



# **KONGU ENGINEERING COLLEGE**

(Autonomous)

Perundurai, Erode – 638 060

**DEPARTMENT OF ELECTRONICS AND INSTRUMENTATION ENGINEERING**



## **Electricity billing system**

**for**

**JAVA PROGRAMMING (22ITC31)**

**Submitted by**

**DEEPIKA S (23EIR016)**

**DEEPTHI AV (23EIR017)**

**DEVAKI M (23EIR018)**

**KONGU ENGINEERING COLLEGE**

(Autonomous)

Perundurai, Erode – 638 060

**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING**



## **BONAFIDE CERTIFICATE**

Name & Roll No.: **DEEPIKA S (23EIR016)**

**DEEPTHI AV (23EIR017)**

**DEVAKI M (23EIR018)**

Course Code : **22ITC31**

Course Name : **JAVA PROGRAMMING**

Semester : **III**

Certified that this is a bonafide record of work for application project done by the above students for **22ITC31-JAVA PROGRAMMING** during the academic year **2024-2025**.

Submitted for the Viva Voce Examination held on \_\_\_\_\_

Faculty In-Charge

Year In-charge

### **TABLE OF CONTENTS**

<b>CHAPTER No.</b>	<b>TITLE</b>	<b>PAGE No.</b>
<b>1.</b>	<b>Abstract</b>	<b>04</b>
<b>2</b>	<b>Problem Statement</b>	<b>05</b>
<b>3.</b>	<b>Methodology</b>	<b>06</b>

<b>4.</b>	<b>Implementation</b>	<b>08</b>
<b>5.</b>	<b>Results and Discussion</b>	<b>12</b>
<b>6.</b>	<b>Conclusion</b>	<b>13</b>
<b>7.</b>	<b>Sample Coding</b>	<b>14</b>

## **ABSTRACT**

Electricity consumers are often faced with the problem of inaccuracy and delay in monthly billing due to some drawbacks. Thus, it is essential to have an efficient system for such purposes via electronic platform with consideration to proximity. The proposed system automates the conventional process of paying electricity bill by visiting the Electricity Board which is tiresome and time consuming. It is also designed to automate the electricity bill calculation and payment for user convenience. The system is developed with JSP SERVLET as the base programming language which can be used to develop websites, web applications and web services. The MYSQL is a relational database management system based on Structured Query Language (SQL) which is used for the purpose of web database. The system would be having two logins: the administrative and user login. The administrator can view the user's account details and can add the customer's information of consuming units of energy of the current month in their account. The Admin must feed the system with the electricity usage data into respective user's account. The system then calculates the electricity bill for every user and updates the information into their account every month. Users can then view their electricity bill and pay before the month end.

# **PROBLEM STATEMENT**

We, the owners of our project, respect all customers and make them happy with our service.

The main aim of our project is to satisfy customer by saving their time by payment process, maintaining records, and allowing the customer to view his/her records permitting them to update their details.

The firm handles all the work manually, which is very tedious and mismatched.

The objectives of our project are as follows:

- To keep the information of customer.
- To keep the information of consuming unit energy of current month.
- To keep the information of consuming unit energy of previous month.
- To calculate the units consumed every month regularly.
- To generate the bills adding penalty and rent.
- To save the time by implementing payment process online.

The manual system is suffering from a series of drawbacks. Since whole of the bills is to be maintained with hands the process of keeping and maintaining the information is very tedious and lengthy to customer. It is very time consuming and laborious process because, staff need to be visited the customers place every month to give the bills and to receive the payments. For this reason, we have provided features Present system is partially automated(computerized), existing system is quite laborious as one must enter same information at different places.

## **METHODOLOGY**

### **Existing and Proposed System**

The conventional system of electricity billing is not so effective; one staff must visit each customer's house to note the meter readings and collect the data. Then, another staff must compute the consumed units and calculate the money to be paid. Again, the bills prepared are to be delivered to customers. Finally, individual customer must go to electricity office to pay their dues. Hence, the conventional electricity billing system is uneconomical, requires many staffs to do simple jobs and is a lengthy process overall. In order to solve this lengthy process of billing, a web based computerized system is essential. This proposed electricity billing system project overcomes all these drawbacks with the features. It is beneficial to both consumers and the company which provides electricity.

With the new system, there is reduction in the number of staffs to be employed by the company. The working speed and performance of the software is faster with high performance which saves time. Furthermore, there is very little chance of miscalculation and being corrupted by the staffs.

## **Feasibility Study:**

Feasibility study is the phase in which the analyst checks that the candidate system is feasible for the organization or not. This entails identification, description & evaluation of the system. Feasibility study is done to select the best system that meets the performance requirement.

If the feasibility study is to serve as a decision document, it must answer key questions.

1. Is there a new and better way to do the job that will benefit the user?
2. What are the costs and savings of the alternatives?
3. What is recommended?

The most successful system projects are not necessarily the biggest or most visible in the business but rather those truly meet user's expectations.

## **Feasibility considerations**

Three key considerations are involved in the feasibility study. They are as follows:-

### **Economic Feasibility:**

Economic analysis is the most frequently used method for evaluating the effectiveness of the candidate system.

