**Social Media Application**

***A Dissertation submitted in partial fulfillment of the requirements for the award of degree of***

**MASTER OF COMPUTER APPLICATIONS**

**of**

**Chhattisgarh Swami Vivekanand Technical University By**

**NITESH KUMAR OJHA (501402122029)**

**Under the Guidance of Ms. DEEPTI CHANDRA**

**Department of Computer Applications, Shri Shankaracharya Technical Campus.**

**Junwani, Bhilai Session: 2023-2024**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | 1.00 | Applicable for  AY 2022-23 Onwards |
| Chairman (AC) | Chairman (BoS) | Date of Release | Version |

SSTC-MCA

**Declaration**

I, **Nitesh Kumar Ojha,** student of 3rd Semester MCA, **Shri Shankaracharya Technical Campus,** bearing Enrolment Number 501402122029 hereby declare that the project entitled **Electricity Billing System** has been carried out by me under the supervision of **External Guide /(or Guide) Ms. Deepti Chandra**, submitted in partial fulfillment of the requirements for the award of the Degree of Master of Computer Applications by the Chhattisgarh Swami Vivekanand Technical University during the academic year 2023-24. This report has not been submitted to any other Organization/University for any award of degree.

**Signature: Name:**

**University Roll No.:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | 1.00 | Applicable for  AY 2020-21 Onwards |
| Chairman (AC) | Chairman (BoS) | Date of Release | Version |

SSTC-MCA

**FORWARDING CERTIFICATE**

This is to Certify that **Nitesh Kumar Ojha**, a bonafide student of Master of Computer Application (M.C.A) at **SHRI SHANKARACHARYA TECHNICAL CAMPUS**, Has carried out his project work as mentioned in this project entitled **“Nitesh Kumar Ojha” ,** during his/her third semester of studies in M.C.A as a part of a curriculum for obtaining the degree of M.C.A from Chhattisgarh Swami Vivekanand Technical University, Bhilai (C.G) to which the institute is affiliated.

This Certificate Issued by the undersigned does not cover any responsibility regarding the statements made and work carried out by the concerned student.

The current dissertation is hereby being forwarded for evaluation for the purpose for which it has been submitted.

**Project Coordinator (M.C.A) Head of Department (M.C.A)**

Shri Shankaracharya Technical Campus Shri Shankaracharya Technical Campus

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | 1.00 | Applicable for  AY 2022-23 Onwards |
| Chairman (AC) | Chairman (BoS) | Date of Release | Version |

SSTC-MCA

### CERTIFICATE OF APPROVAL

This is to Certify that the project the entitled “**Electricity Billing System”,** carried out by “**Nitesh Kumar Ojha”** a student of third semester, M.C.A. at **Shri Shankaracharya Technical Campus,** is hereby approved after proper examination and evaluation as a creditable work for the partial fulfillment of the requirement for awarding the degree of Master of Computer Applications (M.C.A) from Chhattisgarh Swami Vivekanand Technical University, Bhilai (C.G)

**(Internal Examiner) (External Examiner)**

**Name: Name:**

**Designation: Designation:**

**College Name: College Name:**

**Date: Date:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | 1.00 | Applicable for  AY 2022-23 Onwards |
| Chairman (AC) | Chairman (BoS) | Date of Release | Version |

SSTC-MCA

### ACKNOWLEDGEMENT

I have great pleasure in the submission of this project report entitled **Electricity Billing System** in partial fulfillment of the degree of **Master of Computer Applications**. While submitting this Project report, I take this opportunity to thank those directly or indirectly related to project work.

I would like to thank my guide **Ms. Deepti Chandra** who has provided the opportunity and organizing project for me. Without his active cooperation and guidance, it would have become very difficult to complete task in time.

I would like to express sincere thanks and gratitude to P.B. Deshmukh (Director), **Principal**, Dr. Smita Selot

**Head of Department**, (Computer Application).

While Submission of the project, I also like to thanks to **Ms. Deepti Chandra** and the staff of **Shri Shankaracharya Technical Campus** for their continuous help and guidance throughout the course of the project.

Acknowledgement is due to our parents, family members, friends and all those persons who have helped us directly or indirectly in the successful completion of the project work.

