R.M.D. ENGINEERING COLLEGE

**(An Autonomous Institution)**

**R.S.M Nagar, Kavaraipettai, Gummidipoondi Taluk, Thiruvallur District, Tamil Nadu- 601206**

Affiliated to Anna University, Chennai / Approved by AICTE, New Delhi/Accredited by NAAC

An ISO 9001:2015 Certified Institution / All the Eligible UG Programs are accredited by NBA, New Delhi

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**21IT413**

**Internship**

**LAB MANUAL**

**Regulation 2021**

**IV Semester**

### 20IT401 SOFTWARE ENGINEERING LABORATORY OBJECTIVES:

* To understand the software engineering methodologies for project development.
* To gain knowledge about open source tools for Computer Aided Software Engineering.
* To develop an efficient software using case tools.

### SOFTWARE REQUIRED:

Open source Tools: Star UML / UMLGraph / Topcased

Prepare the following documents for each experiment and develop the software using softwareengineering methodology.

1. **Problem Analysis and Project Planning -**Thorough study of the problem – Identify Projectscope, Objectives and Infrastructure.
2. **Software Requirement Analysis -** Describe the individual Phases/modules of the project andIdentify deliverables.
3. **Data Modelling -** Use work products – data dictionary, use case diagrams and activitydiagrams, build and test class diagrams, sequence diagrams and add interface to class diagrams.
4. **Software Development and Debugging** – implement the design by coding
5. **Software Testing** - Prepare test plan, perform validation testing, coverage analysis, memoryleaks, develop test case hierarchy, Site check and site monitor.

### SAMPLE EXPERIMENTS:

**Academic domain**

1. Course Registration System
2. User marks analyzing system

### Railway domain

1. Online ticket reservation system
2. Platform assignment system for the trains in a railway station

### Medicine domain

1. Expert system to find the ids for the given user id
2. Remote computer monitoring

### Finance domain

1. ATM system
2. Stock maintenance

### Human Resource management

1. Quiz System
2. E-mail Client system.

**INDEX**

|  |  |
| --- | --- |
| **SL.NO** | **NAME OF THE EXPERIMENT** |
|  | **ELECTRICITY BILL MANAGEMENT SYSTEM** |
| 1 | Problem Analysis  1(a)Problem Statement |
|  | 1(b)Project Planning |
| 2 | Software Requirement Analysis |
| 3 | Modeling  3(a)Design 3(b)Data Dictionary |
| 4 | Implementation |
| 5 | Testing - Test Cases |
| 6 | Documentation |

# Ex.No 1(a) PROBLEM ANALYSIS

**Problem Statement**

Computerized Electricity bill Management system is used to monitor and control the transactions in a Electricity bill. It needs to maintain the record of new users and retrieve the details of users available in the Electricity bill . Should be able to perform basic operations in a Electricity bill like adding new users, new consumers, and updating new connection, searching user record and consumer record and provide facility to print bill after discharge.

# Analysis

Electricity bill Management System should allow the user to add new users and connection. The system has three categories of users: Administrator, User and consumer.

Administrator should be able to:

* Register new customer
* View the customer record
* Update the record
* View user record
* Update user details
* Admit new user
* Approve user
* Discharge user
* Generate bill
* View the available connection
* Book new connection
* Approve the awaiting connection

User should be able to:

* Book and view his/her connection
* View the consumers treating him/her
* View the bill

consumer should be able to:

* View his users (present and terminated)
* View and delete connection

# Feasibility study

### Technical feasibility

The Electricity bill Management System (LMS) runs with a minimum system resources:

* Python
* Django
* Sqlite3
* Html

Above said system resources are available as open source. Hence it is feasible to develop HMS in this environment.

### Operational feasibility

As the system has HTML based GUI no special skill set is required for working with the system, hence it is operationally feasible.

### Economic feasibility

As the LMS requires minimum system resources, hence it is economically feasible.

**Ex.No 1(b) PROJECT PLANNING**

# Overview

Automation of Electricity bill Management System is to handle the entire activity of a . The system keeps track of all the information about the users and their complete details. The system contains a database where all the information will be stored. The system is user-friendly.