We analyse the candidate system (computerized system) is feasible as than the manual system because it saves the money, time and manpower. It also feasible according to cost benefits analysis

### **Technical Feasibility:**

Technical feasibility centers around the technology used. It means the candidate system is technically feasible i.e. it don't have any technical fault and work properly in the given environment. Our system is technically feasible; it is providing us required output.

### **Behavioral Feasibility:**

Behavioral feasibility is the analysis of behavior of the candidate system. In this we analyse that the candidate system is working properly or not. If working than it communicating proper with the environment or not. All this matters are analysed and a good candidate system is prepared. Due to the change of system what is the change in behaviour of the users, this factors are also analysed.

## **SYSTEM DEVELOPMENT ENVIRONMENT**

System development environment shows the hardware and software requirement, which is necessary for developing the software. Necessary software and hardware requirement, which are necessary for making this software are as follows:

### **Hardware Requirements:**

- Hardware specification: Intel Pentium Processor
- 32 MB RAM or Higher
- 1.2 GB Hard Disk or Greater
- Video Display Unit
- Keyboard

### **Software Requirements:**

- Operating System: Windows 10
- Software: Microsoft SQL Server

- Front End: Jsp Servlet
- Back End: Database – Mysql Business Logic - Java

## **IMPLEMENTATION**

### **Implementation of operations**

- **Adding customer:** Here admin can add new customer list who started using electricity bill system.
- **Searching deposit details:** Here admin can search according to meter number and month to view deposit details.
- **Viewing Details:** Here admin and user can view customer details and about details.
- **Updating customer:** Here customer can update their details by using meter no of the customer.
- **Delete customer:** Here admin can delete details based on meter number.

### **Implementation of SQL statements**

#### **Insert statement:**

- The INSERT INTO statement is used to insert new records in a table.
- The INSERT INTO syntax would be as follows: INSERT INTO table\_name VALUES (value1, value2, value3, ...).
- The following SQL statement insert's a new record in the “customer” table: Insert into customer VALUES (“sai”, “12345”, “btm”, “Bangalore”, “Karnataka”, “sai@gmail.com”, “9876543333”).

#### **Update statement:**

- An SQL UPDATE statement changes the data of one or more records in a table. Either all the rows can be updated, or a subset may be chosen using a condition.<sup>22</sup>

- The UPDATE syntax would be as follows: UPDATE table\_name SET column\_name =value, column\_name=value... [WHERE condition].

- The following SQL statement update's a new record in the “customer” table: UPDATE TABLE customer SET email=

su@gmail.com WHERE meter\_no =”12345”. **Delete statement:**

- The DELETE statement is used to delete existing records in a table. • The DELETE syntax would be as follows: DELETE FROM table\_name WHERE condition.

- The following SQL statement delete's a record in the “customer” table: delete from customer where meter\_no=12345. **Create**

### **statement:**

- The CREATE TABLE Statement is used to create tables to store data. Integrity Constraints like primary key, unique key, foreign key can be defined for the columns while creating the table.

- The syntax would be as follows: CREATETABLE table\_name (column1datatype, column2datatype, column3 datatype, columnN datatype, PRIMARY KEY (one or more columns)). > The following SQL statement creates a table “customer” table: create table customer (name varchar (30), meter no varchar (20) primary key, address varchar (50), city varchar (20), state varchar (30), email varchar (30), phone varchar (30));

>The following SQL statement creates a table “login” table: create table login (meter no varchar (30), username varchar (30), password varchar (30), user varchar (30), question varchar (40), answer varchar (30));

> The following SQL statement creates a table “bill” table: create table bill (meter no varchar (20), foreign key(meter no) references customer(meter no) on delete cascade, month varchar (20), units int (20), total bill int (20), status varchar (40));

## **Algorithm of implementation**

### **Explanation of Algorithm:**



✓

Start system

✓

Enter login name and password

✓

On clicking the login button

✓

Connect to database

✓

Query database to know whether user credentials are correct

✓

If not, deny access and return login page with an error message

✓

If correct, check if credentials for administrator

✓

If yes, allow login

✓

Set admin session, re-direct administrator to admin login page

✓

If no, allow login set user session

✓

Re-direct user to user home page

## **Algorithm of admin:**

### **Login:**

➤

This program will allow the admin to enter the username and password.

➤

If the entered credentials are correct, then the login will be successful otherwise need to be signup.

➤

If admin forgets password, it can be retrieved by giving username and answer for security question.

After successful login the admin will be redirected to admin portal page where he/she can do following activities.

### **NewCustomer:**

- 

This program will allow the admin to enter the customer details and automatically generates unique meter number.

- 

If customer name, address, city, state, email and phone number is entered, insert the values into customer<sup>24</sup>

else print error while

next=true enter the

meter info details else

print meter

info error

Submit the details of customer that has been entered by clicking onto next button.

- 

If we need to cancel the particulars that has been entered click onto cancel option.

### **CustomerDetails:**

- This program will allow the admin to view customer details.

- If we need to print the particulars that has been viewed click onto print option.