**Name of the Student Roll No: Enrollment:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | 1.00 | Applicable for  AY 2023-24 Onwards |
| Chairman (AC) | Chairman (BoS) | Date of Release | Version |

SSTC-MCA

# CONTENTS

## INTRODUCTION

**Project Description**

## SYSTEM STUDY

#### Existing And Proposed System Feasibility Study

**Tools and Technologies Used Hardware and Software Requirements**

## SOFTWARE REQUIREMENTS SPECIFICATION

#### Users

**Functional Requirements**

**Non-Functional Requirements**

## SYSTEM DESIGN

#### System Perspective Context Diagram Use Case Diagram Sequence Diagram

**Collaboration Diagram Activity Diagram DataBase Design**

## IMPLEMENTATION

1. SCREENSHOTS

## SOFTWARE TESTING

1. CONCLUSION

## FUTURE ENHANCEMENTS Appendix A BIBLIOGRAPHY

Chapter-1

#### Introduction

* + 1. **Project Description**

**Electricity Billing System Project in Java** is a software-based application developed in Java programming language. The project aims at serving the department of electricity by computerizing the billing system. It mainly focuses on the calculation of Units consumed during the specified time and the money to be paid to electricity offices. This computerized system will make the overall billing system easy, accessible, comfortable and effective for consumers.

To design the billing system more service oriented and simple, the following features have been implemented in the project. The application has high speed of performance with accuracy and efficiency.

The software provides facility of data sharing, it does not require any staff as in the conventional system. Once it is installed on the system only the meter readings are to be given by the admin where customers can view all details, it has the provision of security restriction.

# Chapter-2

* 1. **SYSTEM STUDY**

#### Existing And Proposed System

The proposed system is developed based on the client server architecture, a

request-response paradigm and is implemented with the help of advanced java using the tomcat web container. The employees can maintain and do the transactions online.The application starts by asking for username and password which provides authentication. Thissystem provides high security where the unauthorized users cannot access the data.Later we have different options for the employee like

* Customer management
* Admin management
* Meter details
* Billing and Accounting
* Reports

# FEASIBILITY STUDY

Feasibilities are studied from an Economical, technical, operational and legal point of view and hence found no obstacles to continue with our proposed project to be developed. So, the feasibility studies are undergone as follows:

**Economic Feasibility:** More commonly known as Cost/Benefit Analysis. The procedure is to determine the benefits and savings that are expected from a candidate system and compare them with costs.If the benefits outweigh costs, then a decision is made to design and implement the system.

**Technical Feasibility:** Technical feasibility centers on the existing computer system(hardware, software, etc.) and to what extent it can support the proposed addition. If the budget is a serious constraint, then the project is judged not feasible. In our case this does not become an obstacle.

# TOOLS AND TECHNOLOGIES USED

#### Eclipse IDE

**Eclipse** is an [integrated development environment](https://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) used in [computer programming](https://en.wikipedia.org/wiki/Computer_programming). It contains a base [workspace](https://en.wikipedia.org/wiki/Workspace) and an extensible [plug-in](https://en.wikipedia.org/wiki/Plug-in_(computing)) system for customizing the environment. It is the second-most-popular IDE for [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) development, and, until 2016, was the most popular.Eclipse is written mostly in Java and its primary use is for developing Java applications.

In November 2001, a consortium was formed with a board of stewards to further the development of Eclipse as [open-source software](https://en.wikipedia.org/wiki/Open-source_software).

Eclipse 3.0 (released on 21 June 2004) selected the [OSGi](https://en.wikipedia.org/wiki/OSGi) Service Platform specifications as the runtime architecture.

Eclipse supports development for [Tomcat](https://en.wikipedia.org/wiki/Apache_Tomcat), [GlassFish](https://en.wikipedia.org/wiki/GlassFish) and many other servers and is often capable of installing the required server (for development) directly from the IDE. It supports

remote debugging, allowing a user to watch variables and step through the code of an application that is running on the attached server.

#### System requirements

###### Windows macOS Linux

**OS Version** 64 Bit Microsoft

Windows 8 or later

macOS 10.13 or later

Any Linux distribution that supports Gnome, KDE, or Unity DE

**RAM** 2 GB RAM minimum, 8 GB RAM recommended

**Disk space** 2.5 GB and another 1 GB for caches minimum, [solid-state drive](https://en.wikipedia.org/wiki/Solid-state_drive) with at least 5 GB of free space recommended

**JDK Version** Add support for Java 17

**JRE Version** JRE 8 is bundled.

###### Screen

**resolution**

1024×768 minimum screen resolution. 1920×1080 is a recommended screen resolution.