# Goals and Scope

**Goal :** To automate the Electricity bill Information System with the following functional goals

* + 1. Login
    2. Adding User/consumer Details.
    3. Updating User/consumer Details
    4. Deleting User/consumer Details
    5. Generating Bill
    6. Viewing/Searching for Details.

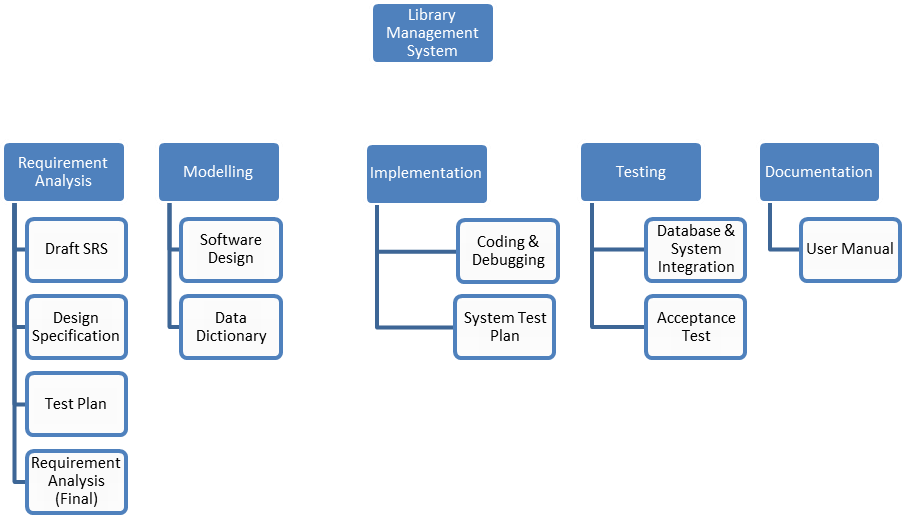
**Scope :** The system accepts the General Electricity bill Transactions of connection issue, consumer record and for the members. Different areas where we can use these applications are:

1. Any Electricity bill institutes can make use of it for providing information about consumers, connection of the available users.
2. The system would provide basic set of features to add/update users, add/update consumers, and manage connection and generate bills.

# Schedule and Budget Work Breakdown Structure

Electricity bill

Management



**Schedule and Milestones**

|  |  |  |  |
| --- | --- | --- | --- |
| **Milestones** | **Description** | **Milestone Criteria** | **Planned week** |
| M0 | Problem Analysis |  | 1st week |
|  |  | Problem statement, Analysis, Feasibility Study |  |
| M1 | Project Planning |  | 2nd week |
|  |  | Scope and concept described |  |
| M2 | Requirement Analysis |  | 2nd and 3rd week |
|  |  | Draft SRS, Design Specification, Test Plan, Requirement Analysis (Final) |  |
| M3 | Study of UML Notations |  | 3rd week |
|  |  | Architecture reviewed and stable |  |
| M4 | Modeling |  | 4th week |
|  |  | Software Design, Data Dictionary |  |

**Problem Analysis**

**Create Data Dictionary**

**Problem Planning**

**Require- ment Analysis**

**Model- ing**

**Coding**

**UML**

**design**

**Testing**

|  |  |  |  |
| --- | --- | --- | --- |
| **Milestones** | **Description** | **Milestone Criteria** | **Planned week** |
| M5 | Implementation |  | 5th week |
|  |  | Coding of functionality, Debugging, System Test Plan. |  |
| M6 | Testing |  | 6th week |
|  |  | Database & System Integration, Acceptance Testing |  |
| M7 | Documentation |  | 7th week |
|  |  | User Manual |  |

# Budget

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Category** | **Budget for Period in kUS$** | | | | | |
| **M0-M1** | **M1-M2** | **M2-M3** | **M3-M4** | **M4-M5** | **M5-M6** |
| Human Resources (internal) |  |  |  |  |  |  |
| Human Resources (external) |  |  |  |  |  |  |
| Purchases (COTS) |  |  |  |  |  |  |
| Equipment |  |  |  |  |  |  |
| Premises |  |  |  |  |  |  |
| Tools |  |  |  |  |  |  |
| Travel costs |  |  |  |  |  |  |
| Training |  |  |  |  |  |  |
| Review activities |  |  |  |  |  |  |
| Other |  |  |  |  |  |  |
| Total | 1 | 1 | 2 | 5 | 2 | 1 |
| **Total cumulated** | **1** | **2** | **4** | **9** | **11** | **12** |

For a detailed list of costs of all resources see <document> [x].

