INDEX

TABLE OF CONTENTS

LIST OF FIGURES

ABSTRACT

The Automatic Electricity Bill Generation System is designed to streamline the process of calculating and issuing electricity bills with minimal human intervention. By leveraging smart meters, microcontrollers, or software-based algorithms, the system efficiently records real-time electricity consumption, applies appropriate tariff rates, and generates detailed billing statements for users.

Key features of this system include automated consumption tracking, real-time bill computation, digital invoice generation, and secure payment integration. The implementation of automated notifications and user authentication mechanisms ensures accuracy and accessibility while reducing manual errors and operational costs. Furthermore, the system enhances efficiency by eliminating the need for physical meter readings, contributing to a more sustainable and intelligent utility management framework.

The project serves as a stepping stone toward smart grid integration, offering scalability for future enhancements such as IoT-based monitoring, AI-driven consumption forecasting, and blockchain-secured billing. By modernizing electricity billing, this system improves reliability, transparency, and user convenience, making utility management more seamless and efficient.

Chapter 1: INTRODUCTION

The advent of technology has transformed numerous aspects of daily life, including how utilities are managed and billed. The Smart Automatic Electricity Bill Generation System is a groundbreaking solution that automates the process of calculating electricity consumption and generating bills with high accuracy and minimal human intervention. This system leverages modern technologies such as smart meters, data processing algorithms, and digital payment platforms to enhance efficiency and provide a seamless experience for both consumers and utility providers.

1.1 Purpose of the Project:

The primary objective of this project is to eliminate manual errors, reduce labor costs, and enhance transparency in electricity billing. By automating the billing process, this system aims to ensure accurate calculation of electricity consumption and facilitate timely communication of billing details to users. Additionally, the project seeks to support sustainability goals by reducing the need for paper-based bills and streamlining energy resource management.

1.2 Scope of the Project:

The scope of this project extends beyond traditional billing systems by incorporating real-time data tracking, user-friendly interfaces, and secure online payment integration. It offers scalability to adapt to future advancements such as Internet of Things (IoT) technologies for advanced monitoring, AI-driven insights to predict consumption trends, and Blockchain-based security for tamper-proof billing. This system can be implemented in residential, commercial, and industrial setups, serving as a precursor to fully automated smart grid networks, thereby transforming the landscape of utility management.

1.3 Features of the project:

Here is a stepwise breakdown of the features of the Automatic Electricity Bill Generation System:

1. Data Collection:
   Retrieve real-time electricity consumption data from smart meters or user inputs.
2. Bill Calculation:
   Compute electricity charges using predefined tariff rates, considering usage, taxes, and other applicable fees.
3. Invoice Generation:
   Create digital bills with details like consumption units, total charges, and payment deadlines.
4. User Interface:
   Provide a simple and intuitive platform for users to access billing information.
5. Notification System:
   Automatically send alerts regarding bill availability, due dates, and payment confirmations.
6. Error Prevention:
   Reduce manual errors by automating calculations and communication processes.
7. Security Features:
   Implement user authentication and encryption to safeguard sensitive billing information.
8. Scalability:
   Ensure the system is adaptable to future technologies like IoT, AI, and blockchain.
9. Customization:
   Allow modifications tailored to different user categories (residential, commercial, or industrial).

1.4 Hardware and Software Requirements:

1.4.1 Hardware Requirements:

i. ESP32 WiFi Module: Enables wireless communication for data transmission.
ii. ZMPT101B AC Voltage Sensor Module: Measures AC voltage accurately for monitoring.
iii. SCT-013-030 Non-invasive AC Current Sensor: Tracks current consumption without direct connection.
iv. 16x2 LCD Display: Presents consumption details and billing information.
v. Potentiometer (10K): Adjusts LCD contrast.
vi. Resistors (10K & 100 Ohm): Provides current control in the circuit.
vii. Capacitor (10uF): Stabilizes voltage variations.
viii. Connecting Wires: Facilitates inter-component connections.
ix. Breadboard: Serves as the platform for assembling and testing circuits.

1.4.2 Software Requirements:

i. EmonLib Library: Used for real-time electricity monitoring, measuring voltage and current to calculate true average quantities for energy metering.
ii. Blynk Library: Connects hardware (like ESP32) to the Blynk Cloud for IoT integration, enabling remote control, data visualization, and management.
iii. Programming Language: C++, or similar for system logic development.
iv. Integrated Development Environment (IDE): Arduino IDE or similar for coding and compiling.
v. Encryption Tools: Ensures data security during transmission.

Chapter 2: LITERATURE SURVEY

Smart meter technologies are essential for modernizing energy management, enabling real-time communication between consumers and utilities to enhance grid reliability and support the shift to renewable energy sources. These systems are critical for digitizing the energy sector, particularly in smart cities, by providing accurate consumption data and facilitating efficient billing. Knayer and Kryvinska (2022) conducted a systematic literature review to investigate the acceptance and benefits of smart meters, with a specific focus on organizations, using the Technology Acceptance Model (TAM) to frame their analysis. Their study highlights the potential of smart meters while identifying significant research gaps, particularly in organizational contexts.