# Chapter-3

* 1. **SOFTWARE REQUIREMENTS SPECIFICATION**

**Hardware Requirements:-**

* + - Processors: Pentium 4 or onwards.
    - Hard Disc : 80 Gb.
    - Monitor : 15’’ Color Monitor.
    - Mouse.
    - Keyboard.

##### SOFTWARE REQUIREMENTS:-

Operating System :- Windows 10 or onwards or Linux.

Here's a list of essential software requirements:

1. Development Environment:
   * Java Development Kit (JDK):
   * Version 17 or later for Java development.Integrated Development Environment (IDE).
   * rs2xml.jar
   * Eclipse IDE for Java Developers or any preferred Java IDE.
2. Database:
   * Xampp or MySQL for storing customer information, meter readings, and billing data.

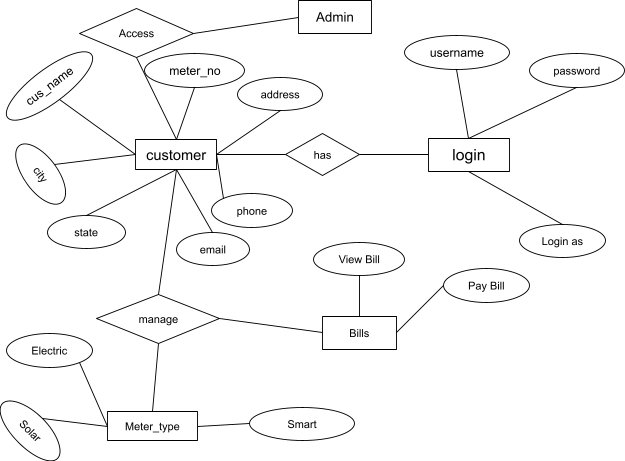
Database Connectivity:

* + JDBC (Java Database Connectivity) driver for connecting Java applications with the chosen DBMS.

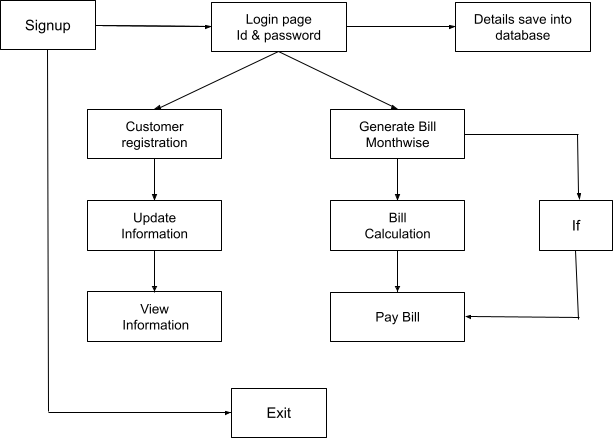
# Chapter-4

* 1. **SYSTEM DESIGN**

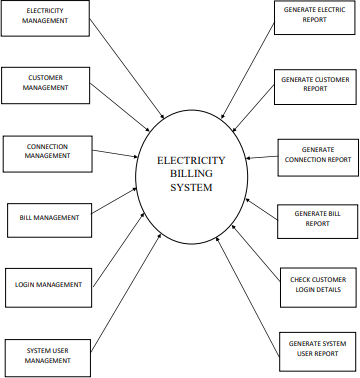
#### ER Diagram:-



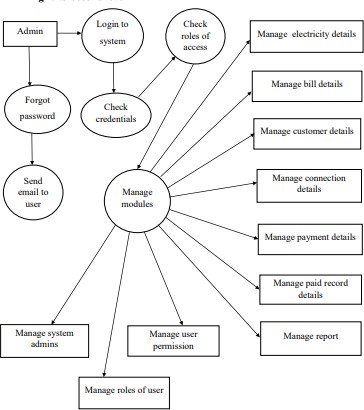
* + 1. **Data Flow Diagram**

Level 0

LEVEL1



LEVEL2



# Chapter-5

**5.1. IMPLEMENTATION**

###### Login.java

package electricity.billing.system; import java.awt.\*;

import java.sql.\*; import javax.swing.\*; import java.awt.event.\*;

