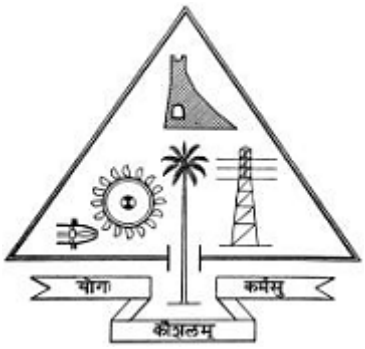
HOME AUTOMATION AND ENERGY MONITORING SYSTEM

Thesis submitted in partial fulfillment of the requirements for the award of the degree of ***Master of Computer Applications*** *of the* ***APJ Abdul Kalam Technological University***

submitted by

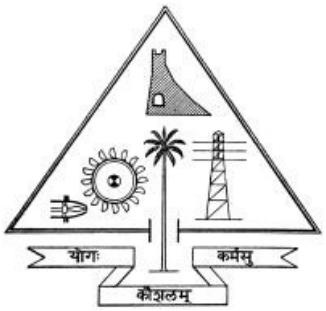
ANUSREE JNANAKRISHNAN
(TCR20MCA-2011)



DEPARTMENT OF COMPUTER APPLICATIONS
GOVERNMENT ENGINEERING COLLEGE
THRISSUR - 680009

JULY 2022

DEPARTMENT OF COMPUTER APPLICATIONS
GOVERNMENT ENGINEERING COLLEGE, THRISSUR
THRISSUR, KERALA STATE, PIN 680009



CERTIFICATE

This is to certify that the main project titled **"HOME AUTOMATION AND ENERGY MONITORING SYSTEM"** *is a bonafide work done by* **ANUSREE JNANAKRISHNAN (TCR20MCA-2011)** *under my supervision and guidance, and is submitted in May 2022 in partial fulfillment of the requirements for the award of the Degree of Master of Computer Applications from APJ Abdul Kalam Technological University(KTU).*

Asst.Prof Hussain Ahmed
Project Guide

Dr. Sminesh C N
Project Coordinator and HOD

Place : THRISSUR
Date : 16-07-2022

External Examiner

DECLARATION

I hereby declare that the main project named, **HOME AUTOMATION AND ENERGY MONITORING SYSTEM**, is my own work and that, to the best of my knowledge and belief, it contains no material previously published by another person nor material which has been accepted for the award of any other degree or course of the university or any other institute of higher learning, except where due acknowledgement and reference has been made in the text.

Place : THRISSUR Signature

Date : 16-07-2022 **ANUSREE JNANAKRISHNAN (TCR20MCA-2011)**

ACKNOWLEDGEMENT

I would like to thank Department of Computer Application, GEC Thrissur, for giving me this opportunity to pursue this project and successfully complete it.

I am highly indebted to head of the department **Dr.Sminesh C N** and our guide **Asst.Prof Hussain Ahmed** for the guidance and constant supervision as well as for providing necessary information regarding the project and also for his support in completing the project.

I express my heart-felt gratitude to project coordinator , **Dr.Sminesh C N**, for his committed guidance, valuable suggestions, constructive criticisms and precious time that she invested throughout the work.

I sincerely thank all other faculties of MCA department for guiding through the processes involved in the project.

ABSTRACT

The main objective of this project is to develop a home automation system which is also able to monitor and control overall energy usage of appliances in a house. As technology is advancing so houses are also getting smarter. Modern houses are gradually shifting from conventional switches to centralized control system, involving remote controlled switches. Presently, conventional wall switches located in different parts of the house makes it difficult for the user to go near them to operate. It becomes more difficult for the elderly or physically handicapped people to do operate. Remote controlled home automation system provides a most modern solution with smart phones. In order to achieve this, a WiFi module is interfaced to the Arduino board at the receiver end while on the transmitter end, a GUI application on the mobile phone sends ON/OFF commands to the receiver where loads are connected. The loads can be turned ON/OFF remotely through the specified buttons in flutter application. The user is able to monitor all the connected devices and it's power consumption through the mobile application. The electricity bill can be minimized by monitoring and controlling home appliances. The main attraction of the system is that user will be able to control the appliances from anywhere in the world.

CONTENTS

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

Nowadays, we have remote controls for our television sets and other electronic systems, which have made our lives real easy. Have you ever wondered about home automation which would give the facility of controlling tube lights, fans and other electrical appliances at home using a remote control? Off-course, Yes! But, are the available options cost-effective? If the answer is No, we have found a solution to it. The proposed system called Arduino based home automation using WiFi. This system is super-cost effective and can give the user, the ability to control any electronic device without even spending for a remote control. This project helps the user to control all the electronic devices using his/her smartphone.

Time is a very valuable thing. Everybody wants to save time as much as they can. New technologies are being introduced to save our time. To save people's time we are introducing Home Automation and Energy Monitoring System. With the help of this system you can control your home appliances from your mobile phone as well as from Firebase. You can turn on/off your home appliances from anywhere in the world. Also user is able to monitor the power consumption of appliances.

Smart home technology generally refers to any suite of devices, appliances, or systems that connect into a common network that can be independently and remotely controlled. The Home Automation and Energy Monitoring System which have so many benefits. The user can manage all of the home devices from one place. Being able to keep all of the technology in your home connected through one interface is a massive step forward for technology and home management.

Smart home systems tend to be wonderfully flexible when it comes to the accommodation of new devices and appliances and other technology. No matter how state-of-the-art your your appliances seem today, there will be newer, more impressive models developed as time goes on. Don't underestimate the power of being able to control your home's functions from a distance. On an exceptionally hot day, you can order your house to become cooler in just enough time before you get home from work. If you're in a hurry to get dinner started but you're still at the store, you can have your oven start to preheat while you're still on your way home. You can even check to see if you left the lights on, who is at your front door, or make sure you turned off all your media while you're away.

Another benefit is Increased energy efficiency. Depending on how you use your smart-home technology, it's possible to make your space more energy-efficient. Smart homes can also help you run your appliances better. A smart TV will help you find better apps and channels to locate your favorite programming. A smart oven will assist you with cooking your chicken to perfection – without ever worrying about overcooking or under-cooking it. An intelligently designed home theater and audio system can make managing your movie and music collection effortless when entertaining guests. Ultimately, connecting your appliances and other systems with automation technology will improve your appliance effectiveness and overall make your home life much more easier and enjoyable! .

Nowadays, people's dependency on electricity is extreme, as power consumption has increased for the past few years. It is imperative to consider monitoring and measuring the electric system or appliances that operate every day for residential and commercial buildings. Electricity bill is one of the major operating expenses in most of the commercial buildings and industrial plants. Thus, the building's energy management system is an essential element that should be utilized to optimize the energy usage.

CHAPTER 2
ENVIRONMENTAL STUDY

2.1 System Configuration

System configuration describe the hardware and software requirement of the system for development

2.1.1 Hardware Requirements

- Memory : 4 GB of RAM
- Processor : Intel Core i3 or equivalent CPU
- Speed : 2.4 GHz
- Hard Disk Space : 320 GB

2.1.2 Software Requirements

- Operating system : Windows 8 or above
- Front End : Flutter
- Back End : Dart, Firebase Realtime Database
- Framework : Flutter
- IDE Used : Arduino IDE, Android Studio

2.2 Software Specification

2.2.1 Flutter

Flutter is Google's portable UI toolkit for crafting beautiful, natively compiled applications for mobile, web, and desktop from a single codebase. Flutter works with existing code, is used by developers and organizations around the world, and is free and open source. For developers, Flutter lowers the bar to entry for building apps. It speeds app development and reduces the cost and complexity of app production across platforms. For designers, Flutter provides a canvas for high-end user experiences. Flutter is different than most other options for building mobile apps because it doesn't rely on web browser technology nor the set of widgets that ship with each device. Instead, Flutter uses its own high-performance rendering engine to draw widgets. In addition, Flutter is different because it only has a thin layer of C/C++ code. Flutter implements most of its system (compositing, gestures, animation, framework, widgets, etc) in Dart (a modern, concise, object-oriented language) that developers can easily approach read, change, replace, or remove.

Features :

- Heavily optimized, mobile-first 2D rendering engine with excellent support for text.
- Modern react-style framework.
- Rich set of widgets implementing Material Design and iOS-style.
- APIs for unit and integration tests.
- Interop and plugin APIs to connect to the system and 3rd-party SDKs.
- Headless test runner for running tests on Windows, Linux, and Mac.
- Dart DevTools for testing, debugging, and profiling your app.
- Command-line tools for creating, building, testing, and compiling your apps.

2.2.2 Dart

Dart is an open-source, general-purpose, object-oriented programming language with C-style syntax developed by Google in 2011. The purpose of Dart programming is to create a frontend user interfaces for the web and mobile apps. It is under active development, compiled to native machine code for building mobile apps, inspired by other programming languages such as Java, JavaScript, C, and is Strongly Typed. Since Dart is a compiled language so you cannot execute your code directly; instead, the compiler parses it and transfer it into machine code.

It supports most of the common concepts of programming languages like classes, interfaces, functions. Dart language does not support arrays directly. It supports collection, which is used to replicate the data structure such as arrays, generics, and optional typing. Dart is an open-source programming language, which means it is freely available. It is developed by Google, approved by the ECMA standard, and comes with a BSD license. Dart supports all primary operating systems such as Windows, Linux, Macintosh, etc. The Dart has its own Virtual Machine which known as Dart VM, that allows us to run the Dart code in every operating system. It also supports advance concepts like mixin, abstract, classes, reified generic, and robust type system. Dart consists of many useful inbuilt libraries including SDK (Software Development Kit), core, math, async, math, convert, html, IO, etc. It also provides the facility to organize the Dart code into libraries with proper namespacing. It can reuse by the import statement. Dart provides the flexibility to compile the code and fast as well. It supports two types of compilation processes, AOT (Ahead of Time) and JIT (Just-in-Time). The Dart code is transmitted in the other language that can run in the modern web-brewers.