The methodology employed by Knayer and Kryvinska (2022) was rigorous, involving a systematic literature review process. They formulated research questions to explore factors influencing smart meter acceptance in organizations and their associated benefits. Using mind map software, they refined 300 search terms, which were then used to search four major databases—Google Scholar, Web of Science, Scopus, and Wiley Online Library—for peer-reviewed papers published between 2000 and 2021. The search process progressed from a basic search yielding 4.6 million documents to a fine and exact search, ultimately narrowing down to 47 relevant publications. These papers were analyzed based on eight criteria, including origin, year, and data type, and subjected to a SWOT analysis to identify strengths, weaknesses, opportunities, threats, and research gaps.

The review revealed that most studies on smart meter adoption focus on households, with limited attention to organizations, particularly small businesses. The Technology Acceptance Model (TAM), proposed by Davis (1989), posits that higher perceived benefits and ease of use lead to positive attitudes and greater intention to adopt technology. While TAM has been validated for households (e.g., Chen et al., 2017; Chawla et al., 2020a, 2020b), no studies have empirically confirmed its applicability to organizations, marking a significant research gap. Technically, smart meters offer benefits like real-time pricing, time-of-use tariffs, and integration with vehicle-to-grid systems (Almeida et al., 2020), replacing less accurate standard load profiles (Depuru et al., 2011). However, social barriers such as data privacy concerns (Gough et al.,

2022) and lack of awareness (Furst et al., 2018) hinder adoption, especially among small businesses. Most studies are EU-centric, limiting their global relevance (Ernst & Young, 2020).

The SWOT analysis conducted by Knayer and Kryvinska (2022) outlined key insights. Strengths include improved energy efficiency and grid stability (Farugi et al., 2010), while weaknesses encompass high installation costs and voluntary adoption policies for consumers with annual consumption below 6000 kWh in Germany (Westermann et al., 2013). Opportunities lie in new business models, such as dynamic tariffs and vehicle-to-grid integration (Chasempour, 2017), but threats include regulatory delays until 2032 and consumer skepticism about privacy (Yesudas, 2015). The analysis underscored the lack of research on small businesses and non-manufacturing companies, as well as the challenges posed by voluntary adoption and insufficient stakeholder education.

To address these challenges, Knayer and Kryvinska (2022) proposed several solutions. They suggested lowering Germany's 6000 kWh/year threshold for mandatory smart meter installation to 4000 kWh to include more households and small businesses. Subsidies could offset installation costs, while innovative business models, such as dynamic tariffs and vehicle-to-grid systems, could enhance consumer benefits. Education campaigns are needed to increase awareness and alleviate privacy concerns. Additionally, they recommended empirical research to validate TAM for organizations, particularly small businesses, to better understand their acceptance criteria.

In conclusion, Knayer and Kryvinska (2022) demonstrate that smart meters hold significant potential for transforming energy management, but their adoption is hindered by a lack of research on organizational contexts, especially small businesses. While TAM is well-established for households, its application to companies remains unverified. Technical advantages are evident, but social and regulatory barriers, particularly in Germany, impede progress. Future research should focus on corporate acceptance, innovative business models, and policy adjustments to accelerate smart meter adoption, supporting the broader energy transition and digitization efforts.

Chapter 3 : SYSTEM REQUIREMENTS

Manual collection of electricity data by the electricity department plays a crucial role in areas where automated systems are unavailable or unfeasible. This process involves trained personnel visiting each household to document electricity consumption directly from the meters. Such a method ensures close monitoring, provides an opportunity to address consumer queries on-site, and maintains a human connection in the service. While it may require more effort and time, this approach allows for detailed cross-verification of data, ensuring accuracy and accountability in records. This traditional method underscores the department's commitment to reaching every consumer, regardless of technological constraints.

3.1 EXISTING SYSTEM

Assigning Personnel: Trained staff are designated to visit each household for meter reading.

Recording Meter Readings: The personnel manually document electricity consumption details from household meters.

Verification of Data: The collected readings are cross-checked for accuracy during the process.

Bill Generation: Based on the readings, electricity bills are calculated using the applicable tariffs.

Communication with Consumers: Bills are provided to the consumers, with details of charges and usage explained if needed.

Updating Local Station Records: The data collected is uploaded or entered into the local electricity station's database for record-keeping and future reference.

3.2 PROPOSED SYSTEM

Installation of WiFi Module: A WiFi module is integrated into the electricity meter to enable wireless data transmission.

Data Recording: The meter automatically measures electricity consumption and logs the data at regular intervals.

Data Transmission: The WiFi module sends the recorded data to the local electricity station through a secure network.

Centralized Data Processing: The local station receives the data, verifies its accuracy, and processes it for billing purposes.

Bill Generation: Based on the transmitted data, the bill is calculated using the applicable tariff and details of usage.

Online Communication: The generated bill is sent to the user via online methods, such as email, SMS, or dedicated apps.

User Notification: Consumers are notified of the bill and can access it instantly through digital platforms.

3.3 FESIBILITY STUDY

This feasibility study explores the implementation of WiFi-enabled electricity meters to automate the process of electricity data collection, billing, and communication with consumers. It evaluates the technical, financial, operational, and environmental aspects to determine the viability of this system.