@SuppressWarnings("serial")

public class Login extends JFrame implements ActionListener{

JButton login, cancel, signup; JTextField username, password; Choice loginas;

Login(){

super("Login Page"); getContentPane().setBackground(Color.white); setLayout(null);

JLabel lusername = new JLabel("username"); lusername.setBounds(300,20,100,20); add(lusername);

username = new JTextField(); username.setBounds(400,20,150,20); add(username);

JLabel lpassword = new JLabel("Password"); lpassword.setBounds(300,60,150,20); add(lpassword);

password = new JTextField(); password.setBounds(400,60,150,20); add(password);

JLabel logininas = new JLabel("Login in as"); logininas.setBounds(300,100,100,20); add(logininas);

loginas = new Choice(); loginas.add("Admin"); loginas.add("Customer"); loginas.setBounds(400,100,150,20); add(loginas);

ImageIcon i1 = newImageIcon (ClassLoader.getSystemResource ("icons/login.png")); Image i2 = i1.getImage().getScaledInstance(16,16, Image.SCALE\_DEFAULT);

login = new JButton("Login", new ImageIcon(i2)); login.setBounds(330,160,100,20); login.addActionListener(this);

add(login);

ImageIcon i3 = new ImageIcon(ClassLoader.getSystemResource("icons/cancel.jpg")); Image i4 = i3.getImage().getScaledInstance(16, 16, Image.SCALE\_DEFAULT); cancel = new JButton("Cancel", new ImageIcon(i4)); cancel.setBounds(450,160,100,20);

cancel.addActionListener(this); add(cancel);

ImageIcon i5 = new ImageIcon(ClassLoader.getSystemResource("icons/signup.png")); Image i6 = i5.getImage().getScaledInstance(16, 16, Image.SCALE\_DEFAULT); signup = new JButton("Signup", new ImageIcon(i6)); signup.setBounds(380,200,100,20);

signup.addActionListener(this); add(signup);

ImageIcon i7 = new ImageIcon(ClassLoader.getSystemResource("icons/second.jpg")); Image i8 = i7.getImage().getScaledInstance(250, 250, Image.SCALE\_DEFAULT);

ImageIcon i9 = new ImageIcon(i8); JLabel image = new JLabel(i9); image.setBounds(0, 0, 250, 250); add(image);

setSize(640, 300);

setLocation(400, 200); setVisible(true);

}

@Override

public void actionPerformed(ActionEvent e) { if (e.getSource() == login) {

String susername = username.getText(); String spassword = password.getText(); String user = loginas.getSelectedItem(); try {

Conn c = new Conn();

String query = "select \* from login WHERE username = '"+susername+"' and password = '"+spassword+"' and user = '"+user+"' ";

ResultSet rs = c.s.executeQuery(query); if(rs.next()) {

String meter = rs.getString("meter\_no"); setVisible(false);

new project(user, meter);

}else {

}

JOptionPane.showMessageDialog(null, "Invalid Login"); username.setText("");

password.setText("");

}catch (Exception ae){

ae.printStackTrace();

}

}else if (e.getSource() == cancel){ setVisible(false);

}else if (e.getSource() == signup) { setVisible(false);

new Signup();

}

}

public static void main(String[] args) { new Login();

}

}

**Project.java**

package electricity.billing.system; import javax.swing.\*;

import java.awt.\*; import java.awt.event.\*;

@SuppressWarnings("serial")

public class project extends JFrame implements ActionListener { String atype, meter;

project(String atype,String meter) { this.atype = atype; this.meter = meter;

setExtendedState(JFrame.MAXIMIZED\_BOTH);

ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icons/elect1.jpg")); Image i2 = i1.getImage().getScaledInstance(1550, 850, Image.SCALE\_DEFAULT);

ImageIcon i3 = new ImageIcon(i2); JLabel image = new JLabel(i3); add(image);

JMenuBar mb = new JMenuBar(); setJMenuBar(mb);

JMenu master = new JMenu("Master"); master.setForeground(Color.BLUE);