2.2.3 Firebase Realtime Database

The Firebase Realtime Database is a cloud-hosted database. Data is stored as JSON and synchronized in realtime to every connected client. When

you build cross-platform apps with our Apple platforms, Android, and JavaScript SDKs, all of your clients share one Realtime Database instance and automatically receive updates with the newest data. Firebase apps remain responsive even when offline because the Firebase Realtime Database SDK persists your data to disk. Once connectivity is reestablished, the client device receives any changes it missed, synchronizing it with the current server state.

The Firebase Realtime Database can be accessed directly from a mobile device or web browser; there's no need for an application server. Security and data validation are available through the Firebase Realtime Database Security Rules, expression-based rules that are executed when data is read or written. The Firebase Realtime Database lets you build rich, collaborative applications by allowing secure access to the database directly from client-side code. Data is persisted locally, and even while offline, realtime events continue to fire, giving the end user a responsive experience. When the device regains connection, the Realtime Database synchronizes the local data changes with the remote updates that occurred while the client was offline, merging any conflicts automatically.

The Realtime Database provides a flexible, expression-based rules language, called Firebase Realtime Database Security Rules, to define how your data should be structured and when data can be read from or written to. When integrated with Firebase Authentication, developers can define who has access to what data, and how they can access it.

The Realtime Database is a NoSQL database and as such has different optimizations and functionality compared to a relational database. The Realtime Database API is designed to only allow operations that can be executed quickly. This enables you to build a great realtime experience that can serve millions of users without compromising on responsiveness

2.2.4 Arduino IDE

The Arduino IDE is an open-source software, which is used to write and upload code to the Arduino boards. The IDE application is suitable for different operating systems such as Windows, Mac OS X, and Linux. It supports the programming languages C and C++. Here, IDE stands for Integrated Development Environment.

The program or code written in the Arduino IDE is often called as sketching. We need to connect the Genuino and Arduino board with the IDE to upload the sketch written in the Arduino IDE software. The sketch is saved with the extension '.ino.' Arduino IDE is an open-source tool that makes it possible for users to write as well as upload code to a work environment in real-time. Since the written code will be moved to the cloud, it's frequently used by those who need an additional level of redundancy. Arduino IDE offers full compatibility to any Arduino-based software board. The software can easily be deployed in any Linux, Mac, or Windows operating systems. Most of its parts are written within JavaScript for seamless compilation and editing. While the tool's main aim is based on code writing, it offers several noteworthy functionalities.

For instance, Arduino IDE lets users share important project information to company stakeholders. Users are given the freedom to make internal layouts and schematic modifications when needed. Comprehensive guides are available for those who need help in the installation process. Tutorials are present for users who have little experience dealing with the tool's framework. Arduino IDE is highly rated by users for its ease of use. It can conduct complex processes while keeping computer resources to a minimum. The tool makes it easy for users to access their libraries. At the same time, it offers updated support for the latest Arduino boards, which can help users with their sketches using the latest IDE version.

2.2.5 Android Studio

Android Studio is the official Integrated Development Environment (IDE) for android application development. Android Studio provides more features that enhance our productivity while building Android apps.

- It has a flexible Gradle-based build system.
- It has a fast and feature-rich emulator for app testing.
- Android Studio has a consolidated environment where we can develop for all Android devices.
- Apply changes to the resource code of our running app without restarting the app.
- Android Studio provides extensive testing tools and frameworks.
- It supports C++ and NDK.
- It provides build-in supports for Google Cloud Platform. It makes it easy to integrate Google Cloud Messaging and App Engine.

2.3 Functional Requirements

2.3.1 Controlling Devices

For controlling the devices, a Relay is required. In order to control four devices a four channel relay is required. The user can control the devices through Firebase real-time database as well as using Flutter application. By connecting Firebase and flutter application user can control the devices through application. To connect the devices to internet, NodeMCU which has an inbuilt wifi module is used.

2.3.2 Monitoring Power Consumption

The Current and Voltage consumption details can be acquired through ACS712 current sensor and ZMPT101B voltage sensor respectively. These

sensors are connected to Arduino UNO R3. Sensor data stored in Arduino is transferred to NodeMCU. NodeMCU is connected to Firebase. Data can be visualized in firebase. By connecting flutter application and firebase the data can be visualize in application.

2.3.3 Electricity Bill Generation

Using the voltage and current sensor data, power consumption of each devices can be identified. Time duration of each devices stored. Using these details , unit of energy each device consumed is detected, thereby electricity bill is generated.

2.4 Performance Requirements

Performance requirements define how well the software system completes the given task under some specific conditions. The system would need least 4 GB of RAM. Less RAM will result in the poor performance of the system. Proper internet connection is necessary. The response time depends upon the internet connection. If there is a high speed network available the Request-Response time will be minimized. The system is scalable that is if we increase in the system's workload that the system should be able to process. There are 3 devices connected. User can add more devices. The application is able to control multiple devices at a time.

CHAPTER 3
LITERATURE REVIEW

Study on Home automation and energy monitoring has been a very active research field for a long time. Technologies are developing day by day. New features in home automation field are also taking place.

Smart Home Automation Using Intelligent Electricity Dispatch [1] , date of publication August 20, 2021, date of current version August 31, 2021 , This work was supported in part by the University of Engineering and Technology Taxila and the Higher Education Commission (HEC) under NRPU Project 6338, and in part by Taif University Researchers Supporting Project (TURSP) under Grant TURSP-2020/73, Taif University, Ta'if, Saudi Arabia. According to this paper, smart phones were the solution to efficient and secure remote control access.This research investigates the smart home automation using an intelligent electricity dispatch model. This study focuses intelligent automation in three different ways and making the automated household act smartly at the time of voltage distortion. The proposed methodology is beneficial for electricity saving as it overcomes electricity consumption, which is the main target to achieve as unwanted appliances will be automatically switched off according to given conditions. In addition, proposed technique provides convenient solutions to the user to switch devices from remotely.

Another work done in the same field is "Live Power Consumption Monitoring and Home Automation using Google Assistant" [2] published by Saran Kumar.R. , Srevignesh, Vijay, Sivaraman and T.Tamilarasan in 2020, International Advanced Research Journal in Science, Engineering and Technology. This paper suggests an idea to reduce power consumption in home

by continuously monitoring the power consumed and allow the user to view the estimated cost for the usage. SMS and E-mail notifications are sent to the user as Alert messages sot that the user may consume the power more carefully. The Automation and the Intruder alert system are done using the PIR sensor which helps in providing an easy, effective and a secure system for the house. Controlling of Home appliances using Google assistant also helps everyone in the ease by just giving voice commands rather than getting up from the seat to turn ON/OFF the appliances. IBM and AWS IoT services are used in this paper for providing all the information the user needs and also for the Notifications Alert system.The IBM IoT platform is used for displaying the total power consumed and the estimated cost to be payed. The (Amazon Web Service) AWS is used for the SMS and E-mail Alert system.

In 2020 ”IOT BASED SMART HOME ENERGY MANAGEMENT SYSTEM” [3] Research paper was proposed by Jishnu Jayakumar, Amal T and Asst. Professor Shinu James in International Research Journal of Engineering and Technology (IRJET). The major processing unit of the entire system is the Arduino unit, which is been connected to the various modules. The primary connection measures the current consumed by various devices. Further a connection is unified from this system to the relay and then further to other components like temperature sensor (LM 35), LDR, buzzer, LCD display (16*2). The calculated power and energy by the Arduino are displayed on a 16×2 LCD display module. The inbuilt Wi-Fi chip of the Arduino UNO is connected to the Home Router and linked to the Blynk App. So, you can monitor the parameters as well as calibrate and modify different settings from your Smartphone via OTA.

The paper ”Wireless Energy Meter with Home Automation” [4] proposed by Vigneshwaran , Rakesh , Roshith , Vimalraj , Mrs. Lakshmi in International Research Journal of Engineering and Technology (IRJET) 2020, presents a study on IoT based smart energy meter solves the issues of prepaid energy metering by minimizing the complexities and mitigates the

non-technical losses by ensuring the credibility of data. It also brings new important features, such as real-time viewing of consumption data and remote controlling of home appliances. A single phase static watt-hour meter is used to calculate the consumed energy and these data are extracted from the meter through a LED.

"Automatic Electricity Bill Generating System" [5] by N. Rajathi, N. Suganthi and Shilpa R , International Journal of Recent Technology and Engineering (IJRTE) in 2018. The proposed system measures the energy consumed at each house automatically and the readings are being displayed to them at their EB box. The system proposed measures the energy consumed at each house automatically and the readings are being displayed in the LED present at their EB box. This is very useful because it helps the commons to be aware of their usage and can reduce if they are over using the resource available to them. The readings and bill are transferred to the EB database via Wi-Fi module. Thus reducing human labor.

"IoT based Home Automation System for Electricity Usage" [6] proposed by Swapnil Talkar and Ruhi Bajaj in International Journal of Advance Research, Ideas and Innovations in Technology in 2017. In the proposed system the Arduino micro controller is used to read the electric consumption data from energy meter when the electric appliances (load) are connected to energy meter. Whenever a specific amount of energy is utilized by the load, the calibration led of energy meter will blink.

All the researches in the field of home automation are effective to control and monitoring the devices inside home. But most of them are not cost efficient. The component cost and other requirements will become a barrier for many of the people to afford the system. There is a need to develop a system which is cost efficient and effective.

CHAPTER 4
SYSTEM ANALYSIS

System Analysis by definition is a procedure of deliberate examination to accumulate information, deciphering the realities, diagnosing the issue and utilizing this data to either manufacture a totally new framework or to prescribe the enhancements to the current framework.

A good system investigation includes the way toward looking at a business circumstance with the purpose of improving it through better strategies and systems. In its center sense, the investigation stage characterizes the necessities of the framework and the issues which client is attempting to explain independent of how the prerequisites would be practiced.

4.1 Existing System

A common practice in India is that a person from utility services goes to individual consumers to collect the energy meter data. The bill is orderly prepared using those data and the bill paper, which contains all the information about the usage of electricity in that month, is sent to individual consumers for payment. Finally, consumers pay their bill according to the bill paper, which was sent to them by going to a particular station or a bank. Therefore, this process is very much hassling and requires a lot of time and steps, and since there is direct involvement of human in collecting the meter data, the credibility of this system is compromised. Normally, the power consumption is measured by using conventional energy meters and digital energy meters are used.