### **CalculateBill:**

- This program will allow the admin to calculate total bill when units consumed are inserted where meter no and month is selected. insert the values into bill else print error

Submit the details of tax that has been entered by clicking onto submit button.

- If we need to cancel the particulars that has been entered click onto cancel option.

**DeleteBill:**

- This Program will allow the admin to delete the customer info when meter no is selected.
- If we need to delete the particulars that has been saved click onto delete option.<sup>25</sup>
- If we need to cancel the particulars that has been entered click onto back option.

**About:**

- This program will allow the admin to view details of the project in short.
- If we need to exit the particulars that has been viewed click onto exit Option.

## **Results and Discussion**

The conventional system of electricity billing is not so effective; one staff must visit each customer's house to note the meter readings and collect the data. Then, another staff must compute the consumed units and calculate the money to be paid. Again, the bills prepared are to be delivered to customers. Finally, individual customer must go to electricity office to pay their dues. Hence, the conventional electricity billing system is uneconomical, requires many staffs to do simple jobs and is a lengthy process overall. In order to solve this lengthy process of billing, a web based computerized system is essential. This proposed electricity billing system project overcomes all these drawbacks with the features. It is beneficial to both consumers and the company which provides electricity.

With the new system, there is reduction in the number of staffs to be employed by the company. The working speed and performance of the software is faster

with high performance which saves time. Furthermore, there is very little chance of miscalculation and being corrupted by the staffs.

## **CONCLUSION**

Usability testing was part of the post implementation review and performance evaluation for the Electricity Online Bill Payment System, in order to ensure that the intended users of the newly developed system can carry out the intended task effectively using real data so as to ascertain the acceptance of the system and operational efficiency. It caters for consumers' bills and also enables the administrator to generate monthly reports. It is possible for the administrator to know the consumers have made payment in respect of their bills for the current month, thereby improving the billing accuracy, reduce the consumption and workload on the Electricity Board employees or designated staff., increase the velocity of electricity distribution, connection, tariff scheduling and eliminates variation in bills based on market demand. The conceptual framework allows necessary adjustments and enhancement maintenance to integrate future demands according to the technological or environmental changes with time. It manages the consumers' data and validates their input with immediate notification centralized in Electricity Board offices across the nation.

## **SAMPLE CODING**

```
import java.util.Scanner; public class ElectricityBillingSystem { private  
String[] userNames = {"Alice", "Bob", "Charlie", "David", "Eve"};  
  
    private String[] phoneNumbers = {"1234567890", "2345678901", "3456789012",  
"4567890123", "5678901234"}; private String[] ebNumbers = {"EB123", "EB234",  
"EB345", "EB456", "EB567"}; private double[] meterReadings = {100.0, 150.0,  
200.0, 250.0, 300.0}; private double[] availableKW = {50.0, 60.0, 70.0, 80.0,  
90.0}; public ElectricityBillingSystem() {
```

```

    }

    public int getUserIndex(String input) {

        for (int i = 0; i < userNames.length; i++) {
            if (input.equals(userNames[i]) || input.equals(phoneNumbers[i]) ||
input.equals(ebNumbers[i])) {

                return i;
            }
        } return -
1; }
    public void calculateBill(int index) {
        if (index != -1) { double

            ratePerKW = 5.0; double

            totalAmount =

            meterReadings[index] *

            ratePerKW;

            System.out.println("Postpaid User: Total bill based on the meter reading is: Rs. " +
totalAmount);

        } else {

            System.out.println("Error: User not found!");
        }
    } public void rechargePrepaid(int index, double amount)
    { if (index != -1) { double convertedKW = amount;

        availableKW[index] += convertedKW;

        System.out.println("Prepaid User: Recharge successful! You have now " +
availableKW[index] + " KW available.");

    } else {

```

```

        System.out.println("Error: User not found!");
    }
} public static void main(String[] args)
{
    Scanner scanner = new Scanner(System.in);

    ElectricityBillingSystem system = new ElectricityBillingSystem();

    System.out.print("Enter your user name, phone number, or EB number: ");

    String input = scanner.nextLine(); int userIndex =
    system.getUserIndex(input); if (userIndex != -1) {

        System.out.println("User validated successfully!");
        System.out.print("Choose your plan (1 for Prepaid, 2 for Postpaid):

        "); int planChoice = scanner.nextInt(); scanner.nextLine(); if

        (planChoice == 1) {

            // Prepaid option
            System.out.print("Enter the amount to recharge: ");

            double amount = scanner.nextDouble();

            system.rechargePrepaid(userIndex, amount);

        } else if (planChoice == 2) {

            // Postpaid option
            System.out.println("Your current meter reading: " +
system.meterReadings[userIndex]);

            System.out.print("Enter the meter reading: ");

            double meterReading = scanner.nextDouble();

            system.meterReadings[userIndex] = meterReading;

            system.calculateBill(userIndex);

```

```
    } else {  
        System.out.println("Invalid plan choice!");  
    }  
} else {  
    System.out.println("Error: User not found! Please check the details.");  
}  
scanner.close();  
}  
}
```

**Faculty Incharge    Academic Coordinator HOD**