JMenuItem newcustomer = new JMenuItem("New Customer"); newcustomer.setFont(new Font("monospaced", Font.PLAIN, 12)); newcustomer.setBackground(Color.WHITE);

ImageIcon icon1 = new ImageIcon(ClassLoader.getSystemResource("icons/icon1.png"));

Image image1 = icon1.getImage().getScaledInstance(20, 20, Image.SCALE\_DEFAULT);

newcustomer.setIcon(new ImageIcon(image1)); newcustomer.addActionListener(this); master.add(newcustomer);

JMenuItem customerdetails = new JMenuItem("Customer Details"); customerdetails.setFont(new Font("monospaced", Font.PLAIN, 12)); customerdetails.setBackground(Color.WHITE);

ImageIcon icon2 = new ImageIcon(ClassLoader.getSystemResource("icons/icon2.png")); Image image2 = icon2.getImage().getScaledInstance(20, 20, Image.SCALE\_DEFAULT);

customerdetails.setIcon(new ImageIcon(image2)); customerdetails.addActionListener(this); master.add(customerdetails);

JMenuItem depositedetails = new JMenuItem("Deposite Details"); depositedetails.setFont(new Font("monospaced", Font.PLAIN, 12)); depositedetails.setBackground(Color.WHITE);

ImageIcon icon3 = new ImageIcon(ClassLoader.getSystemResource("icons/icon3.png")); Image image3 = icon3.getImage().getScaledInstance(20, 20, Image.SCALE\_DEFAULT);

depositedetails.setIcon(new ImageIcon(image3)); depositedetails.addActionListener(this); master.add(depositedetails);

JMenuItem calculatebill = new JMenuItem("Calculate Bill"); calculatebill.setFont(new Font("monospaced", Font.PLAIN, 12)); calculatebill.setBackground(Color.WHITE);

ImageIcon icon4 = new ImageIcon(ClassLoader.getSystemResource("icons/icon5.png")); Image image4 = icon4.getImage().getScaledInstance(20, 20, Image.SCALE\_DEFAULT);

calculatebill.setIcon(new ImageIcon(image4)); calculatebill.addActionListener(this);

master.add(calculatebill);

JMenu info = new JMenu("Information"); info.setForeground(Color.RED);

JMenuItem updateinformation = new JMenuItem("Update Information"); updateinformation.setFont(new Font("monospaced", Font.PLAIN, 12)); updateinformation.setBackground(Color.WHITE);

ImageIcon icon5 = new ImageIcon(ClassLoader.getSystemResource("icons/icon4.png")); Image image5 = icon5.getImage().getScaledInstance(20, 20, Image.SCALE\_DEFAULT);

updateinformation.setIcon(new ImageIcon(image5)); updateinformation.addActionListener(this); info.add(updateinformation);

JMenuItem viewinformation = new JMenuItem("View Information"); viewinformation.setFont(new Font("monospaced", Font.PLAIN, 12)); viewinformation.setBackground(Color.WHITE);

ImageIcon icon6 = new ImageIcon(ClassLoader.getSystemResource("icons/icon6.png")); Image image6 = icon6.getImage().getScaledInstance(20, 20, Image.SCALE\_DEFAULT);

viewinformation.setIcon(new ImageIcon(image6)); viewinformation.addActionListener(this); info.add(viewinformation);

JMenu user = new JMenu("User"); user.setForeground(Color.BLUE);

JMenuItem paybill = new JMenuItem("Pay Bill"); paybill.setFont(new Font("monospaced", Font.PLAIN, 12)); paybill.setBackground(Color.WHITE);

ImageIcon icon7 = new ImageIcon(ClassLoader.getSystemResource("icons/icon7.png")); Image image7 = icon7.getImage().getScaledInstance(20, 20, Image.SCALE\_DEFAULT);

paybill.setIcon(new ImageIcon(image7)); paybill.addActionListener(this); user.add(paybill);

JMenuItem billdetails = new JMenuItem("Bill Details"); billdetails.setFont(new Font("monospaced", Font.PLAIN, 12)); billdetails.setBackground(Color.WHITE);

ImageIcon icon8 = new ImageIcon(ClassLoader.getSystemResource("icons/icon8.png")); Image image8 = icon8.getImage().getScaledInstance(20, 20, Image.SCALE\_DEFAULT);

billdetails.setIcon(new ImageIcon(image8)); billdetails.addActionListener(this); user.add(billdetails);