There are system losses occurring in the power network due to technical and non-technical aspects. Technical reasons include losses in power

generation equipment, transmission lines, distribution lines, etc.; whereas nontechnical losses occur mainly due to customer level meter tampering. In order to reduce the losses occurring at the customer level, the prepaid meter has been introduced in some cities and the rest are on the way. Although the prepaid metering system offers more advantages than the conventional meter system, it also has some major drawbacks. The customers need to purchase a card to refill the account but purchasing the card is quite troublesome sometimes. The consumers also need to predict the monthly bill and recharge accordingly, which can create a situation of uncertainty. Furthermore, prepaid energy meters do not provide the consumer with the ability to control electrical appliances remotely

4.2 Limitations of existing system

- Most of the people does not aware of Home Automation concept.
- No cost efficient system is available.
- User does not know about the power consumption of each devices.
- Can't identify which device consume more power.

4.3 Proposed System

The evolution of technology has increased the consumption of electric power locally and globally which lead to a dramatic increase in demand for electric power. Electricity consumption rate in different forms at home and commercially increased. Sometimes, it affects household appliances due to the raised demands based on conditions of load shedding, electricity shortfall, and emergencies. There is a need to develop a system that uses various appliances effectively and efficiently while keeping the cost of electricity minimum. This system is the combination of software and hardware based technologies. Hardware is comprised of microcontroller which is connected to sensors , Wi-Fi module and relays , appliances can be controlled under given conditions. Software based part is comprised of an application which utilizes the energy usage data of appliances. From every corner of the world, household appliances can be controlled in the proposed smart efficient energy house automation. With smart energy monitoring system, track power consumption for single or multiple devices. Identify abnormal patterns of energy use , Estimate your energy bill, allowing you to see the overall impact of energy use on your wallet. By analyzing the energy consumption user can reduce the energy usage in which the application allowed to set a limit for energy consumption. The system is beneficial for electricity saving as it overcomes electricity consumption, Also provides ease and comfort to the user. It is cost-efficient in terms of electricity usage of hardware/software. The system is effective and reliable.

4.4 Advantages of proposed System

- Efficient.
- Can monitor the abnormal usage of energy by appliances.
- Can monitor and control appliances.

- track live power consumption. in real-time for single or multiple devices.
- Estimation of your energy bill.

4.5 Feasibility

Feasibility refers to whether or not a project will be successful and how to overcome potential obstacles for the project.A feasibility study presents the pros and cons of a project.Feasibility studies answer questions about the availability of resources, the tools teams need to complete the project.

4.5.1 Scope Management

The project aims to solve the energy problem , the government or country is facing the upcoming years. By monitoring and controlling appliances, user can reduce the energy consumption of devices there by overall energy management is possible. Project goals and tasks: Reading the sensor data, store the data in Arduino, send data to NodeMCU, from NodeMCU send data to firebase, firebase configuration, Flutter application which is able to send and receive data, Store data temporarily and generate electricity bill , Erase the data when electricity bill is downloaded. Overall budget is below 3000 rupees. The project duration is 2 month.

4.5.2 Components Feasibility

Most of the required components are available in the market. All the IoT related components are easy to use and have many purposes other than home automation and energy monitoring. Most of the components have multiple uses.

4.5.3 Technical feasibility

All the technical resources are available in the internet. Most of the required knowledge is available in us and have sources where we can get the

knowledge.

4.5.4 Market feasibility

The project aims to solve the energy crisis faced by many of the countries in the world today. So the project has relevance. The project targets Upper-middle class and high class in the long term goal. The overall cost of the proposed system is afforded by common people.

4.5.5 Scheduling feasibility

The project duration is feasible. Within 2 months the project can be completed. The hardware connection and software development require maximum 2 months time.

4.5.6 Operational Feasibility

The project can be completed by individual person there is no team required. The hardware soldering and software coding can be done by individual person.

4.5.7 Financial feasibility

The proposed system is Cost efficient. All the components are available in the market. Also it can be bought through online.

Component cost :

- Arduino UNO R3 : 800
- NodeMCU : 400
- Current sensor : 150
- Voltage sensor : 75
- Relay : 165
- Adapter : 150

CHAPTER 5
SYSTEM DESIGN

5.1 Application Architecture

Architecture of Home Automation and Energy Monitoring System.

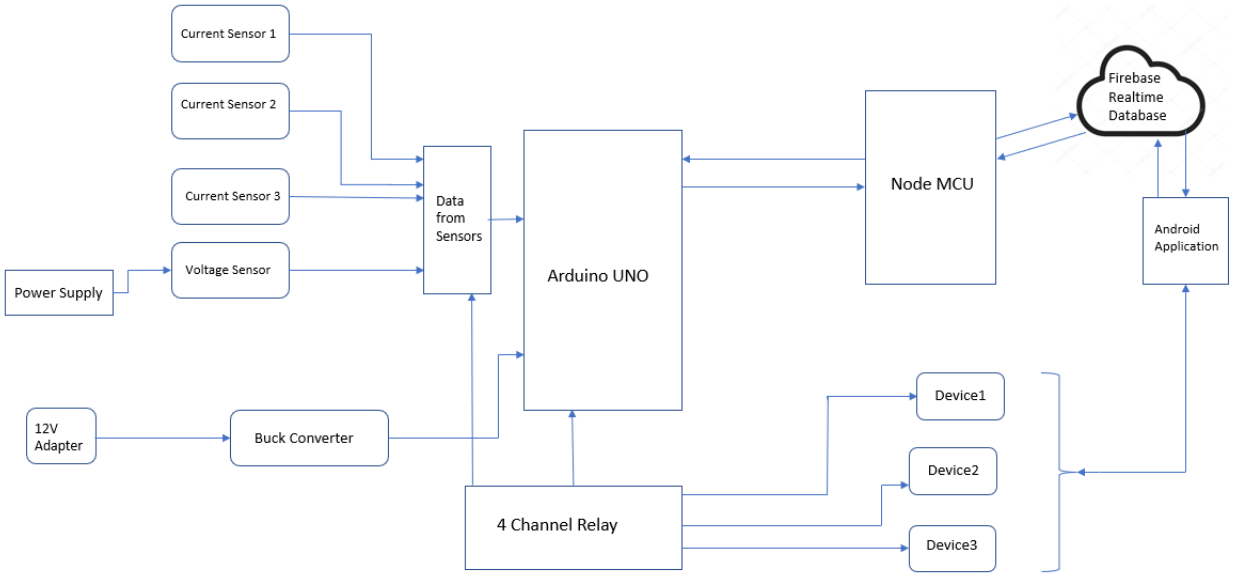


Fig. 5.1: Workflow Diagram

Steps for the implementation of my project :

1. Connect all the required components as per the circuit diagram.
2. Verify the Sensor Reading.
   Through serial monitor can check the readings.
3. Connect NodeMCU to WiFi. Check whether it is connected to WiFi or not.

```
#if defined(ESP32)
#include <WiFi.h>
#include <FirebaseESP32.h>
#elif defined(ESP8266)
#include <ESP8266WiFi.h>
#include <FirebaseESP8266.h>
#endif
#include <addons/RTDBHelper.h>

#define WIFI_SSID "Redmi 9A"
#define WIFI_PASSWORD "123456780"
#define DATABASE_URL "test-9911b-default-rtdb.firebaseio.com"
FirebaseData fbdo;
FirebaseAuth auth;
FirebaseConfig config;
unsigned long dataMillis = 0;
String prev_data1 = "0";
String prev_data2 = "0";
String prev_data3 = "0";
String d1;
String d2;
String d3;
String out;
void setup() {
  Serial.begin(4800);
  pinMode(D1, OUTPUT);
  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();
  digitalWrite(D1, HIGH);
  delay(1000);
  digitalWrite(D1, LOW);
  delay(1000);
  config.database_url = DATABASE_URL;
  config.signer.test_mode = true;
  Firebase.reconnectWiFi(true);
  Firebase.begin(&config, &auth); }
```

Fig. 5.2: WiFi Connection

4. Update Firebase fingerprint.

5. Transfer data to Firebase Realtime Database using firebase URL.

```
void loop() {
  //  Firebase.getInt(fbdo, F("/test/int")) ? String(fbdo.to<int>()).c_str()
  if (Firebase.getInt(fbdo, F("/devices/device1"))) {
    d1 = String(fbdo.to<int>()).c_str() ;
  }
  delay(100);
  if (Firebase.getInt(fbdo, F("/devices/device2"))) {
    d2 = String(fbdo.to<int>()).c_str() ;
  }
  delay(100);
  if (Firebase.getInt(fbdo, F("/devices/device3"))) {

    d3 = String(fbdo.to<int>()).c_str() ;
  }
  else {
    Serial.println( fbdo.errorReason().c_str());
  }

  delay(1000);
  if (prev_data1 != d1 or  prev_data2 != d2 or  prev_data3 != d3) {
    out = "*a" + d1 + "b" + d2 + "c" + d3 + "*";
    //Serial.println(out.length());
    if (out.length() >= 8) {
      Serial.println(out);
    }
    prev_data1 = d1;
    prev_data2 = d2;
    prev_data3 = d3;
    delay(10);
  }

  if (Serial.available() > 0) {

    String data = Serial.readStringUntil('\n');
    data.replace("\n", "");
    data.trim();
    int startIndx = data.indexOf('v');
    if (startIndx == 0) {
      if ( !Firebase.setString(fbdo, "/sensorData/sensor", data)) {
        Serial.println( fbdo.errorReason().c_str());
      }
    }
    else if (data.length() > 4) {

      if (!Firebase.setString(fbdo, "/info/device", data)) {
        Serial.println( fbdo.errorReason().c_str());
      }
    }
    delay(10);
  }
```

Fig. 5.3: Firebase Connection

6. Configure Firebase and Connect Firebase and Flutter Application. Firebase Documentation is available in the firebase official website.

