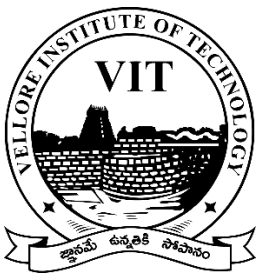
CSE1005 Software Engineering

PROJECT REPORT ON
WEB BASED BILLING SOFTWARE

Faculty: Dr. Hussian Ahmed Choudhary

Submitted by :

1. Gudi Varaprasad **-** 19BCE7048 **(Team Lead)**
2. S S V M Harshith Kanigalpula **-** 19BCE7349
3. Abhipray Adhik Chavan **-** 19BCN7062
4. Talasila Sri Harsha **-** 19BCE7490
5. Nandigam Pramodini **-** 19BCN7037



VIT®
AP

INDEX

INTRODUCTION:

The main objective of this project is to make Billing management much easier for Business Enterprises. In this project, we have developed a **"Web-based Billing Software"** which is an application to automate the process of billing and managing a departmental store, which would be responsive by using HTML, CSS, Bootstrap, and JavaScript as front end tools and the background is managed by a database (MySQL), PHP, phpMyAdmin.

This project includes several modules such as the register/login module, the admin module, bill management module, report generation & giveaway module.
This software allows company owners to set up online bill management system , consumers to read bills, and owners to save or download invoices for new businesses. It is developed on a network of department stores around the country. It also assists clients in creating, updating, and calculating bills/invoices without to manually listing them.

Also, this application leads to a secure reliable, error-free, and fast management system for maintaining bills/invoices or computing whole bill computations. This also maintains computerized records of data and this data can be stored for a certain period and provides summary reports to owners monthly or weekly & yearly with easy accessing.

LITERATURE SURVEY:

i. Problem Definition:

This project is an application to automate the process of billing and managing a Departmental store.

ii. Understand the Root Causes:

Existing systems include a lot of paperwork and require a lot of manual labour; this may be remedied by automating the process with a web-based GUI.

iii. Identify the Stakeholder and Users:

Stakeholders - admin, developers, target audience, funders, clients.
Users - Businessmen, Enterprises, Shopkeepers, Vendors, Govt., officers.

iv. Identifying the constraints to be imposed on the solution:

a. Customers without Internet Access.
b. Information Privacy.
c. not cost-effective for small scale business owners.

PROBLEM STATEMENT:

i. Problems of Existing Softwares :

a. Difficult to maintain bills and manage huge data manually.

b. Reports are difficult to generate from existing data.
c. Full or a lot of paperwork and Time consuming.
d. Inability of modification of data.

ii. Specifications of the Proposed Software:
a. Easiness in maintaining and modification of sold products.
b. Reports are simple to implement, create, generate within no time.
c. No or very few paperwork.

OBJECTIVE:

The main objective of this project is to generate and manage invoices in a matter of few seconds in order to avoid a lot of paperwork and Time consuming process by managing huge data manually. This project benefits owners of enterprises as well as customers. This software allows to maintain all the products with their shorthand notation by providing summary reports to the owner, which gives a convenient solution to the billing pattern with a Secured User Authentication and Validation. And also user Giveaway generator whenever offers or discounts are available.

1. Account creation module (Sign in, Sign up)
2. Bill Generation module (Create, Read, Update, Save)
3. Report Generation module (Print, Download)
4. Bill Calculation Module (Math, computation)
5. Admin module (Managing user modules)
6. Product Shorthand notation module
7. Random giveaway generator module
8. Logout module (Session Delete)

FEASIBILITY:

The key consideration involved in the feasibility study are:
a. Technical Feasibility :

S. No.	Tools / Technology needed	Description and uses	Type
1.	HTML5	Appearance of website	Front-end Software
2.	CSS3	Adding style and presentation to the web-pages	Front-end Software
3.	JavaScript	Adding interactive user behavior to web pages	Front-end Software