JMenu report = new JMenu("Report"); report.setForeground(Color.RED);

JMenuItem generatebill = new JMenuItem("Generate Bill"); generatebill.setFont(new Font("monospaced", Font.PLAIN, 12)); generatebill.setBackground(Color.WHITE);

ImageIcon icon9 = new ImageIcon(ClassLoader.getSystemResource("icons/icon9.png")); Image image9 = icon9.getImage().getScaledInstance(20, 20, Image.SCALE\_DEFAULT);

generatebill.setIcon(new ImageIcon(image9)); generatebill.addActionListener(this); report.add(generatebill);

JMenu mexit = new JMenu("Exit"); mexit.setForeground(Color.BLUE);

JMenuItem exit = new JMenuItem("Exit"); exit.setFont(new Font("monospaced", Font.PLAIN, 12)); exit.setBackground(Color.WHITE);

ImageIcon icon12 = new ImageIcon(ClassLoader.getSystemResource("icons/icon11.png")); Image image12 = icon12.getImage().getScaledInstance(20, 20, Image.SCALE\_DEFAULT);

exit.setIcon(new ImageIcon(image12)); exit.addActionListener(this); mexit.add(exit);

if (atype.equals("Admin")) { mb.add(master);

} else {

mb.add(info);

mb.add(user);

mb.add(report);} mb.add(mexit); setLayout(new FlowLayout()); setVisible(true);}

public void actionPerformed(ActionEvent ae) { String msg = ae.getActionCommand(); if (msg.equals("New Customer")) {

new NewCustomer();

} else if (msg.equals("Customer Details")) { new CustomerDetails();

} else if (msg.equals("Calculate Bill")) { new CalculateBill();

} else if (msg.equals("Deposite Details")) { new DepositeDetails();

} else if (msg.equals("View Information")) { new ViewInformation(meter);

} else if (msg.equals("Update Information")) { new UpdateInformation(meter);

} else if(msg.equals("Bill Details")) { new BillDetails(meter);

} else if(msg.equals("Exit")) { setVisible(false);

new Login();

} else if(msg.equals("Pay Bill")) { new PayBill(meter);

} else if(msg.equals("Generate Bill")) { new GenerateBill(meter);

}}

public static void main(String[] args) { new project("","");

}

}

##### Paytm.java

package electricity.billing.system; import javax.swing.\*;

import java.awt.event.\*;

public class Paytm extends JFrame implements ActionListener{ String meter;

JButton back;

Paytm(String meter) {

this.meter = meter;

JEditorPane j = new JEditorPane(); j.setEditable(false);

try {

j.setPage("https://cspdcl.co.in/cseb/billPayment.aspx");

} catch (Exception e) { j.setContentType("text/html"); j.setText("<html>Could not load<html>");

}

JScrollPane pane = new JScrollPane(j); add(pane);

back = new JButton("Back"); back.setBounds(640, 20, 80, 30); back.addActionListener(this); j.add(back);

setSize(800, 600);

setLocation(400, 150); setVisible(true);

}

public void actionPerformed(ActionEvent ae) { setVisible(false);

new PayBill(meter);

}

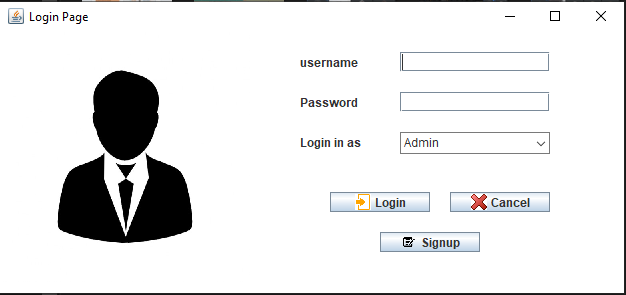
public static void main(String[] args) { new Paytm("");