7. Install the required plugins and add dependencies.

   cupertino_icons: 1.0.2
   firebase_database: 9.0.15
   firebase_core: 1.17.1 syncfusion_flutter_gauges: 20.1.61
   animated_text_kit: 4.2.2
   screenshot: 1.2.3
   image_gallery_saver: 1.7.1
   shared_preferences: 2.0.15

8. The Application is able to display the sensor reading and control appliances connected to it. The voltage reading is displayed through Radial gauge. Buttons are available to ON/OFF the devices.

9. Based on the data received the application is able to generate Electricity bill. The user can download the bill at the end of each month. After downloading the electricity bill all the data of the month is erased and begin to store data of next month.

5.2 Methodology

5.2.1 Introduction

The proposed system is implemented using Arduino UNO, Node MCU, Current Sensors, Voltage Sensor, 4-Channel Relay, Buck Converter. Integration of IoT with energy management system has been demonstrated to make a more effective and reliable system compared with the conventional energy management system. Arduino UNO, NodeMCU , Senors, Relay etc. used in Home Automation System. Firebase,Thing-speak,AWS,etc. has been used for the development of IoT which is compatible with the ESP 8266 WiFi module. It can also show, analyze, and calculate the required data and store them publicly or privately in the cloud. Remote control of domestic appliances and real time energy monitoring has also been made possible through this system.

5.2.2 Circuit Diagram

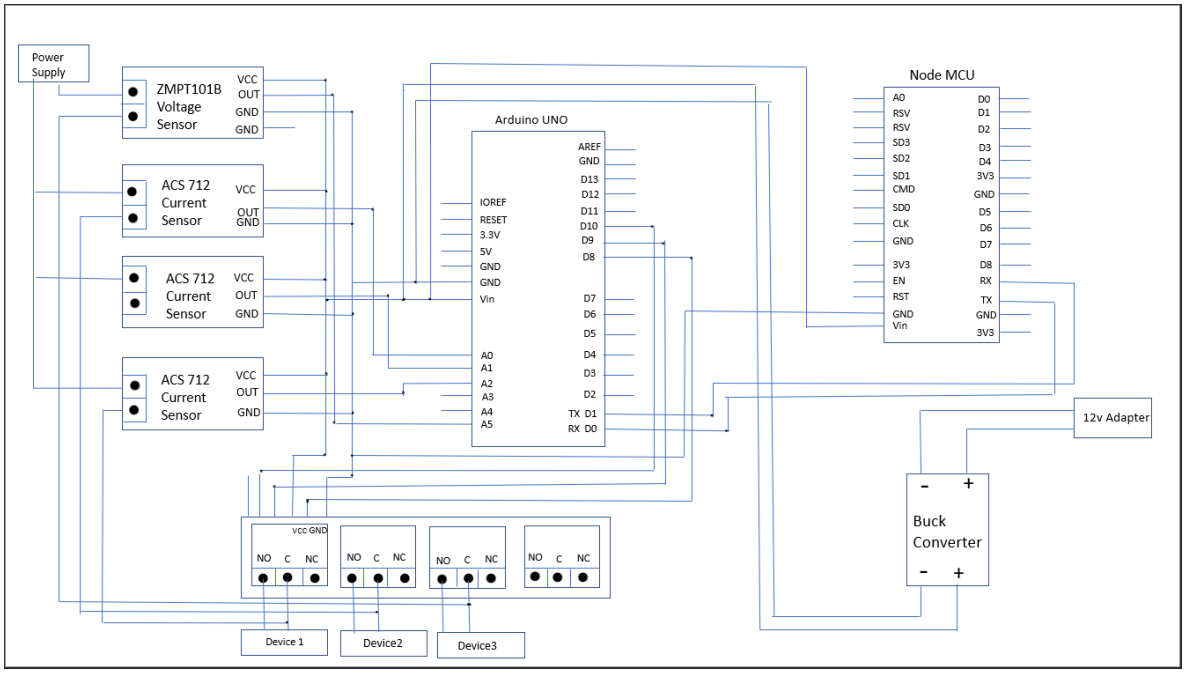


Fig. 5.4: Circuit Diagram

The circuit consists of Arduino Uno R3, Voltage Sensor, Current Sensor, Four Channel Relay, NodeMCU, Buck Converter, 12V Adapter.

Three Current Sensors are used. Each current sensor have 3 input pins GND, VCC, OUT. VCC connected to Arduino's Vin. GND connected to Arduino's GND. OUT is the signal pin which is connected to Arduino's Analog pins. There are 3 Current sensors. Therefore Analog pins A0, A1, A2 are used. The other side of current sensor have 2 connection one with Relay other is with power supply.

Voltage sensor is connected directly to power supply through a 2 pin plug. The GND pin is connected to GND of Arduino. The VCC is connected to Vin of Arduino. The OUT pin connected to Analog pin A5.

A 12V Adapter is used , which is connected to an Buck Converter in order to reduce the voltage to 5V. The output of Buck Converter is connected to Arduino's GND and Vin.

A 4 Channel relay is used. Devices are connected to Normally Opened and Common terminals of Relay. The input pins are connected to digital pins D0, D1, D2 . GND and VCC connected to GND and Vin of Arduino respectively.

NodeMCU is connected to Arduino through TX and RX pins. GND and VCC connected to GND and Vin of Arduino respectively.

5.2.3 Firebase

The Firebase Realtime Database lets you build rich, collaborative applications by allowing secure access to the database directly from client-side code. Data is persisted locally, and even while offline, realtime events continue to fire, giving the end user a responsive experience.

- Open firebase using https://console.firebase.google.com/
- Login using Gmail.
- Add a project.
- Enable google analytics.
- Create a Realtime Database.
- Add fields to receive data from NodeMCU.

5.2.3.1 Firebase Configuration

Install Firebase CLI. The Firebase CLI provides a variety of tools for managing, viewing, and deploying to Firebase projects. You can install the Firebase CLI using a method that matches your operating system, experience level, and/or use case. Regardless of how you install the CLI, you have access to the same functionality and the firebase command.

Steps to follow :

- Download the Firebase CLI binary for Windows.
- Access the binary to open a shell where you can run the firebase command.
- Install Node.js . Installing Node.js automatically installs the npm (the Node Package Manager) command tools
- Continue to log in and test the CLI.

Log in and test the Firebase CLI

- Log into Firebase using your Google account by running the following command: firebase login
- Test that the CLI is properly installed and accessing your account by listing your Firebase projects. Run the following command: Firebase projects:list
- Update to the latest CLI version : npm install -g firebase-tools
- To initialize Firebase: Firebase init
- Select realtime database and emulator. Select project name.
- Download the emulator.
- At the end of initialization, Firebase automatically creates the following two files at the root of your local app directory: A firebase.json configuration file that lists your project configuration. A .firebaserc file that stores your project aliases.
- From the root of your Flutter project, run the following command to install the plugin: flutter pub add firebase_database, flutter pub add firebase_core
- From any directory, run this command: dart pub global activate flutterfire_cli
- Then, at the root of your Flutter project directory, run this command: flutterfire configure –project=fir-a46ee
- This automatically registers your per-platform apps with Firebase and adds a lib/firebase_options.dart configuration file to your Flutter project.

Initialize Firebase and add plugins

- To initialize Firebase, call Firebase.initializeApp from the firebase_core package with the configuration from your new firebase_options.dart file:

```
import 'package:firebase_core/firebase_core.dart';
import 'firebase_options.dart';

// ...

await Firebase.initializeApp(
    options: DefaultFirebaseOptions.currentPlatform,
);
```

- Then, add and begin using the Flutter plugins for the Firebase.
- flutter pub add PLUGIN_NAME
  flutter pub add firebase_database

5.2.4 Android Studio Setup

Android Studio offers a complete, integrated IDE experience for Flutter.

Install the Flutter and Dart plugins

- Go to the site https://docs.flutter.dev/get-started/install/windows.
- Download the installation bundle to get the latest stable release of the Flutter SDK: flutter_windows_3.0.4stable.zip
- Extract the zip file and place the contained flutter in the desired installation location for the Flutter SDK (for example, C:).
- Do not install Flutter to a path that contains special characters or spaces. Do not install Flutter in a directory like C:Files that requires elevated privileges.
- If you wish to run Flutter commands in the regular Windows console, take these steps to add Flutter to the PATH environment variable: From the Start search bar, enter 'env' and select Edit environment variables for your account. Under User variables check if there is an entry called Path: If the entry exists, append the full path to "flutter/bin" using ; as a separator from existing values. If the entry doesn't exist, create a new user variable named Path with the full path to flutter/bin as its value. You have to close and reopen any existing console windows for these changes to take effect.
- Run flutter doctor: flutter doctor
- If flutter doctor returns that either the Flutter plugin or Dart plugin of Android Studio are not installed, move on to Set up an editor to resolve this issue.

- Install Android Studio. Android Studio, version 2020.3.1 (Arctic Fox) or later.
- Install Flutter and Dart Plugins.
- Create new flutter project.
- Flutter SDK path added. Dart SDK path added.
- Minimum SDK version set to 31.

5.3 Flow Chart

A flowchart is a type of diagram that represents a workflow or process. A flowchart can also be defined as a diagrammatic representation of an algorithm, a step-by-step approach to solving a task. The flowchart shows the steps as boxes of various kinds, and their order by connecting the boxes with arrows. Flowcharts are used in designing and documenting simple processes or programs. Like other types of diagrams, they help visualize what is going on and thereby help understand a process, and perhaps also find less-obvious features within the process.