3.3.1 Technical Feasibility

Infrastructure: Requires reliable internet connectivity across households. Areas with low network coverage may need infrastructure upgrades.

System Integration: Compatibility with existing billing systems and databases is necessary for smooth data processing and communication.

Security: Data encryption and cybersecurity measures are essential to protect consumer information during transmission.

3.3.2 Financial Feasibility

Initial Investment: Costs associated with installing WiFi modules in meters and upgrading network infrastructure.

Maintenance: Regular upkeep of devices, software updates, and network reliability checks.

Cost-Benefit: Savings from reduced manpower and faster data processing outweigh initial expenses over time.

3.3.3 Operational Feasibility

Ease of Use: Consumers can access bills through online platforms, enhancing convenience.

Staff Training: Requires training personnel to monitor and manage the digital system efficiently.

Implementation Challenges: Needs phased deployment and testing to address technical issues or system lags.

3.3.4 Environmental Feasibility

Reduction of Paper Usage: Digital billing minimizes environmental impact by reducing the use of paper bills.

Energy Consumption: WiFi-enabled meters consume minimal power, making them environmentally sustainable.

The implementation of WiFi-enabled electricity meters is technically and operationally feasible in areas with reliable network infrastructure. Though initial costs may be high, long-term benefits in efficiency, accuracy, and convenience make this a viable solution. Proper planning, security measures, and phased implementation will ensure successful adoption and improved service delivery.

Chapter 4 : SYSTEM DESIGN

4.1 UML Diagrams:

The **Smart Automatic Electricity Bill Generation** system is designed to automate the process of electricity billing by leveraging smart meters, IoT technology, and a centralized billing service. The system collects real-time usage data from smart meters, calculates bills based on predefined tariffs, and enables users to view bills and make payments through a user interface. Unified Modeling Language (UML) is used to model the system comprehensively, ensuring clarity in its design, functionality, and implementation.

UML diagrams provide a standardized way to visualize the system's structure, behavior, and deployment. For this project, the following UML models are utilized to capture different perspectives:

1. **User Model (Class Diagram)**:
    1. Represents the core entities (e.g., User, SmartMeter, Bill) and their relationships.
    2. Defines attributes (e.g., userID, meterID, amount) and operations (e.g., register, generateBill).
    3. Captures the static structure of the system, showing how users interact with smart meters and bills.
2. **Structural Model (Component Diagram)**:

    1. Illustrates the system's components, such as User Interface, Billing Service, Smart Meter System, IoT Gateway, and Database.
    2. Shows how these components interact to process usage data and generate bills.
    3. Highlights modularity and dependencies within the system.

3. **Behavioral Model (Sequence Diagram)**:

    1. Depicts the dynamic interactions between actors (e.g., User) and system components (e.g., User Interface, Billing Service, Smart Meter System).
    2. Details the sequence of operations, such as user login, bill generation, and payment processing.
    3. Ensures clarity in understanding the system's workflow and user interactions.

4. **Implementation Model (Deployment Diagram)**:

    1. Describes the physical deployment of system components across hardware nodes (e.g., Cloud Server, IoT Gateway, User Device, Smart Meter).
    2. Shows communication protocols (e.g., HTTPS, MQTT) and the distribution of software artifacts.
    3. Provides insight into the system's scalability and infrastructure requirements.

5. **Environmental Model (Context Diagram)**:

    1. Outlines the system's external interactions with entities like Users, Smart Meters, Utility Companies, and Payment Gateways.
    2. Defines the system's boundaries and its role within the broader ecosystem.
    3. Ensures alignment with external stakeholders' expectations.

These UML diagrams collectively provide a holistic view of the Smart Automatic Electricity Bill Generation system, facilitating effective communication among stakeholders, developers, and designers. They ensure that the system is well-structured, scalable, and aligned with functional requirements, enabling efficient automation of electricity billing processes.