# Development Process

**Risk Management**

Unexpected Holidays, Non availability of computer resources, Absence of Human Resource are the identified risks for not meeting the deadlines. Additional efforts need to put in by the human resources to complete the work within the deadline by the way of working after working hours.

# Delivery Plan

|  |  |  |  |
| --- | --- | --- | --- |
| **Ident.** | **Deliverable** | **Planned Date** | **Receiver** |
| D1 | Analysis and Feasibility Report | 1st week | Client |
| D2 | Project Plan | 2nd week | Client |
| D3 | SRS | 3rd week | Client |
| D4 | Design | 4th week | Client |
| D5 | Test Plan | 5th week | Client |
| D6 | Code | 6th week | Client |
| D7 | Test Report | 6th week | Client |

**Ex.No.2 SOFTWARE REQUIREMENT ANALYSIS**

# Software Requirement Specification (SRS)

1. **Introduction**

Electricity bill Management system is used to monitor and control the transactions in a Electricity bill. It needs to maintain the record of new users and retrieve the details of users available in the Electricity bill. Should be able to perform basic operations in a Electricity bill like adding new consumers/ users, updating consumer/user information, searching for consumers/users and provide facility to user connection and view bill. The report generation facility should allow generating various reports.

Booking connection or viewing the available connection at the Electricity bill is currently done manually. This project is specifically designed for the use of receptionists and Electricity bill users. It is especially useful for an corporate Electricity bills where modifications in the content can be done easily according to the requirements.

# Purpose of the requirements document

The purpose of this document is to analyze and elaborate on the high-level needs and features of the Electricity bill System**.** It focuses on the capabilities and facilities provided by a Electricity bill. The main Objective of this document is to illustrate the requirements of electricity bill Management System. This document gives a detailed description of both functional and non-functional requirements proposed by the clients.

This document defines and describes the operations, interfaces, performance, and quality assurance requirements of the Electricity bill System. The document also describes the nonfunctional requirements such as the user interfaces. It also describes the design constraints that are to be considered when the system is to be designed, and other factors necessary to provide a complete and comprehensive description of the requirements for the software.

# Scope of the product

The Software Requirements Specification captures all the requirements in a single document. The Electricity bill Management System that is to be developed provides the Administrator, user and consumers with connection information, and many other facilities.

The system accepts the General Electricity bill Transactions of connection booking, view and pay for the bill. The system would provide basic set of features to add/update consumers,

add/update users and manage bills in specifications for the systems based on the client's statement of need.

# Definitions, acronyms and abbreviations

PHP- Hypertext Preprocessor SQL- Structured Query Language GUI- Graphical User Interface

HMS – Electricity bill Management System

# References

1. Roger s Pressman,” Software Engineering -a practitioner's approach” 7th edition.
2. Ian sommerville, “software engineering “ 9th edition

# 1.5. Overview of the remainder of the document

The SRS will provide a detailed description of the Electricity bill Management System. This document will provide the outline of the requirements, overview of the characteristics and constraints of the system.

Section 2 of this document provides the General description such as Product perspective, Product functions and the characteristics of the user’s of this product. Section 3 describes the Specific requirements which cover the functional, non-functional and interface requirements. This is obviously the most substantial part of the document but because of the wide variability in organizational practice, it is not appropriate to define a standard structure for this section. The requirements may document external interfaces, describe system functionality and performance, specify logical database requirements, design constraints, emergent system properties and quality characteristics.

# General description Product perspective

The Electricity bill Management System is a package to be used by Electricity bills to improve the efficiency of Administrator and Users. The system provides consumers catalog and information to members and helps them decide on the consumers to treat from the Electricity bill. The receptionist can keep the connection catalog updated all the time so that the users get the updated information all the time.

# Product functions

The Electricity bill Management System provides real time information about the connection available in the

Electricity bill and the user information. The Product functions are more or less the same as described

in the product perspective. The functions of the system include the system providing different type of services based on the type of users [Administrator/User].