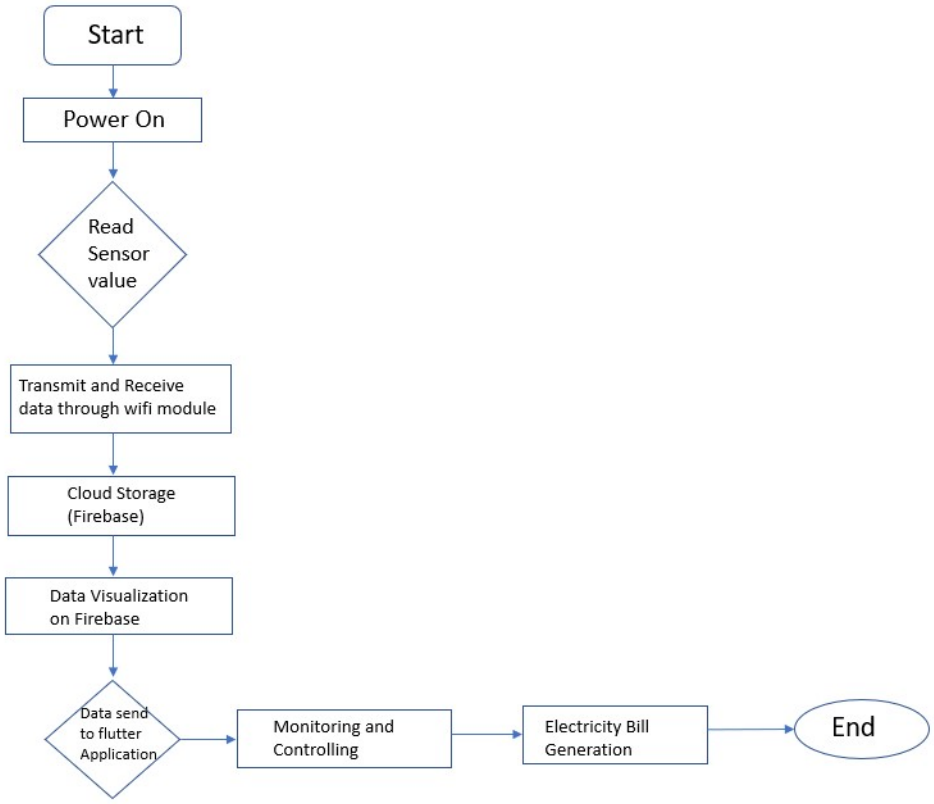


Fig. 5.5: Flow Chart

CHAPTER 6
SYSTEM IMPLEMENTATION

6.1 System Implementation

6.1.1 Hardware Requirements

The main hardware components required for the implementation of the system are Arduino UNO, Node MCU, Current Sensor, Voltage Sensor, 4 Channel Relay,Buck Converter.

6.1.1.1 Arduino UNO

Arduino is a deployment board that incorporates an Atmega328P microcontroller. The micro controller integrated circuit (IC) has 14 digital Input and Output (I/O) pins . About four of those pins are used in this application for switching the home appliances. The development board includes power jack, USB connections and reset button.

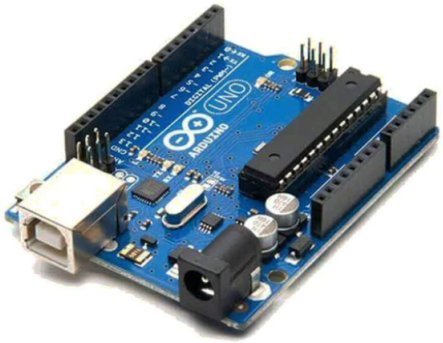


Fig. 6.1: Arduino UNO

Features :

- Inexpensive - Arduino boards are relatively inexpensive compared to other microcontroller platforms. The least expensive version of the Arduino module can be assembled by hand, and even the pre-assembled Arduino modules cost less than $50
- Cross-platform - The Arduino Software (IDE) runs on Windows, Macintosh OSX, and Linux operating systems. Most microcontroller systems are limited to Windows.
- Simple, clear programming environment - The Arduino Software (IDE) is easy-to-use for beginners, yet flexible enough for advanced users to take advantage of as well. For teachers, it's conveniently based on the Processing programming environment, so students learning to program in that environment will be familiar with how the Arduino IDE works.
- Open source and extensible software - The Arduino software is published as open source tools, available for extension by experienced programmers. The language can be expanded through C++ libraries, and people wanting to understand the technical details can make the leap from Arduino to the AVR C programming language on which it's based. Similarly, you can add AVR-C code directly into your Arduino programs if you want to.
- Open source and extensible hardware - The plans of the Arduino boards are published under a Creative Commons license, so experienced circuit designers can make their own version of the module, extending it and improving it. Even relatively inexperienced users can build the breadboard version of the module in order to understand how it works and save money.

6.1.1.2 Node MCU

Wi-Fi (wireless fidelity) is a wireless communication technology which provides a hotspot to connect communicating module. The router will consign an inimitable IP address to the module for forming a connection between smart mobile phone and ESP8266-01module. NodeMCU which will provide a facility to connect devices to internet. This module is the System on Chip Module which consists of integrated TCP/IP protocol stack that helps the controller to access the Wi-Fi network. This type of module is highly cost effective, works effectively in the on board processing and it is highly compact too. This module requires 3.3 V to energize it and it is automated with FTDI. The pins in this module are power pins which includes 3.3 V DC and GND, RX / TX, CHPD to enable the chip and 2 General Purpose I/O.

Fig. 6.2: Node MCU

Features:

- Wi-Fi Module – ESP-12E module similar to ESP-12 module but with 6 extra GPIOs.
- USB – micro USB port for power, programming and debugging
- Headers – 2x 2.54mm 15-pin header with access to GPIOs, SPI, UART, ADC, and power pinsMisc – Reset and Flash buttons
- Power – 5V via micro USB port
- Dimensions – 49 x 24.5 x 13mm

6.1.1.3 ACS 712 Current Sensor

The ACS712 is a fully integrated, hall effect-based linear current sensor with 2.1kVRMS voltage isolation and a integrated low-resistance current conductor. Technical terms aside, it's simply put forth as a current sensor that uses its conductor to calculate and measure the amount of current applied.

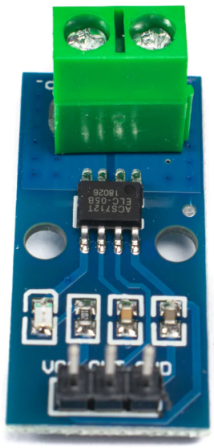


Fig. 6.3: ACS 712 Current Sensor

The features of ACS712 include :

- 80kHz bandwith
- 66 to 185 mV/A output sensitivity
- Low-noise analog signal path
- Device bandwith is set via the new FILTER pin
- 1.2 m internal conductor resistance
- Total output error of 1.5
- Stable output offset voltage.
- Near zero magnetic hysteresis

6.1.1.4 ZMPT101B Voltage Sensor

ZMPT101B a voltage transform ideal to measure the AC voltage. It has high accuracy, good consistency for voltage and power measurement up to 250V AC. Easy to use and comes with a multi turn trim potentiometer for adjusting the ADC output.

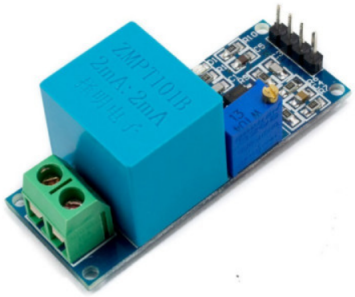


Fig. 6.4: ZMPT101B Volatge Sensor

Specifications:

- Output Signal: Analog 0-5V
- Operating Voltage: DC 5V-30V
- Measure within 250V AC
- Rated input current: 2mA
- Size: 49.5 mm x 19.4 mm
- Onboard micro-precision voltage transformer
- Analog output corresponding quantity can be adjusted
- Good consistency for voltage and power measurement
- Active output single-phase AC voltage sensor module

6.1.1.5 Four Channel Relay

The 4 Channel Relay Module is a convenient board which can be used to control high voltage, high current load such as motor, solenoid valves, lamps and AC load. It is designed to interface with microcontroller such as Arduino, PIC and etc. The relays terminal (COM, NO and NC) is being brought out with screw terminal. It also comes with a LED to indicate the status of relay. The four-channel relay module contains four 5V relays and the associated switching and isolating components, which makes interfacing with a microcontroller or sensor easy with minimum components and connections. The contacts on each relay are specified for 250VAC and 30VDC and 10A in each case, as marked on the body of the relays.

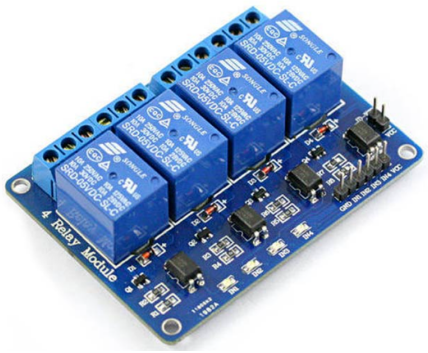


Fig. 6.5: Four Channel Relay

Specifications:

- Digital output controllable
- Compatible with any 5V microcontroller such as Arduino.
- Rated through-current: 10A (NO) 5A (NC)
- Control signal: TTL level
- Max. switching voltage 250VAC/30VDC
- Max. switching current 10A
- Size: 76mm x 56mm x 17mm

6.1.1.6 Buck Converter

A buck converter or step down voltage regulators. It's a type of DC-DC converter, so it accomplishes the task using a few transistor switches and an inductor. The Buck Converter is used in SMPS circuits where the DC output voltage needs to be lower than the DC input voltage. The DC input can be derived from rectified AC or from any DC supply. It is useful where electrical isolation is not needed between the switching circuit and the output, but where the input is from a rectified AC source, isolation between the AC source and the rectifier could be provided by a mains isolating transformer.

Fig. 6.6: Buck Converter

The switching transistor between the input and output of the Buck Converter continually switches on and off at high frequency. To maintain a continuous output, the circuit uses the energy stored in the inductor L, during the on periods of the switching transistor, to continue supplying the load during the off periods. The circuit operation depends on what is sometimes also called a Flywheel Circuit. This is because the circuit acts rather like a mechanical flywheel that, given regularly spaced pulses of energy keeps spinning smoothly (outputting energy) at a steady rate.

CHAPTER 7
RESULTS AND DISCUSSION

7.1 Flutter Application

The Flutter framework is a software development kit (SDK) for mobile apps created by Google. From the tool it is possible to create, with a single database, applications that run on Android and iOS platforms.