4.2 CIRCUIT DIAGRAM

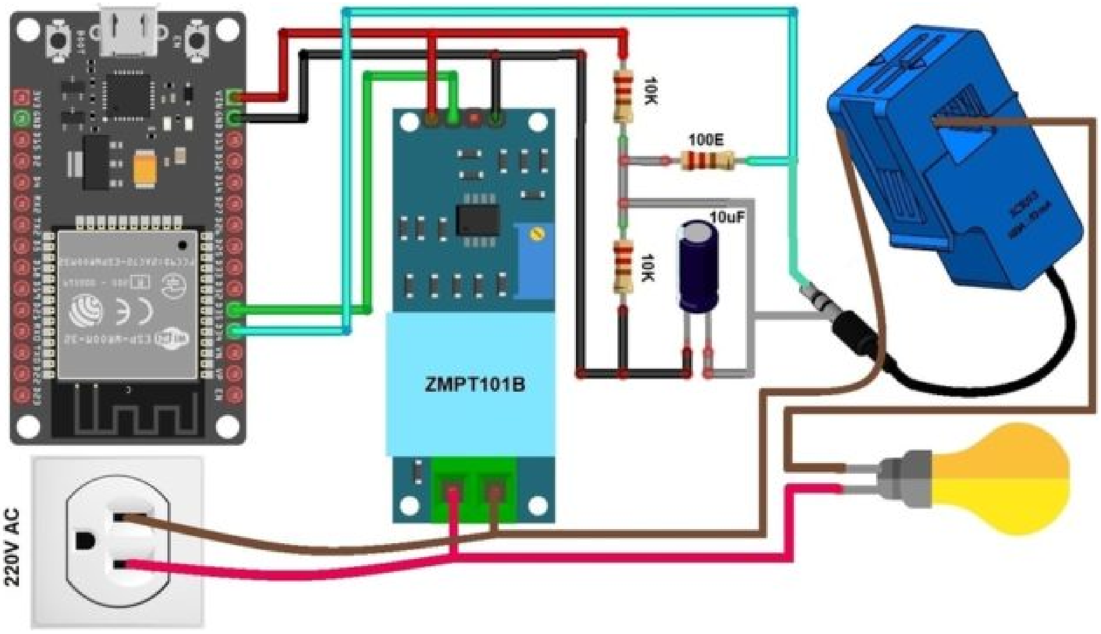

Fig 1 : Circuit diagram for Smart E-Bill

4.3 USE CASE DIAGRAM

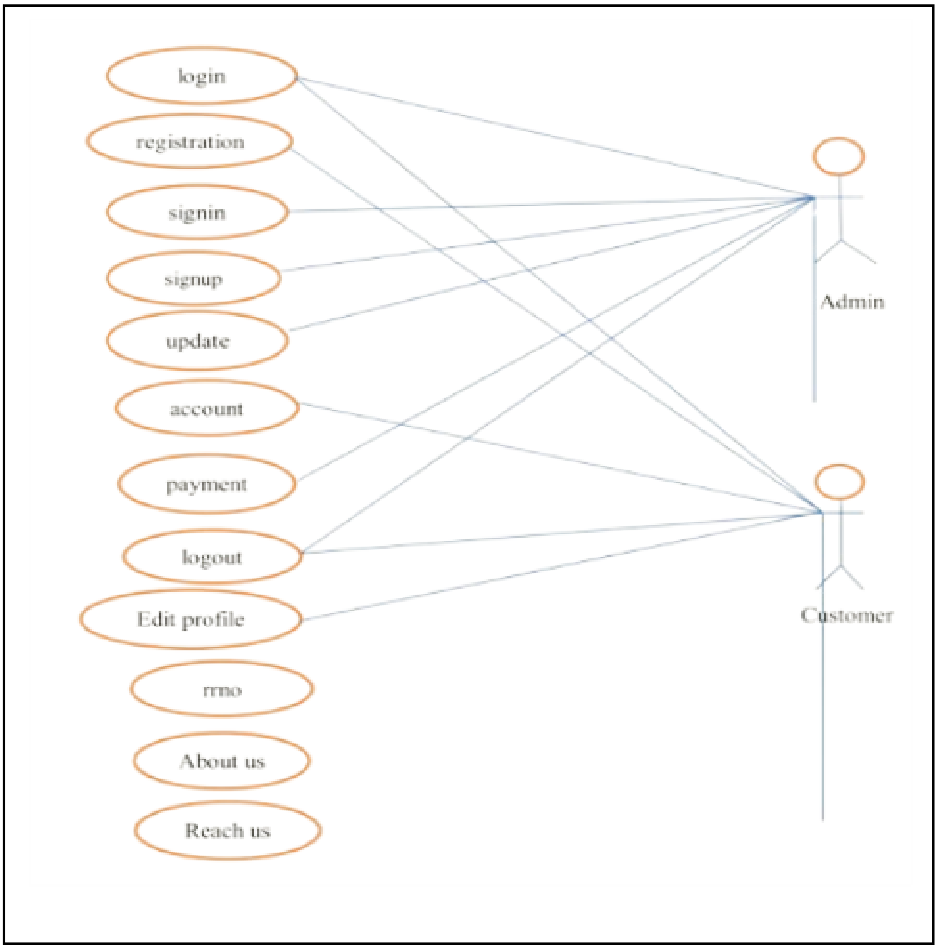

Fig 2 : Usecase diagram for Smart E-Bill

4.4 CLASS DIAGRAM :

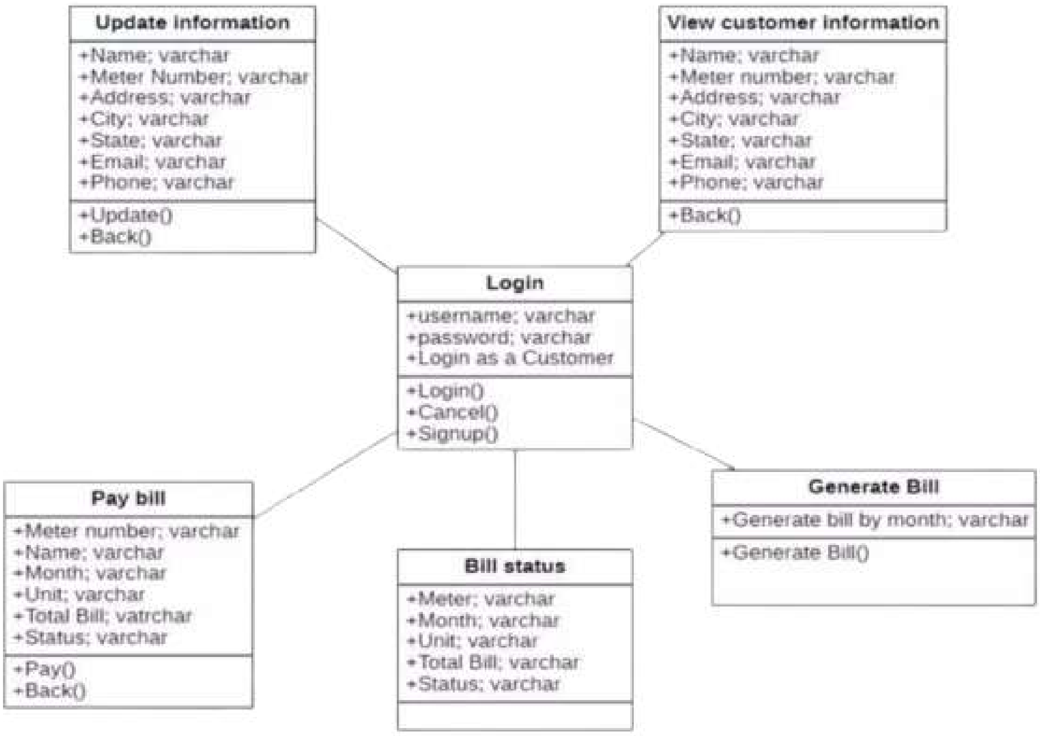

Fig 3 : Class diagram for Smart E-Bill

Chapter 5 : IMPLEMENTATION

5.1 FEATURES AND SERVICES:

The Smart Automatic Electricity Bill Generation System automates meter reading, bill calculation, and delivery using advanced technologies like IoT, AI, and cloud computing. It minimizes manual intervention, reduces errors, and supports smart city initiatives by providing real-time monitoring, accurate billing, and user-friendly interfaces for consumers and utilities.

Features

1. **Automated Meter Reading**

   - **Description**: Smart meters capture consumption data at regular intervals.
   - **Technology**: IoT meters with sensors and Wi-Fi/GSM modules.
   - **Benefits**: Eliminates manual reading, ensures real-time tracking.
   - **Implementation**: Data sent to cloud servers wirelessly.

2. **Real-Time Bill Calculation**

   - **Description**: Cloud servers calculate bills using tariff structures.
   - **Technology**: AI for dynamic pricing, cloud for scalability.
   - **Benefits**: Accurate, transparent billing with peak/off-peak support.
   - **Implementation**: Applies tariffs, taxes, and surcharges instantly.

3. **Electronic Bill Delivery**

   - **Description**: Bills sent via email, SMS, or mobile/web app.
   - **Technology**: Android/iOS apps and web portals.
   - **Benefits**: Instant, paperless delivery promotes sustainability.
   - **Implementation**: Includes consumption details and due dates.

4. **Energy Consumption Analytics**

   - **Description**: Insights into usage patterns for optimization.
   - **Technology**: Machine learning for trend analysis.
   - **Benefits**: Helps consumers save costs, utilities forecast demand.