1. Administrator should be able to add consumer ,user connection and generate bill.
   * Add consumer
   * user connection
   * Generate bill
2. consumer should be able to access user under the respective consumer and discharge summary

Of the user

1. User should able to view the corresponding consumer for their respective problem and get the

consumer information.

# User characteristics

The users of the system are administrators, User and consumer members, the administrator who maintain the system. The members are assumed to have basic knowledge of the computers . The administrators of the system to have more knowledge of the internals of the system and is able to rectify the small problems that may arise due to disk crashes, power failures and other catastrophes to maintain the system. The proper user interface, user’s manual, online help and the guide to install and maintain the system must be sufficient to educate the users on how to use the system without any problems.

# General constraints

* + The information of all the users must be stored in a database that is accessible by the Electricity bill Management System.

.

* + The Electricity bill Management System is connected to the server computer and is running all 24 hours a day.

# Assumptions and dependencies

* + The users have sufficient knowledge of computers.
  + The users know the English language, as the user interface will be provided in English
  + The product can access the college user database

# Specific requirements Functional Requirements

This section describes in detail all the functional requirements.

(It shows what the system can do)

* 1. Login
  2. Appoint user
  3. Add consumers
  4. Get user information
  5. Appoint user to respective consumer
  6. Generating discharge summary
  7. Generate bill

# Non- Functional Requirements

### Usability

* The system is user friendly and self-explanatory.

### Reliability

The system has to be very reliable due to the importance of data and the damages incorrect or incomplete data can do.

### Availability

The system is available 100% for the user and is used 24 hrs a day and 365 days a year. The system shall be operational 24 hours a day and 7 days a week.

### Mean Time Between Failures (MTBF)

The system will be developed in such a way that it ***may*** fail once in a year.

### Mean Time to Repair (MTTR)

Even if the system fails, the system will be recovered back up within an hour or less.

### Accuracy

The accuracy of the system is limited by the accuracy of the speed at which the employees of the Electricity bill and users of the Electricity bill use the system.

### Maximum Bugs or Defect Rate

Not specified.

### Access Reliability

The system shall provide 100% access reliability.

### Performance Response Time

The system shall respond to the member in not less than two seconds from the time of the request

submittal. The system shall be allowed to take more time when doing large processing jobs.

The requirements may document external interfaces, describe system functionality and performance, specify logical database requirements, design constraints, emergent system properties and quality characteristics.

# Hardware and software requirements

## Hardware Interfaces

Processor: Pentium or Higher. RAM: 312MB or Higher.

## 3.3..2. Software Interfaces

Operating System: Unix, Linux, Mac, Windows etc.. Development tool: Django ,python

Data Base: Sqlite3

# 3.4 External Interfaces

### User Interfaces

The user-interface of the system shall be designed as shown in the user-interface prototypes.

# Appendices

1. **Index**

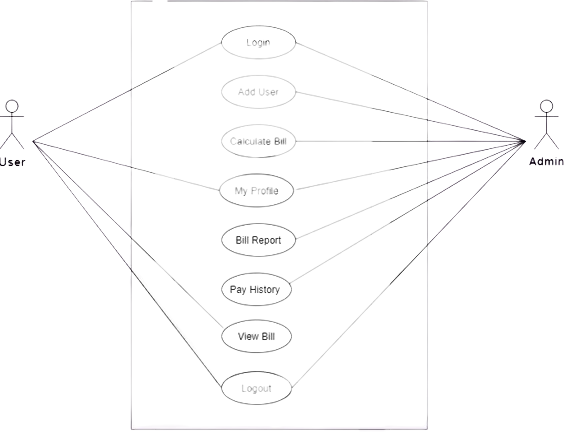
**Result:** Thus the Software Requirement Specification Document for Electricity bill Management System has been completed.