This use of single code has profound impacts on mobile app development. The cross-platform application helps to save business resources, in addition to preventing applications from getting different functionality.

Flutter's code re-usability allows you to write just one codebase and use it on not only for mobile Android and iOS but even for web, desktop and more. This cuts development time significantly, removes cost and enables you launch your app that much faster.

Features :

- Hot reload

  Hot reload makes it possible to see the changes in the code instantly reflected on the UI. This speeds up the process to work on the outlook of the application; moreover, it enables developers to correct errors that save cost and effort.

- Cross-platform development

  Flutter enables developers to write code that works on different platforms. Two different applications can use the same codebase. In addition to sharing the UI code, the UI itself is also shareable. This

makes maintenance of the single codebase much easier as opposed to different codes for different platforms.

- Widget library

  Everything in Flutter is defined as a widget. A widget can be a color, padding, or menu. Flutter is capable of creating complex widgets that can be customized according to the requirement of the application. Built-in widgets are also available for usage. Some examples include Cupertino pack and Material Design, which have sets of widgets that provide a glitch-free user experience.

- Native Performance

  In Flutter, platform-specific widgets are provided for Google Fuchsia, Android, and iOS. These widgets can be integrated into the Flutter application to make use of the different platform-dependent functionalities. Existing Java, Swift, and Objective-C codes can be used to utilize native features such as camera and geolocation. Therefore, Flutter can easily incorporate third-party integrations and APIs.

- Open-source

  Google introduced Flutter as an open-source platform. Various design options can be explored by developers to create Flutter applications. User-friendly applications can be created with Material Design and Cupertino widgets. Flutter Form is a community of Flutter enthusiasts who come together to answer questions related to Flutter and discuss it. Flutter is free of cost and has detailed documentation and communities available online.

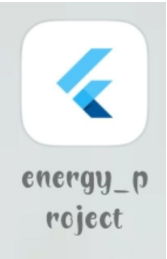


Fig. 7.1: Flutter Application Icon

7.1.1 Code folder structure

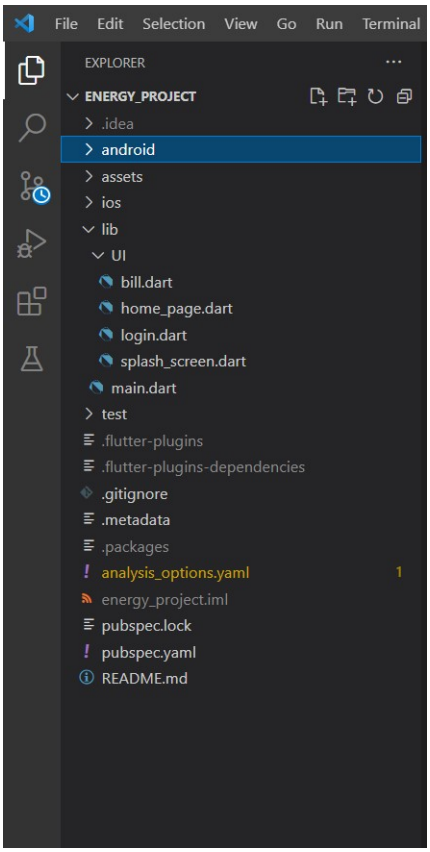


Fig. 7.2: Code Folder Structure

7.2 Splash Screen

A splash screen is a launch screen, start screen, or boot screen, which is a graphical control element containing the image, logo, and current version of the software. It is the first screen of the app that displays whenever the application is loading. It can also be the app's welcome screen that provides

a simple initial experience when a mobile game or program is launching. The splash screen is just a display screen that allows users to look something while the hardware is loading to present software to the user.

The common elements of a splash screen contain a company name and logo or a title. The most common example of a splash screen is the Flutter logo on starting the Flutter application or Microsoft logo while starting the Microsoft operating system.

Splash Screen Characteristics:

- It is mainly used for branding or identity recognition of the application and puts the branding impression to users.
- It can also be used to show some loading progress indicator while the hardware is loading to present software to the user.
- When the loading of the splash screen completes, the user gets another functional screen that would be home screen or dashboard, then is forgotten. Once the loading completes, we cannot press the back button to return the splash screen.
- The smart home application uses an animated splash screen. used is animated_text_kit: 4.2.2

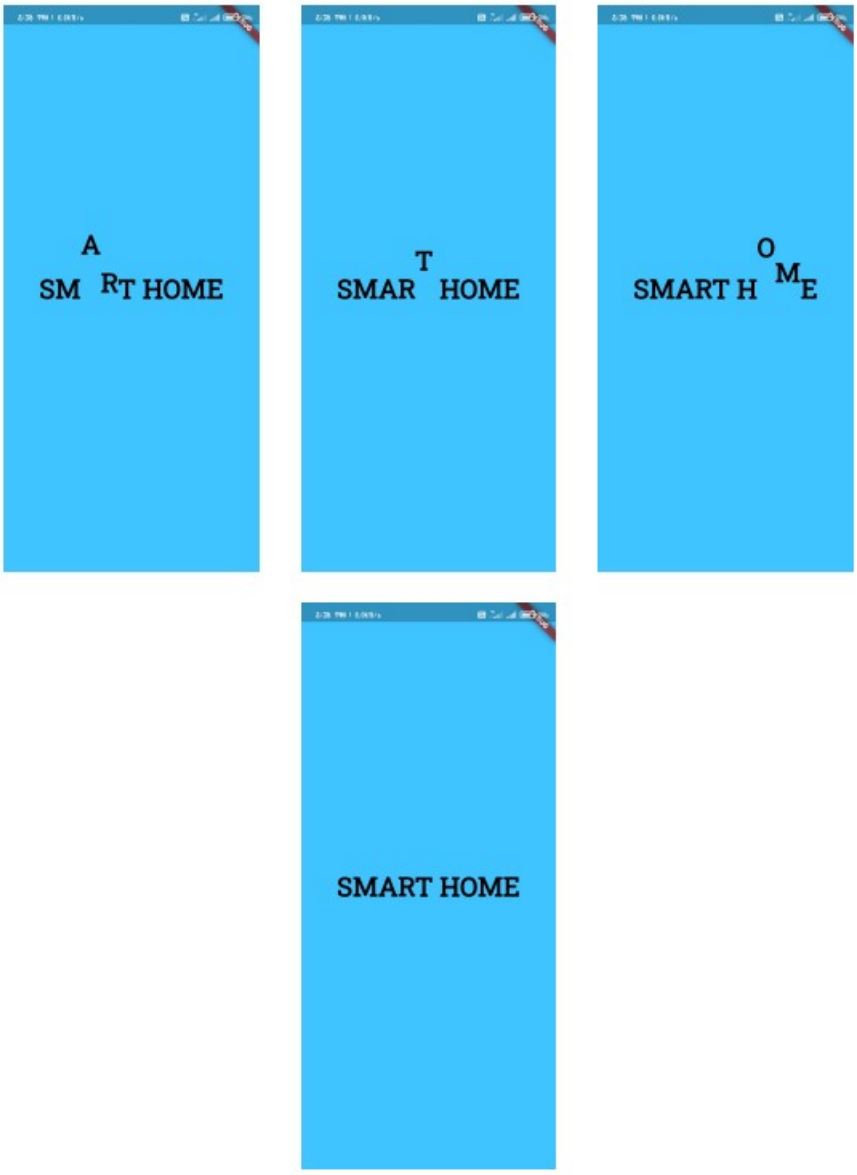


Fig. 7.3: Animated Splash Screen

```
import 'package:flutter/material.dart';
import 'package:animated_text_kit/animated_text_kit.dart';
import 'login.dart';

class SplashScreen extends StatefulWidget {
  const SplashScreen({Key? key}) : super(key: key);

  @override
  State<SplashScreen> createState() => _SplashScreenState();
}

class _SplashScreenState extends State<SplashScreen> {

  @override
  void initState() {
    super.initState();
    Future.delayed(const Duration(seconds: 5), () {
      Navigator.pop(context);
      Navigator.push(
          context,
          MaterialPageRoute(
            builder: (context) => const LoginPage(),
          ));
    });
  }

  @override
  Widget build(BuildContext context) {
    return Container(
      alignment: Alignment.center,
      width: MediaQuery.of(context).size.width,
      height: MediaQuery.of(context).size.height,
      color:  Colors.lightBlueAccent,
      child: AnimatedTextKit(
        animatedTexts: [
          WavyAnimatedText('SMART HOME'
            , textStyle: const TextStyle(
                fontSize: 32.0,
                decoration: TextDecoration.none,
                fontWeight: FontWeight.bold,
                color: Colors.black,
                fontFamily: 'RobotoSlab'
            ),
          ),
        ],
        isRepeatingAnimation: true,
        onTap: () {
          print("Tap Event");
        },
      ),



      /*const Text('ENERGY PROJECT',
        style: TextStyle(
          decoration: TextDecoration.none,
          fontSize: 40,
          fontFamily: 'RobotoSlab',
          color: Colors.white,
        ),
      ),*/
    );
  }
}
```

Fig. 7.4: Splash Screen

7.3 Login Screen

A login application is the screen asking your credentials to login to some particular application. If any of the field does not enter a snack bar will pop up and ask you to enter credentials.