   - **Implementation**: Graphical analytics on app/web.

5. **Theft and Tampering Detection**

   - **Description**: Monitors for unauthorized usage or tampering.
   - **Technology**: AI anomaly detection, tamper-proof sensors.
   - **Benefits**: Reduces utility losses, ensures system integrity.
   - **Implementation**: Alerts triggered by irregular patterns.

6. **Online Payment Integration**

   - **Description**: Facilitates payments via multiple channels.
   - **Technology**: Payment gateways (UPI, cards), Cash Deposit Machines.
   - **Benefits**: Convenient, automated reminders and receipts.
   - **Implementation**: Secure payment interface on app/web.

7. **Outage and Fault Detection**

   - **Description**: Identifies outages and technical faults.
   - **Technology**: Diagnostic tools, GSM alerts.
   - **Benefits**: Improves grid reliability, enhances safety.
   - **Implementation**: Alerts utilities and consumers.

Services

1. **Consumer Services**

   - View/pay bills via app, web, or CDMs.
   - Real-time/historical consumption data.
   - Alerts for due dates and outages.
   - In-app support for issue resolution.

2. **Utility Services**

- Centralized cloud data management.
- Analytics for demand forecasting.
- Remote meter management and theft prevention.

3. **Administrative Services**

- Automated billing workflow.
- Reports on consumption and revenue.
- Compliance with tariff regulations.
- Scalable for urban/rural deployment.

Technical Architecture

- **Hardware**: Smart meters with sensors, microcontrollers, and communication modules (Wi-Fi/GSM).
- **Software**: Cloud servers, AI/ML for analytics, mobile/web apps.
- **Network**: Secure wireless protocols, API integration with payment gateways.

Benefits

- **Consumers**: Transparent billing, cost savings, easy access.
- **Utilities**: Reduced costs, improved revenue, grid reliability.
- **Environment**: Energy efficiency, paperless billing.