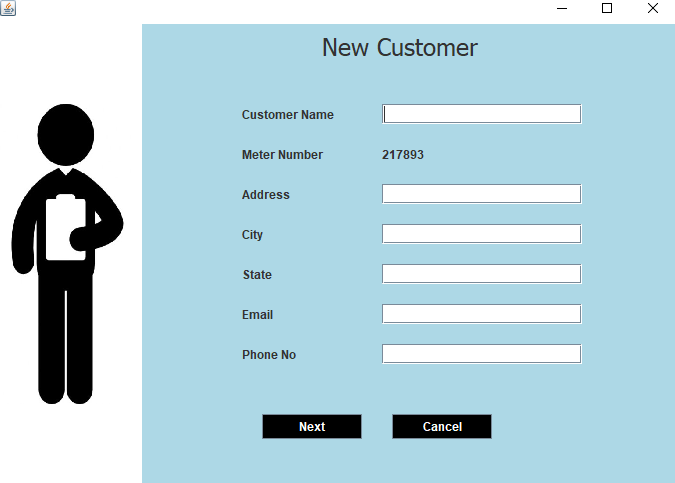
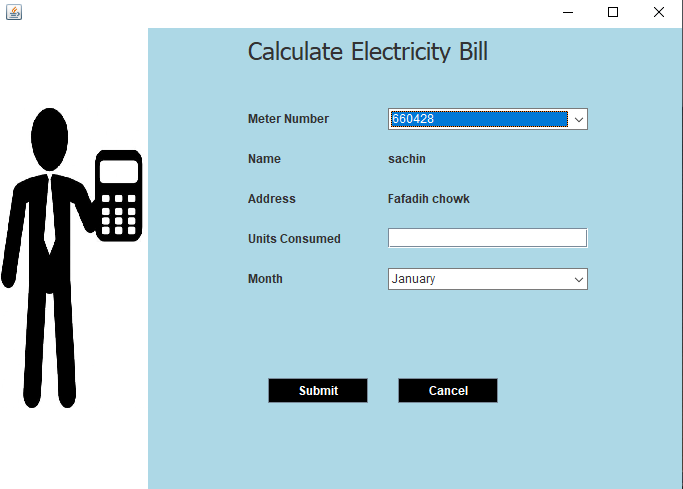
}

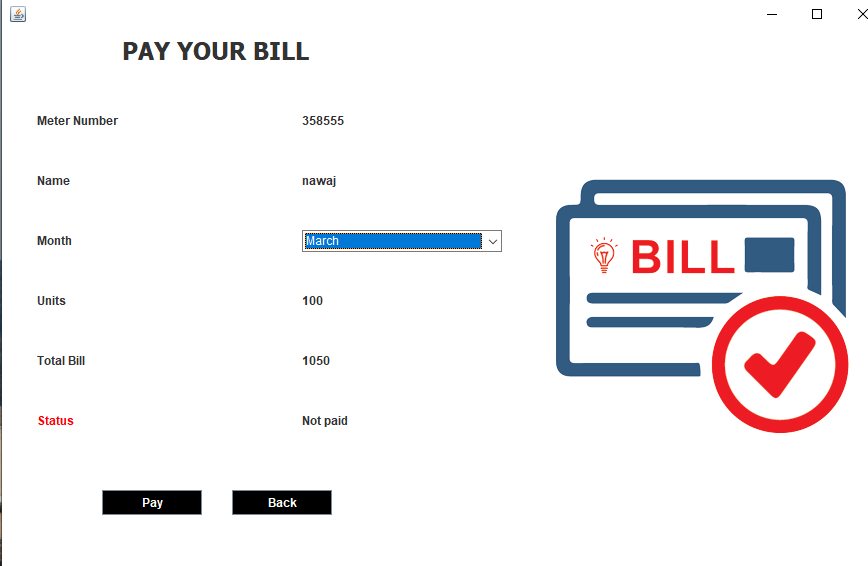
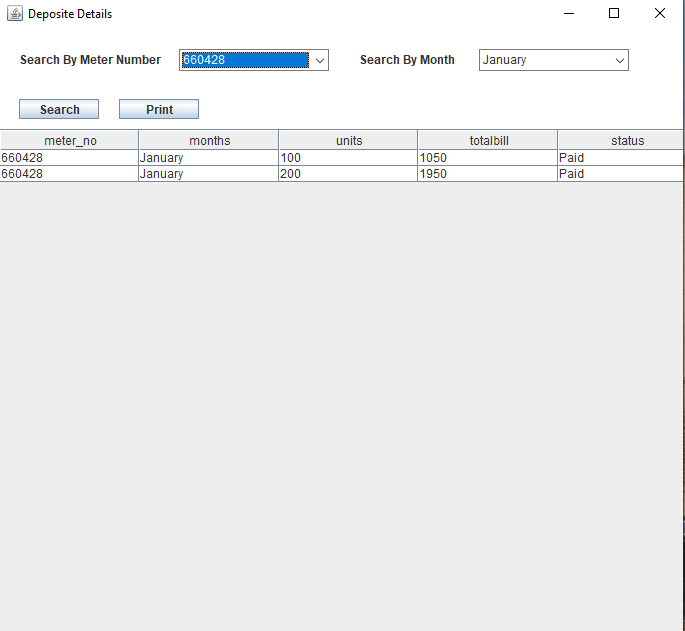
}

