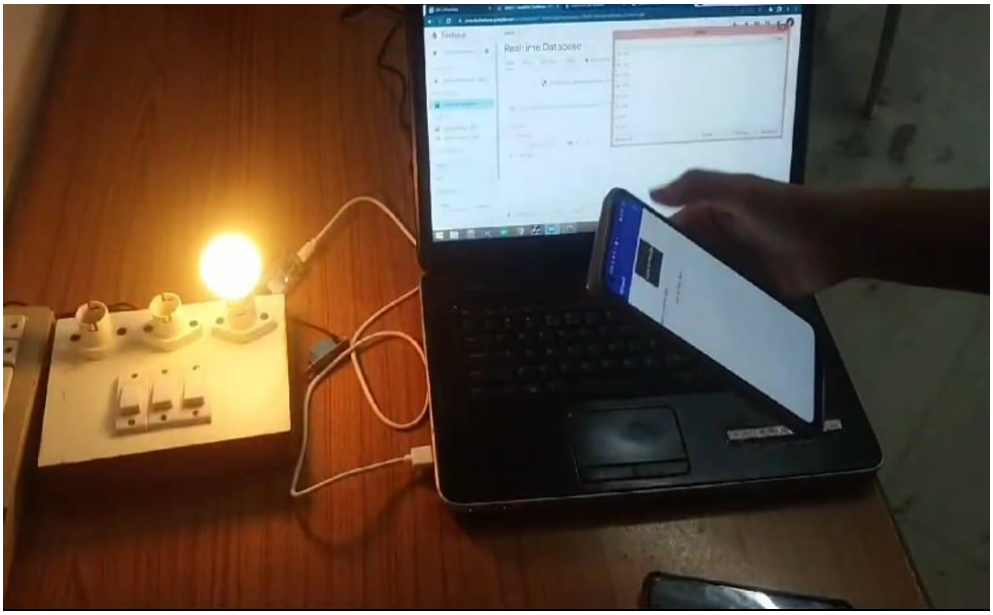


Fig (b) LIGHT ON

iThenticate

Similarity Report ID: oid:3618:63377555

PAPER NAME
Ms A. Lavanya1.docx

AUTHOR
A. Lavanya

WORD COUNT
4493 Words

CHARACTER COUNT
24681 Characters

PAGE COUNT
34 Pages

FILE SIZE
2.6MB

SUBMISSION DATE
Jul 25, 2024 3:13 PM GMT+5:30

REPORT DATE
Jul 25, 2024 3:13 PM GMT+5:30

14% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 13% Internet database
- 1% Publications database
- Crossref Posted Content database
- 11% Submitted Works database

Excluded from Similarity Report

- Crossref database
- Bibliographic material
- Small Matches (Less then 8 words)

Summary