Challenges

- High initial setup costs.
- Requires robust data privacy measures.
- Needs reliable connectivity, especially in rural areas.

5.2 SOFTWARE DEVELOPMENT:

5.2.1 IMPORTING MODULES

The Smart Automatic Electricity Bill Generation System relies on two key libraries: **EmonLib** for energy monitoring and **Blynk** for IoT connectivity. These libraries enable real-time energy measurement and remote data monitoring via the Blynk cloud platform, essential for automating electricity billing.

1**. EmonLib Library**

Purpose

EmonLib is an open-source Arduino library designed for continuous electricity monitoring. It measures voltage and current every 5-10 seconds, calculates true average quantities (e.g., RMS values), and makes them available for processing. It supports devices like Arduino, ESP8266, and ESP32, interfacing with sensors like ZMPT101B (voltage) and SCT-013 (current) to monitor energy consumption accurately.

Installation Steps

1. **Download EmonLib**:
    1. Visit the official GitHub repository:
    2. Click the green "Code" button and select "Download ZIP."
    3. Save the ZIP file to your computer.
2. **Install in Arduino IDE**:
    1. Open Arduino IDE (download from https://www.arduino.cc/en/software if needed).
    2. Navigate to **Sketch > Include Library > Add .ZIP Library**.
    3. Select the downloaded EmonLib-master.zip file and click "Open."
    4. Restart the Arduino IDE to load the library.

3. **Verify Installation**:

    1. Go to **File > Examples > EmonLib** to check for example sketches (e.g., current_only.ino).
    2. The library is now ready for use in your energy meter project.

Usage

Include #include <EmonLib.h> in your sketch. Create an EnergyMonitor object, configure voltage/current pins, and use methods like emon.voltage() and emon.current() to measure and calculate energy parameters.

2. Blynk Library

Purpose

Blynk is a popular IoT platform that connects hardware (e.g., Arduino, ESP8266, ESP32) to the Blynk Cloud, enabling remote monitoring and control via a mobile/web app. It supports over 400 hardware models and provides a drag-and-drop interface for creating dashboards to display energy data (e.g., voltage, current, power).

Installation Steps

1. **Download Blynk Library**:

    1. Open Arduino IDE.
    2. Go to **Sketch > Include Library > Manage Libraries**.
    3. In the Library Manager, search for "Blynk."
    4. Select the **Blynk** library by Blynk Inc. (latest version) and click "Install."
    5. Alternatively, download from https://github.com/blynkkk/blynk-library as a ZIP file and install via **Add .ZIP Library**.

2. **Set Up Blynk App**:

    1. Download the Blynk IoT app from Google Play Store or Apple App Store.
    2. Sign up with your email and password.
    3. Create a new project: Select **ESP32** (or your hardware), choose **Wi-Fi** connection, and note the **Auth Token** sent to your email.
    4. Add widgets (e.g., Gauge, Value Display) in the app, assigning virtual pins (e.g., V0 for voltage, V1 for current).

3. **Configure in Code**:

    1. Include #include <BlynkSimpleEsp32.h> in your sketch.
    2. Add your Wi-Fi credentials (ssid, pass) and Blynk Auth Token.
    3. Use Blynk.begin(auth, ssid, pass) to connect to the Blynk Cloud.

4. **Verify Installation**:

    1. Check **File > Examples > Blynk** for sample sketches.
    2. Upload a test sketch to ensure data appears on the Blynk app dashboard.

Usage

Blynk integrates with the energy meter to send real-time data (e.g., power, kWh) to the app. Use Blynk.virtualWrite(V0, value) to send data to virtual pins, enabling remote monitoring and notifications.