# Ex.No. 3 MODELING

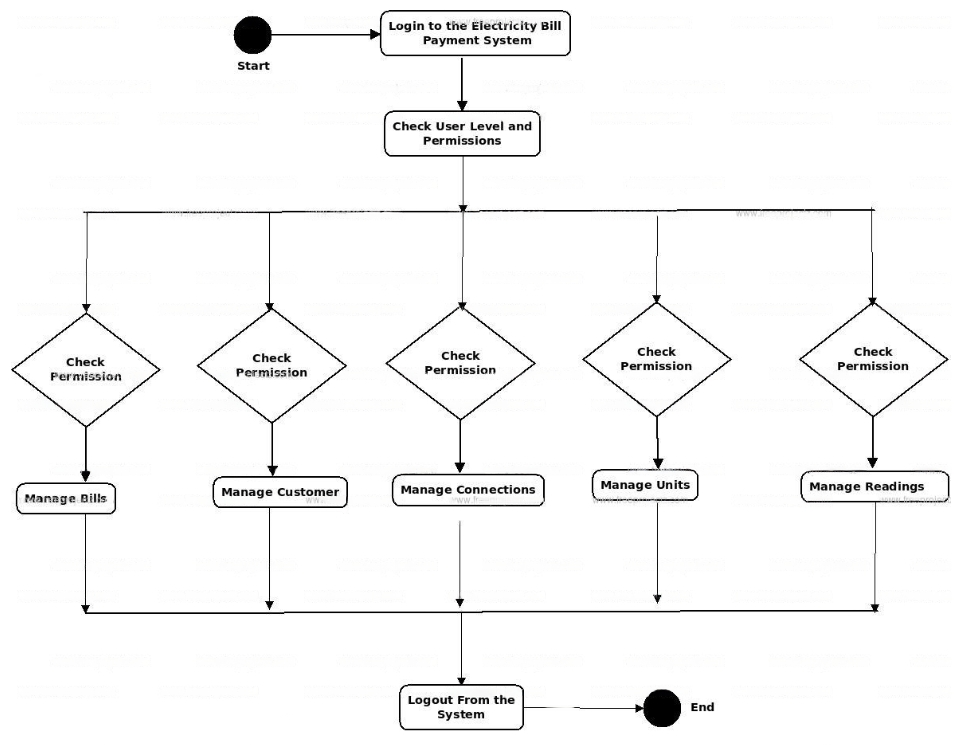
**(i) Design model –UML diagrams**

**Use case diagram**

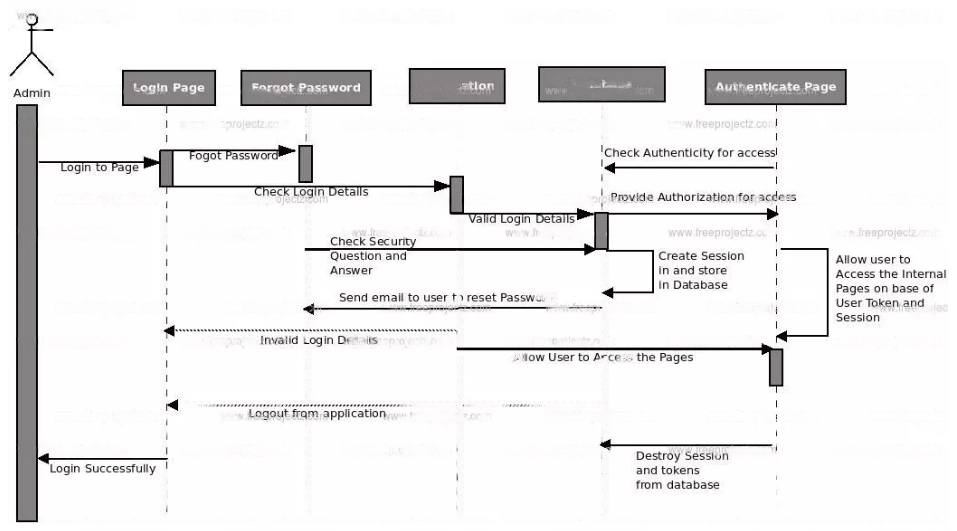
**Step 1:start ARGO UML->Create-> Use Case Diagram**