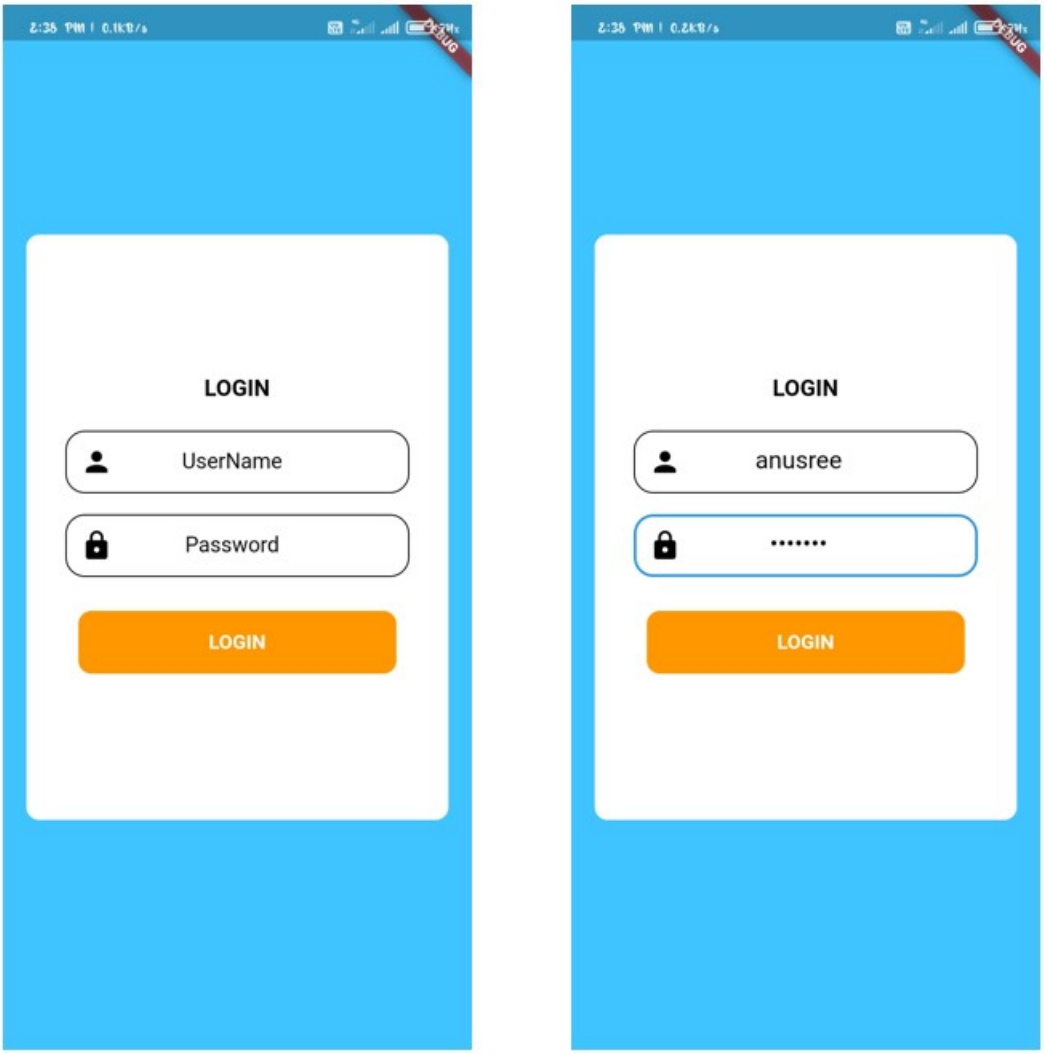


Fig. 7.5: Login Screen

7.4 Home Screen

The Home Screen displays the voltage and current readings. Also it provides buttons to control devices. The voltage reading is displayed using radial gauge.

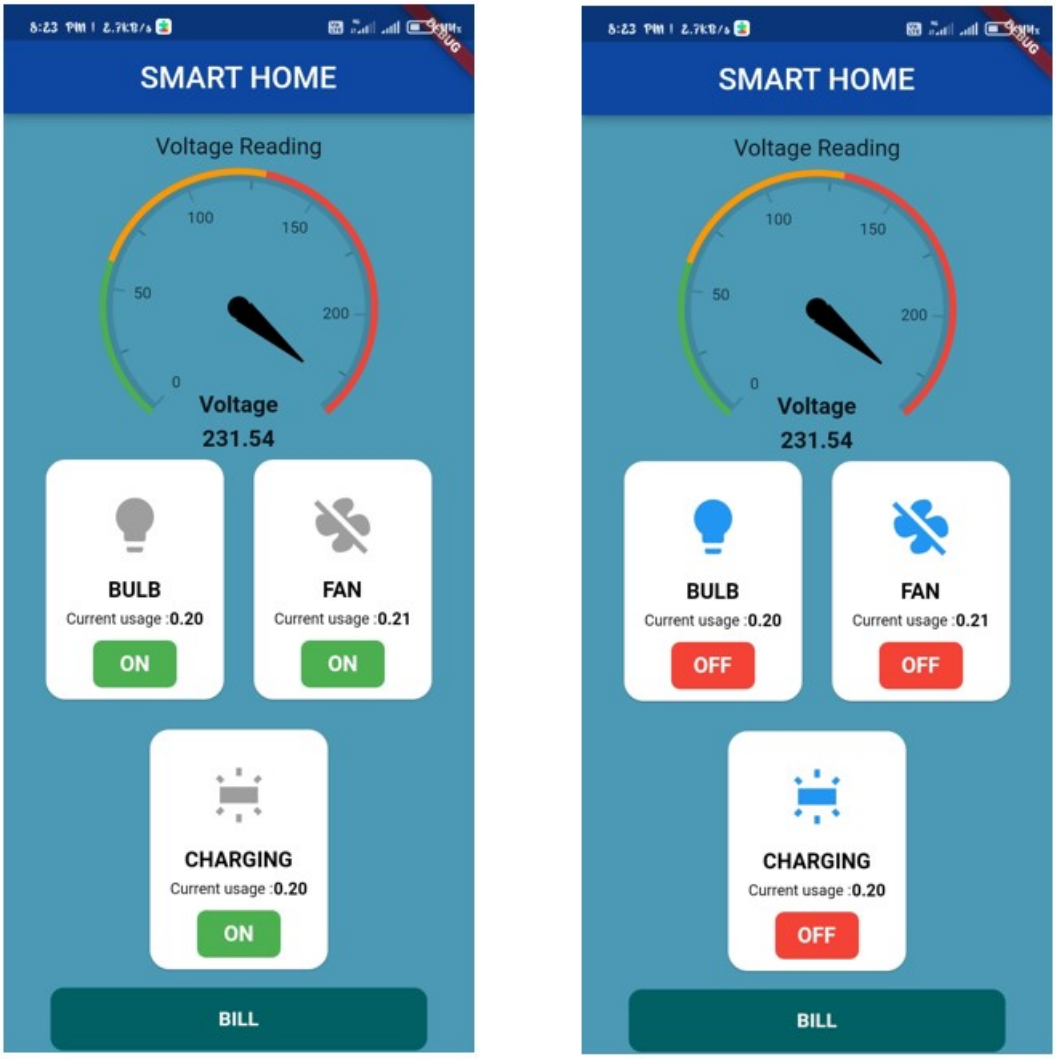


Fig. 7.6: Home Screen

```
DatabaseReference ref = FirebaseDatabase.instance.ref();

void firebaseData(){
  DatabaseReference _dbRef = FirebaseDatabase.instance.ref();
  _dbRef.once().then((DatabaseEvent databaseEvent) {
    print("Database Data" + databaseEvent.snapshot.value.toString());
    sensorData = (databaseEvent.snapshot.value as Map<Object?,dynamic>)['sensorData']['sensor'].toString();
    setState(() async{

      voltage = sensorData.split('a')[0].toString();
      firstCurrent = sensorData.split('a')[1].toString();
      secondCurrent = firstCurrent.split('b')[1].toString();

      exactVoltage = double.parse(voltage.split('v')[1].toString());
      firstDevice = firstCurrent.split('b')[0].toString();
      secondDevice = secondCurrent.split('c')[0].toString();
      thirdDevice = sensorData.split('c')[1].toString();
      if((databaseEvent.snapshot.value as Map<Object?,dynamic>)['devices']['device1'] == 1){
        firstButton = true;
      }
      if((databaseEvent.snapshot.value as Map<Object?,dynamic>)['devices']['device2'] == 1){
        secondButton = true;
      }
      if((databaseEvent.snapshot.value as Map<Object?,dynamic>)['devices']['device3'] == 1){
        thirdButton = true;
      }
```

Fig. 7.7: Data Accessing From Firebase

```
child:SfRadialGauge(
    title: const GaugeTitle(text: "Voltage Reading"),
    enableLoadingAnimation: true,
    animationDuration: 4500,
    axes: <RadialAxis>[
      RadialAxis(minimum: 0,maximum: 240,
          ranges: <GaugeRange>[
            GaugeRange(startValue: 0,endValue: 60, color: Colors.green, startWidth: 5,endWidth: 5),
            GaugeRange(startValue: 60,endValue: 130,color: Colors.orange,startWidth: 5,endWidth: 5),
            GaugeRange(startValue: 130,endValue: 240,color: Colors.red,startWidth: 5,endWidth: 5)
          ],
          pointers:  <GaugePointer>[
            NeedlePointer(value: exactVoltage, )
          ],
          annotations: <GaugeAnnotation>[
            GaugeAnnotation(
                widget: Container(
                    child:  Column(
                      children:  [
                        const Text('Voltage',style: TextStyle(fontSize: 15,fontWeight:FontWeight.bold)),
                        const SizedBox(height: 5,),
                        Text(exactVoltage.toString(),style: const TextStyle(fontSize: 15,fontWeight:FontWeight.bold)),
                      ],
                    )
                ),
                angle: 90,
                positionFactor: 1.7),
          ]
      )]
),
```

Fig. 7.8: Radial Gauge

7.5 Electricity Bill Calculation

An electricity bill is a document showing you how much money you owe for electricity, that you have used.

Calculation of Electric Energy Consumption

The following formula is used for electrical energy consumption.