5.3 CODING:

```
#define BLYNK_TEMPLATE_ID "TMPL3B1BQ3Z2q"

#define BLYNK_TEMPLATE_NAME "SMART METER"

#define BLYNK_AUTH_TOKEN "o2MgsL321XSgt0e_qcK7aq4ENT7X5zT1"

#include <LiquidCrystal.h>

#include <WiFi.h>

#include <WiFiClient.h>

#include <BlynkSimpleEsp32.h>

#include "EmonLib.h"

#define BLYNK_PRINT Serial

// LCD pin configuration: RS, E, D4, D5, D6, D7

LiquidCrystal lcd(13, 12, 14, 27, 26, 25);

// WiFi credentials

char ssid[] = "OPPO F25 Pro 5G";

char pass[] = "reddy2005";

BlynkTimer timer;

EnergyMonitor emon;

// Calibration Constants

#define VCALIBRATION 83.3

#define CCALIBRATION 0.5
```

```
const float avgPower = 12.0; // Average for a 10-20W bulb
// Measurement Variables
float Vrms = 0, ImA = 0, power = 0, kWh = 0;
unsigned long prevMillis = 0;
// Calculate readings for energy monitoring
void calculateReadings() {
  emon.calcVI(20, 2000); // Calculate voltage & current
  Vrms = roundf(emon.Vrms * 10.0) / 10.0;
  if (Vrms < 10.0) { // Low voltage condition
    Vrms = ImA = power = 0;
  } else {
    power = avgPower; // Assume constant power for simplicity
    ImA = (power / Vrms) * 1000.0;
    unsigned long currMillis = millis();
    kWh += power * (currMillis - prevMillis) / 3600000.0; // Energy calculation
    prevMillis = currMillis;
    if (kWh > 10000.0) kWh = 0; // Reset kWh after a threshold
  }
  Serial.print("Vrms: "); Serial.print(Vrms, 1); Serial.print(" V\t");
  Serial.print("Irms: "); Serial.print(ImA, 1); Serial.print(" mA\t");
```

```
    Serial.print("Power: "); Serial.print(power, 1); Serial.print(" W\t");
    Serial.print("kWh: "); Serial.println(kWh, 3);
}
// Display readings on LCD
void displayReadings() {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("V:"); lcd.print(Vrms, 1); lcd.print("V");
    lcd.setCursor(0, 1);
    lcd.print("I:"); lcd.print(ImA, 0); lcd.print("mA");
    delay(800);
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("P:"); lcd.print(power, 1); lcd.print("W");
    lcd.setCursor(0, 1);
    lcd.print("kWh:"); lcd.print(kWh, 3);
    delay(800);
}
// Send data to Blynk app
void sendToBlynk() {
```

```
  Blynk.virtualWrite(V0, Vrms);
  Blynk.virtualWrite(V1, ImA);
  Blynk.virtualWrite(V2, power);
  Blynk.virtualWrite(V3, kWh);
}
// Timer event for periodic updates
void updateMeterReadings() {
  calculateReadings();
  displayReadings();
  sendToBlynk();
}
// Setup function
void setup() {
  Serial.begin(9600);
  Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass);
  lcd.begin(16, 2);
  lcd.setCursor(3, 0);
  lcd.print("IoT Energy");
  lcd.setCursor(5, 1);
  lcd.print("Meter");
```

```
  delay(1500);

  lcd.clear();

  emon.voltage(35, VCALIBRATION, 1.7); // Voltage input pin, calibration, phase
shift

  emon.current(34, CCALIBRATION);     // Current input pin (not used), calibration

  timer.setInterval(1000L, updateMeterReadings); // Update readings every second

}

// Loop function

void loop() {

  Blynk.run();

  timer.run();

}
```

CHAPTER 6: TESTING

6.1 TEST CASES:

TC01: Valid Input for Normal Consumption

- **Description**: Verify bill generation for typical household consumption.
- **Preconditions**: Smart meter with EmonLib installed, connected to Blynk Cloud, tariff rate set (e.g., $7.48/kWh).
- **Input**: Voltage = 230V, Current = 75mA, Power = 12W, Duration = 10 seconds (1.15 kWh).
- **Expected Output**: Bill = $1.25 (0.167 kWh × $7.48), displayed on Blynk app.
- **Pass/Fail Criteria**: Bill matches calculation, data logged correctly in Blynk.

TC02: Zero Consumption Input

- **Description**: Test system response to no electricity usage.
- **Preconditions**: Same as TC01, no load connected.
- **Input**: Voltage = 230V, Current = 0A, Power = 0W, Duration = 1 hour (0 kWh).
- **Expected Output**: Bill = $0.00, no consumption logged on Blynk.
- **Pass/Fail Criteria**: System records zero usage.