**Activity Diagram Electricity bill management**



**Sequence diagram for electricity bill management of user**

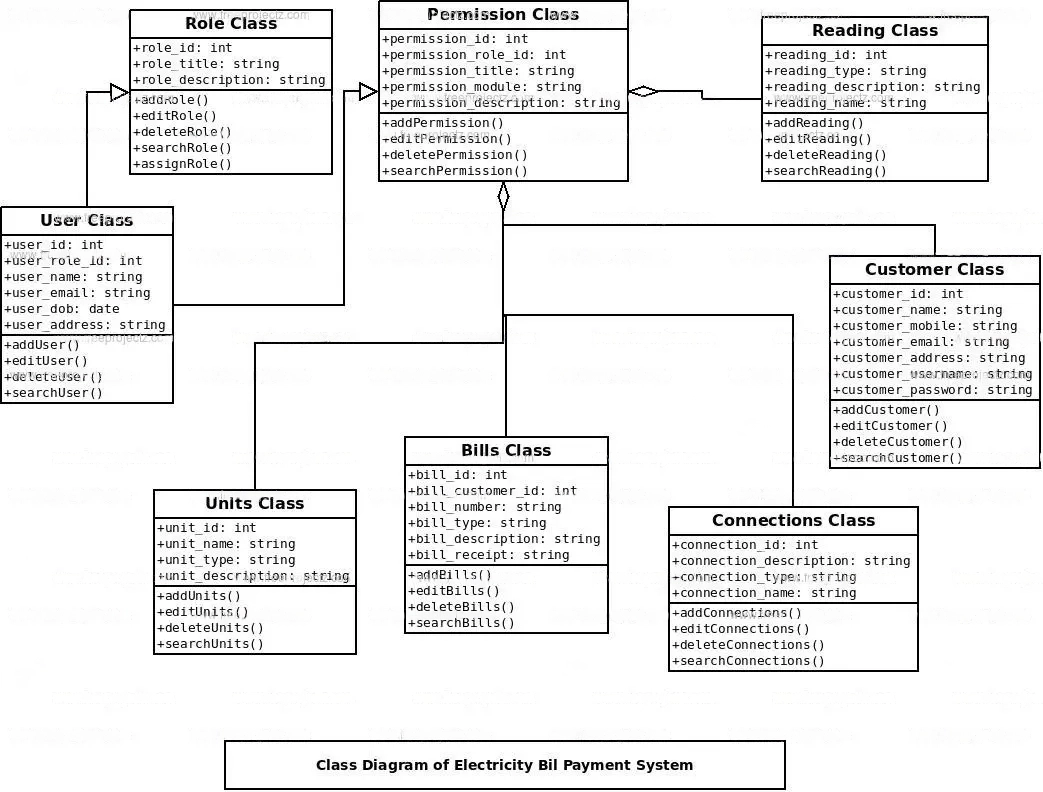
****

**Collaboration diagram for electricity bill management of user**

Diagram

Description automatically generated

**Class Diagram**



# Ex.No.3 (b) DATA DICTIONARY

**User Details**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.No** | **Name** | **Alias Name** | **Where Used/How Used** | **Supplementary Data** | |
| **Data Type** | **Limitations** |
| 1 | consumer name | Title | Borrow, Display | string | Up to 20 char |
| 2 | Symptoms | - | Return, Display | string | Up to 250 char |
| 3 | Address |  | Borrow, Display | string | Up to 200 char |
| 4 | consumer Mobile |  | Borrow, Display | integer | Up to 10 digits |
| 5 | Admit date |  | Borrow, Display | integer | Up to 8 digit |

# consumer Details

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.No** | **Name** | **Alias Name** | **Where Used/How Used** | **Supplementary Data** | |
| **Data Type** | **Limitations** |
| 1 | consumer name | - | Membership, Borrow, Display | string | Up to 20 char |
| 2 | Specialization | - | Membership | string | Up to 20 char |
| 3 | Appointments |  | Membership | string | Up to 20 char |
| 4 | Users |  | Membership | string | Up to 100 char |
| 5 | User Discharge |  | Membership | integer | Up to 8 digit |