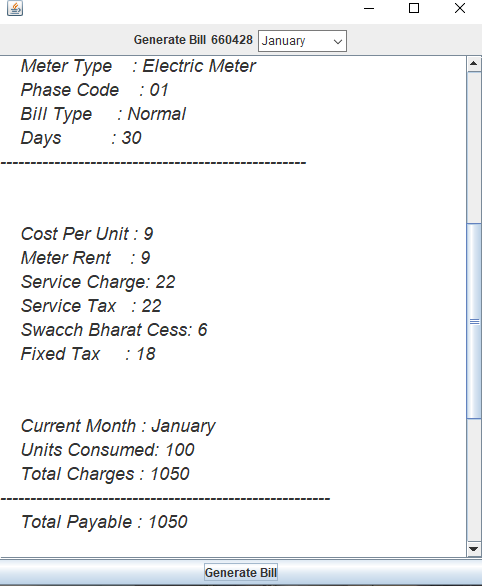
# Chapter-6

1. **SCREENSHOTS**









# Chapter-7

1. **1. SOFTWARE TESTING**

#### Testing process

Testing is an integral part of software development. Testing process, in a way certifies, whether the product, that is developed, complies with the standards that it was designed to. Testing process involves building of test cases, against which, the product has to be tested. In some cases, test cases are done based on the system requirements specified for the product/software, which is to be developed.

#### Testing objectives

The main objectives of testing process are as follows:

* + - * Testing is a process of executing a program with the intent of finding an error.
      * A good test case is one that has a high probability of finding an as yet undiscovered error.
      * A successful test is one that uncovers an as yet undiscovered error.

##### Unit Testing

Unit testing focuses verification effort on the smallest unit of software design the module. The software built is a collection of individual modules. In this kind of testing the exact flow of control for each module was verified. With detailed design consideration used as a guide, important control paths are tested to uncover errors within the boundary of the module.

#### Negative test case for phone number insertion:-

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function name** | **Input** | **Expected error** | **Error** | **Resolved** |
| Input Phone Number | 98777 | Phone is Invalid | Length of phone number is not equal to 10 | Consume() |

**Positive test case for phone number insertion:-**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function name** | **Input** | **Expected error** | **Error** | **Resolved** |
| Input phone number | 7999745668 | Expected output is seen |  |  |

#### Integration testing:-

The second level of testing is called integration testing. In this, many classtested modules are combined into subsystems, which are then tested. The goal here is to see if all the modules can be integrated properly. We have been identified and debugged.

**Test case on basis of generation of bill** ➖

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function name** | **Input** | **Expected error** | **Error** | **Resolved** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Negative searching of total bill | 1234(meter no) January(month) | Details seen but not total bill Output not seen | Output not seen | Consume() |
| Positive searching of total bill | 1234(meter no) January(month) | Must display full generated bill |  |  |

# Chapter-8

1. **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.

# Chapter-9

1. **FUTURE ENHANCEMENTS**

We have left all the options open so that if there is any other future requirement in the system by the user for the enhancement of the system then it is possible to implement them. We hope that the project will serve its purpose for which it is develop there by underlining success of process.

# BIBLIOGRAPHY

* + Java: The Complete Reference, Eleventh Edition
  + [https://www.w3schools.com](http://www.w3schools.com/)
  + [https://GeeksofGeeks.com/java](https://geeksofgeeks.com/java)
  + [https://chatopenAi.com](https://chatopenai.com/)
  + youtube- Apna college, Coding Wallah sir