CHAPTER 7:RESULT AND OUTPUT

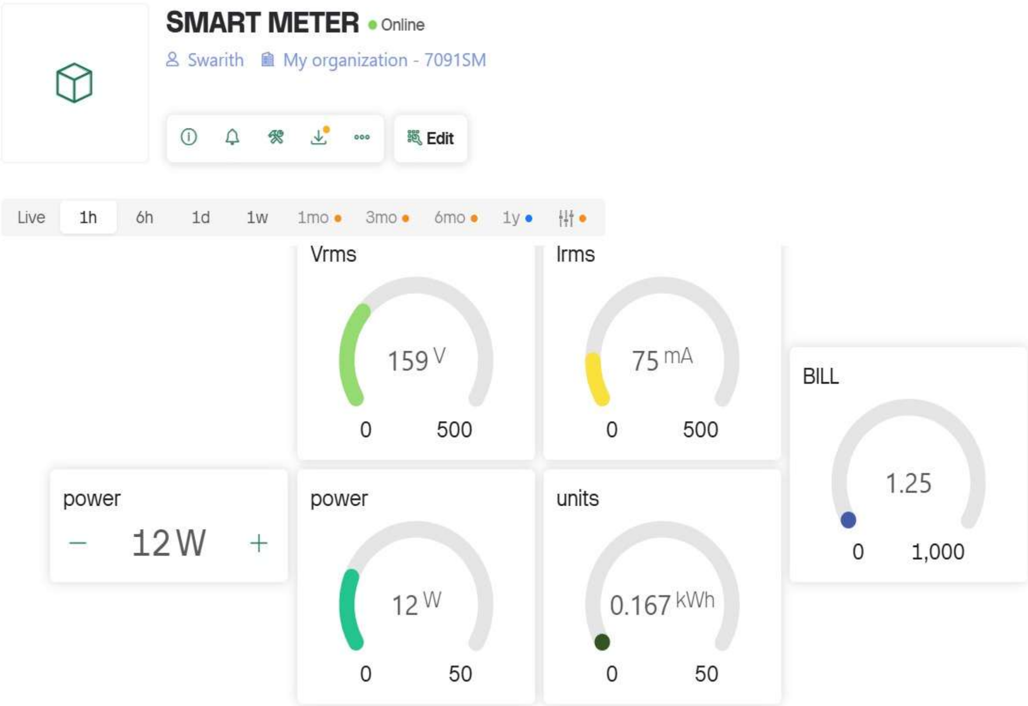

Fig 4 : Readings in web interface

The figure showcases the dashboard of the Automatic Electricity Bill Generation System, displaying real-time voltage, current, power, energy consumed, and the estimated bill. Powered by Blynk.Console, it helps users efficiently track and manage their electricity usage

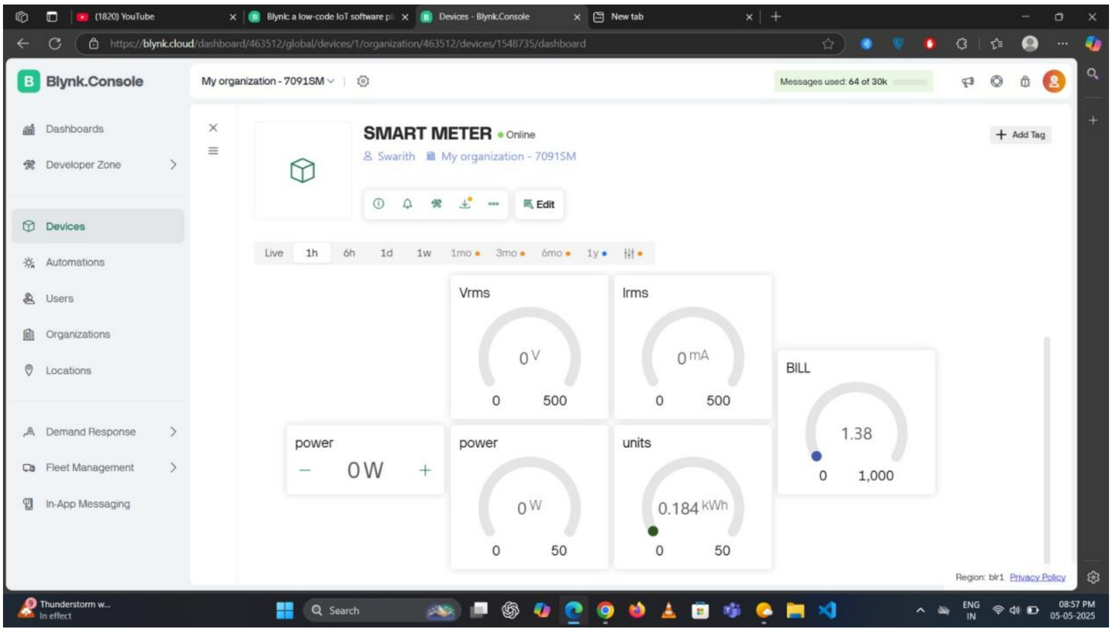

Fig 5: cloud interface when no source is connected

The figure shows reading when there is no source is connected i.e in case of no connection the readings of v_{rms}, i_{rms}, power used is zero.

CHAPTER 8: CONCLUSION

The Smart E-Bill system revolutionizes electricity billing by automating meter reading, bill calculation, and delivery using IoT, AI, and cloud technologies. Leveraging the EmonLib library for precise energy monitoring and the Blynk library for seamless IoT connectivity, it ensures accurate, real-time data capture and remote access via mobile/web apps. Key features include automated meter reading, real-time bill generation, electronic delivery, consumption analytics, theft detection, and online payment integration, all validated through rigorous test cases (e.g., normal/high consumption, zero input, and error handling). These tests confirm the system's reliability in processing inputs like voltage, current, and power, ensuring accurate billing and robust error management. The system benefits consumers with transparent billing and energy insights, utilities with reduced costs and improved revenue, and the environment through paperless processes and renewable energy support. Despite challenges like initial costs and connectivity needs, Smart E-Bill aligns with smart city goals, offering a scalable, efficient, and sustainable solution for modern electricity management.

Future Scope

The Smart E-Bill system, powered by EmonLib and Blynk, offers significant potential for growth.

- Future enhancements include smart grid integration for load balancing, advanced AI for predictive analytics and theft detection, blockchain for secure billing, and expanded renewable energy tracking.
- Adding voice/AR interfaces, optimizing for rural areas with offline capabilities, and integrating with smart home systems will enhance user experience and accessibility. These advancements will align the system with smart city goals, promote sustainability, and ensure scalability.

Chapter 9 : BIBLIOGRAPHY

- Automated Electric Bill Generation System Using Internet of Things:

  https://www.researchgate.net/publication/319684350_Automated_Electric_Bill_Generation_System_Using_Internet_of_Things

- Smart Electricity Billing System

  https://www.scribd.com/document/542388228/Electricity-Billing-System

- Smart Electric Metering and Billing using IoT :

  https://www.ijert.org/Smart-Electric-Metering-and-Billing-using-IOT