**Ex.No.4 IMPLEMENTATION**

# Login use case:

**<html>**

**<style>**

**body {**

**background-image: url(2.jpg);**

**background-repeat: no-repeat;**

**background-size: 100% 100%;**

**height: 610px;**

**padding-top: 100px;**

**}**

**h1 {**

**text-align: center;**

**}**

**ul {**

**display: flex;**

**list-style-type: none;**

**margin: 50px 100px;**

**font: 17px helvetica;**

**text-align: center;**

**}**

**li {**

**flex: 1;**

**background: #1a1a1a;**

**}**

**li:hover {**

**flex: 1.1;**

**transition: all .2s ease;**

**background: #333333;**

**}**

**a {**

**display: block;**

**position: relative;**

**text-decoration: none;**

**color: white;**

**padding: 23px;**

**}**

**.navigation li:not(:first-child) {**

**border-left: 2px solid #ecf0f1;**

**}**

**</style>**

**<head>**

**<meta charset="UTF-8">**

**<meta http-equiv="X-UA-Compatible" content="IE=edge">**

**<meta name="viewport" content="width=device-width, initial-scale=1.0">**

**</head>**

**<body>**

**<div>**

**<nav>**

**<ul class="navigation">**

**<li><a href="home">Home</a></li>**

**<li><a href="customer.php">Add Customer</a></li>**

**<li><a href="bill.php">Billing</a></li>**

**<li><a href="user.php">Add Admin</a></li>**

**<li><a href="logut.php">Logout</a></li>**

**</ul>**

**</nav>**

**</div>**

**</body>**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<meta name="viewport" content="width=device-width, initial-scale=1">**