$E = P \times t \ldots$ (Wh)

$E = P \times t \div 1000 \ldots$ (kWh)

Consumed Energy = Energy Used in Watts x Time in Hours

Where:

E = Electrical Energy (Consumed in kWh)

P = Power in Watts

t = Time in hours per day

Wh (Watt-hour) is a small unit, so we divide the consumed energy on 1000 to get the value of energy in kWh instead of Wh.

```
class _BillState extends State<Bill> {

  double firstunit = 0.0;
  double secondunit = 0.0;
  double thirdunit = 0.0;

  final Future<SharedPreferences> _prefs = SharedPreferences.getInstance();

  @override
  void initState(){
    getPref();
    super.initState();
  }

  List downloadBill = [29,30,31 ];

  Uint8List? image;

  double totalTime1 = 0.0;
  double totalTime2 = 0.0;
  double totalTime3 = 0.0;
  double voltage = 0.0;
  double firstcurrent = 0.0;
  double secondcurrent = 0.0;
  double thirdcurrent = 0.0;

  bool billsaved = false;

  _saved() async {
    final result = await ImageGallerySaver.saveImage(image!);
    print("File Saved to Gallery");
  }

  void getPref() async{
    SharedPreferences sp1 = await _prefs;
    setState((){
      totalTime1 = sp1.getDouble('device1')!;
      totalTime2 = sp1.getDouble('device2')!;
      totalTime3 = sp1.getDouble('device3')!;
      voltage = sp1.getDouble('voltage')!;
      firstcurrent = sp1.getDouble('firstcurrent')!;
      secondcurrent = sp1.getDouble('secondcurrent')!;
      thirdcurrent = sp1.getDouble('thirdcurrent')!;

      firstunit = ((voltage * firstcurrent * totalTime1)/1000);
      secondunit = ((voltage * secondcurrent * totalTime2)/1000);
      thirdunit = ((voltage * thirdcurrent * totalTime3)/1000);
    });
  }
```

Fig. 7.9: Electricity Bill Sample Code

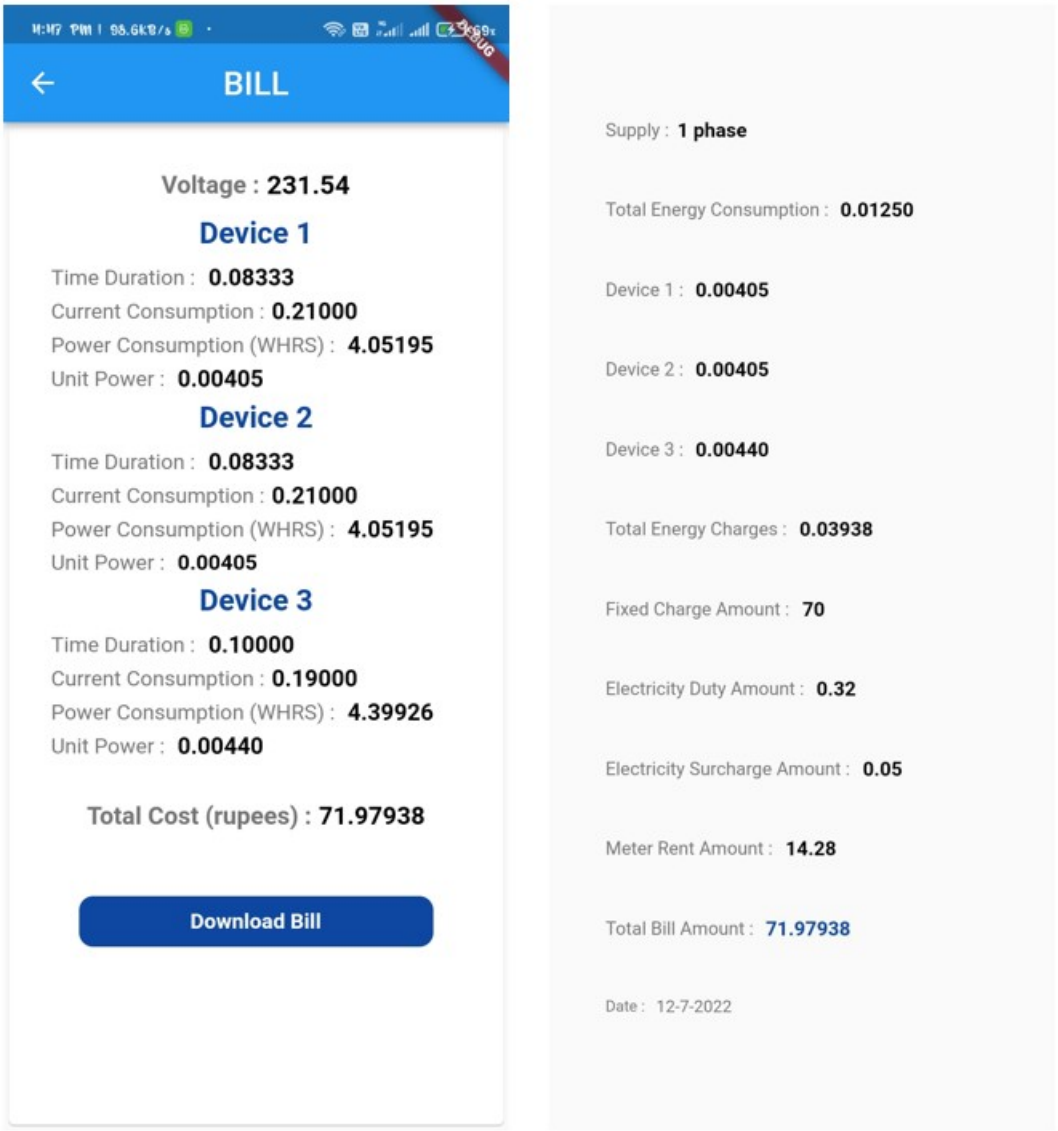


Fig. 7.10: Electricity Bill

7.6 Firebase

Firebase Realtime Database is a cloud-hosted database that supports iOS, Android, Web, C++ and Unity platforms. Realtime means that any changes in data are reflected immediately across all platforms and devices within milliseconds. Data From NodeMCU send to Firebase Realtime database. Data from firebase can be accessed by flutter application by firebase configuration.

The user can view data real-time and also can control the devices. If the user forget to take his phone , he can monitor and control the devices inside home using firebase.

Fig. 7.11: Firebase Realtime Database

7.7 GitHub Repository

https://github.com/AnusreeJnanakrishnanGEC/S4-Main-Project.git

7.8 Testing

Each functionality of the proposed system is tested. **Unit Testing**is a great testing technique. It allows us to test a single function, method, or class individually. This is a fantastic way to make sure that a unit of logic produces the expected value.

7.8.1 Hardware functions testing

The sensor reading are verified from serial monitor. The data from Arduino is send to NodeMCU is also verified via serial monitor.

7.8.2 Firebase functions testing

Manually tested whether the value in the firebase changes affects the controlling of devices. The value 1 and 0 changes randomly to check whether the devices are turn OFF or ON respectively.

7.8.3 Flutter application and firebase connection testing

Sample code to test whether any data present in firebase is able to retrieve. Manually test the button to check whether devices is able to control.

7.8.4 Bill generation

The bill generation button is checked whether it is working properly. The bill download button checked. The download bill is in image format. Verified whether the bill is downloaded in the gallery. After downloaded the bill the previous data erased or not is checked.

CHAPTER 8
CONCLUSION

The existing systems does not provide a cost efficient system. In such a context this application has great importance. The sensors are able to read the data, Arduino UNO will store the data. The data from Arduino is send to NodeMCU by serial communication. NodeMCU is a WiFi module which is able to transfer the data to cloud. The cloud used here is Firebase Realtime Database which is able to access the data and show it in real time. By Firebase configuration the flutter application is able to communicate with firebase, there by send and receive data. The application is very much helpful for users. If the user forgot to take his phone or if it's battery dead then user can control and monitor the appliances through Firebase. The application is user friendly. User can monitor the real time energy usage of appliances. The user is able to control the appliances from anywhere in the world. The main functionality of the application is generation of electricity bill. The user is always bother about the energy usage of devices . So that there is always a restriction on each appliances to work continuously and making electricity cost maximum.

CHAPTER 9
FUTURE ENHANCEMENT

The Home automation and Energy monitoring system can be made advanced by using sensors with more features and capacity. Also we can add some more feature in application. The user can control the speed of fan and control the intensity of light. The user can add security features in application. By setting a camera can recognize people who are not belonging to the house. Using more sensors can detect fire, gas leakage, etc.

BIBLIOGRAPHY

[1] M. J. Iqbal et al., "Smart Home Automation Using Intelligent Electricity Dispatch," in IEEE Access, vol. 9, pp. 118077-118086, 2021, doi: 10.1109/ACCESS.2021.3106541.

[2] M. Vijayaragavan and S. S. Darly, "Automatic Electricity Bill Calculation Using Arduino," 2019 International Conference on Smart Structures and Systems (ICSSS), 2019, pp. 1-3, doi: 10.1109/ICSSS.2019.8882859.

[3] A. Chaudhari, B. Rodrigues and S. More, "Automated IOT based system for home automation and prediction of electricity usage and comparative analysis of various electricity providers: SmartPlug," 2016 2nd International Conference on Contempo- rary Computing and Informatics (IC3I), 2016, pp. 390-392, doi: 10.1109/IC3I.2016.7917995.

[4] Swapnil Talkar, Ruhi Bajaj. IOT based Home Automation System for Electricity Usage, International Journal of Advance Research, Ideas and Innovations in Tech- nology, www.IJARIIT.com.

[5] Automatic Electricity Bill Generating System, N. Rajathi, N. Suganthi, Shilpa R, International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277- 3878, Volume-7 Issue-4S, November 2018.

[6] Wireless Energy Meter with Home Automation Vigneshwaran S1, Rakesh E2, Roshith H3, Vimalraj M4, Mrs. Lakshmi S5 ,International Research Journal of Engi- neering and Technology (IRJET) e-ISSN: 2395-0056, p-ISSN: 2395-0072, Volume: 07 Issue: 02 — Feb 2020 [7] Automation of Electricity Management Sayali Gore , Anmol Bhegade , Priyanka Dhaybar , International Journal Of Engineering And Computer Science

ISSN:2319-7242 Volume 4 Issue 2 February 2015, Page No. 10250-10254

.

[7] Live Power Consumption Monitoring and Home Automation using Google Assis- tant Saran Kumar.R , Srevignesh , Vijay , Sivaraman and T.Tamilarasan, International Advanced Research Journal in Science, Engineering and Technology Vol. 7, Issue 2, February 2020.