**<title> Login Page </title>**

**<style>**

**Body {**

**font-family: Calibri, Helvetica, sans-serif;**

**background-image: url(1.jpg);**

**background-size: 100%;**

**}**

**button {**

**background-color: #e26411;**

**width: 100px;**

**color: rgb(24, 23, 22);**

**padding: 15px;**

**margin: 10px 0px;**

**border: none;**

**cursor: pointer;**

**text-align: center;**

**border-radius: 5px;**

**}**

**input[type=text],**

**input[type=password] {**

**width: 300px;**

**margin: 8px 0;**

**padding: 12px 20px;**

**display: inline-block;**

**border: 2px solid green;**

**box-sizing: border-box;**

**}**

**button:hover {**

**opacity: 0.7;**

**}**

**.cancelbtn {**

**width: auto;**

**padding: 10px 18px;**

**margin: 10px 5px;**

**border-radius: 5px;**

**}**

**.container {**

**padding: 25px;**

**background-color: rgb(115, 201, 194);**

**width: 400px;**

**height: 250px;**

**line-height: 80px;**

**border-radius: 20px;**

**}**

**form {**

**padding-top: 50px;**

**padding-left: 50px;**

**}**

**h1 {**

**padding-left: 100px;**

**padding-top: 50px;**

**}**

**</style>**

**</head>**

**<body>**

**<br><br><br><br><br><br><br><br>**

**<h3> Admin Login </h3>**

**<form method="post" action="process.php">**

**<div class="container">**

**<label>Username : </label>**

**<input type="text" placeholder="Enter Username" name="Username" required><br>**

**<label>Password : </label>**

**<input type="password" placeholder="Enter Password" name="Pass" required><br>&nbsp;&nbsp;**

**<button type="submit">Login</button>**

**</div>**

**</form><br><br>**

**<h3> Customer Login </h3>**

**<form method="post" action="process2.php">**

**<div class="container">**

**<label>Firstname : </label>**

**<input type="int" placeholder="Enter Firstname" name="Firstname" required><br>**

**<label>Phone No. : </label>**

**<input type="int" placeholder="Enter Phone No." name="Phoneno" required><br>&nbsp;&nbsp;**

**<button type="submit">Login</button>**

**</div>**

**</form>**

**<br><br>**

**<?php if(isset($\_GET['err'])){**

**echo "<script>alert('Invalid Username or Pass')</script>";**

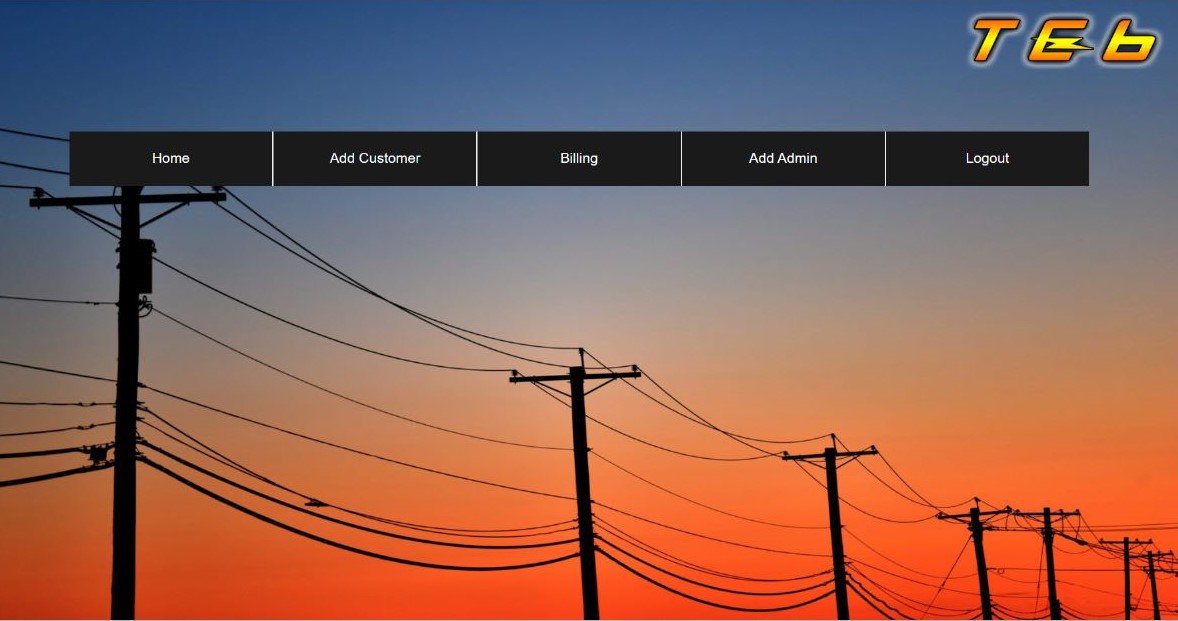
**} ?>**

**</body>**

**</html>**

**</html>**

**OUTPUT:**



# Ex.No 5 TESTING

### Test cases:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Requirement** | **Description** | **Input** | **Expeted o/p** | **Actual o/p** |
| LOGIN | Username & password should be entered | USER: Can login using unique Id and Password after this system shall show his/her profile.  user: Can login using unique Id and Password after this system shall show his/her profile.  ADMIN: Can login using unique Id and Password after this system shall show a  profile with links to maintain the website. | Username,password | Valid msg | Valid msg |
| REGISTRATION | Username & password should be entered | USER: Can Register by filling all the required details, after this the system will verify the details and check if already  registered or not. | Username,password | Valid msg | Valid msg |
| MAKE APPT. | User should apply and admin should approve | USER: Can Select consumer, date time and make an connection request after this system shall show a confirmation for  connection request. | User name , consumer name, date and time. | Valid msg | Valid msg |
| CANCL APPT. | consumer has the facility to cancel connection | USER : Can Cancel connection if want to by just one click after this system shall ask for re-schedule or refund of payment.  user : Can Cancel connection if want to by just one click after this system shall  send a message to the user. | User id number | Valid msg | Valid msg |
| PAYMENT | Should be redirected to Payment portal | USER : Enter payment details and make payment after this system shall show the  generated bill by the Electricity bill. | Bill no ,select payment method | Valid msg | Valid msg |
| user MODULE | All details of consumers | ADMIN : Can add a new consumer by filling all the details after this system shall show a confirmation message.  Can Remove a consumer by just one click after this system shall show confirmation  message. | Username,password | Valid msg | Valid msg |
| USER MODULE | All details of users | USER : Can view payment history or can search for a particular bill also after this system shall show a bill or history.  Can also See or search for a consumer by entering dept. name or consumer id if known after this system will check for the consumer if found shall show consumer’s profile.  Can also update details after this system shall ask for re-enter password and after  verifying password shall update details. | Username,password | Valid msg | Valid msg |
| ADD BILL | Bill generated by consumer | user : Enter User Id and after this all the details and id, remark and advice for the user after this system  shall show a message for update. | user id | Valid msg | Valid msg |