4.	Bootstrap	front-end framework used for UI	Front-end Software
5.	PHP	Server side programming and interactive with phpMyAdmin	Back-end Software
6.	MySQL	For the purpose of a web database and logging application	Database Software
7.	phpMyAdmin	Management of databases, tables, indexes, permissions	Database Software
8.	Lucidchart	a web-based proprietary platform to collaborate on drawing charts and diagrams	Design Tool Software
9.	Heroku	a container-based cloud Platform as a Service to deploy, manage, scale apps	Cloud Platform Software
10.	Visual Studio Code	a code editor redefined, optimized for building, debugging web apps	Desktop Software
11.	Chrome	a cross-platform web browser	Software
12.	Windows 10	an operating system for building apps	OS
13.	PC / Laptop	Intel based processor-run computer system, which have keyboard and mouse as input devices.	Hardware

b. Behavioural Feasibility :

An assessment of end-user behaviour that may have an impact on the system's Encirclement. People are naturally reluctant to change, and computers must be cognizant of this in order to support changes.

Since the proposed system's sole purpose is to meet information demands, no employees would lose their jobs as a result of it. Because it supports the organization and its strategic objective, this software solution is also possible for enterprises.

c. Economic Feasibility :

Type	Quantity	Figures
Human Power	5 to 6 members	$25 - $30 per day per head
Software	-	$40 - $50 approx
Hardware	3 or 4 sets	$80 - $100 approx

Total estimated budget - $4000 - $4500

d. Timeline :

Schedule	Phase
1st week	Planning, Communication
2nd week	Design
3rd week	Development, Coding
4th week	Testing, Deployment

FUNCTIONAL REQUIREMENTS:

This section gives a high-level summary of the system requirements. The system will be able to implement a variety of functional modules, including:

6.1.0 Description:

The key consideration involved in functional requirements are the needs that the end user expresses as essential features that the software should provide.

6.1.1 Signup / Registration:

If a client wishes to create an invoice or manage, download, or print a bill, he or she must first register. Unregistered users are unable to do the above tasks.

6.1.2 Login:

The user logs in to the online application using a valid user id and password that has been successfully validated in order to manage invoices.

6.1.3 Bill Generation:

Every logged-in user will be able to create and generate invoices,as well as store current bills, print, and download those generated bills.

6.1.4 Product shorthand notations:

Every product has a shorthand representation of the entire product label. These shorthand notations can be found on user invoices, and users can search for them.

6.1.5 Logout:

Users can logout after doing the following required processes,and all session activities will be preserved in the database.

NON- FUNCTIONAL REQUIREMENTS:

In the insurance to the internet, the following Non-Functional

Requirements will be present:

(i)Secure access to a customer's personal information.

(ii)Availability 24 hours a day, 7 days a week.

(iii)Improved component design to improve peak performance.

(iv)For future expansion, a flexible service-based architecture will be particularly desired.

(v)System properties and limitations are defined by Non-Functional Requirements**:** **Security, Reliability, Maintainability, Portability, Extensibility, Reusability, Compatibility, and Resource Utilization** are some of the additional non-functional requirements.

METHODOLOGY & DESIGN:

Anyone who owns a department store and wants to utilise this WEB-BASED BILLING SOFTWARE must first sign up for the application. The owner can create a bill for current items or add new products using the product shorthand page and other operations after joining up with an authenticated email address and password. The administrator has the ability to monitor and contact the registered department store owners who have registered using this application.

The owner may then generate a bill for a customer by selecting available items, adding quantities, and printing the bill. The online application will compute the total cost of a bill submitted on a specific date.

If the customer's information has to be altered, the owner can go to the Manage Bills page and amend the information (involving edit, updating, delete details). Through the Giveaway generator, a giveaway is issued for a random number of consumers for each bill made.

Every bill created is maintained in the database and may be accessed at any time. The report generating page functions similarly, generating summary reports to the owner on a monthly, weekly, or yearly basis.

This online application will automate the billing and management of a department shop. This web-based programme is built on a network of department stores located around the country.

7.1 3-Tier Architecture:

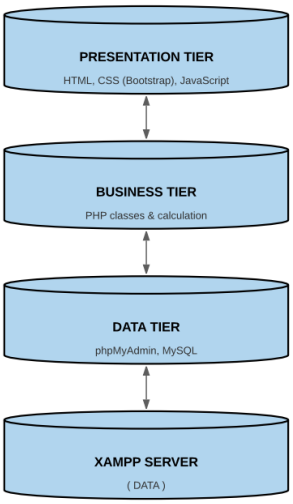


Fig 1: 3 Tier Architecture Diagram

7.2 Use case diagram:

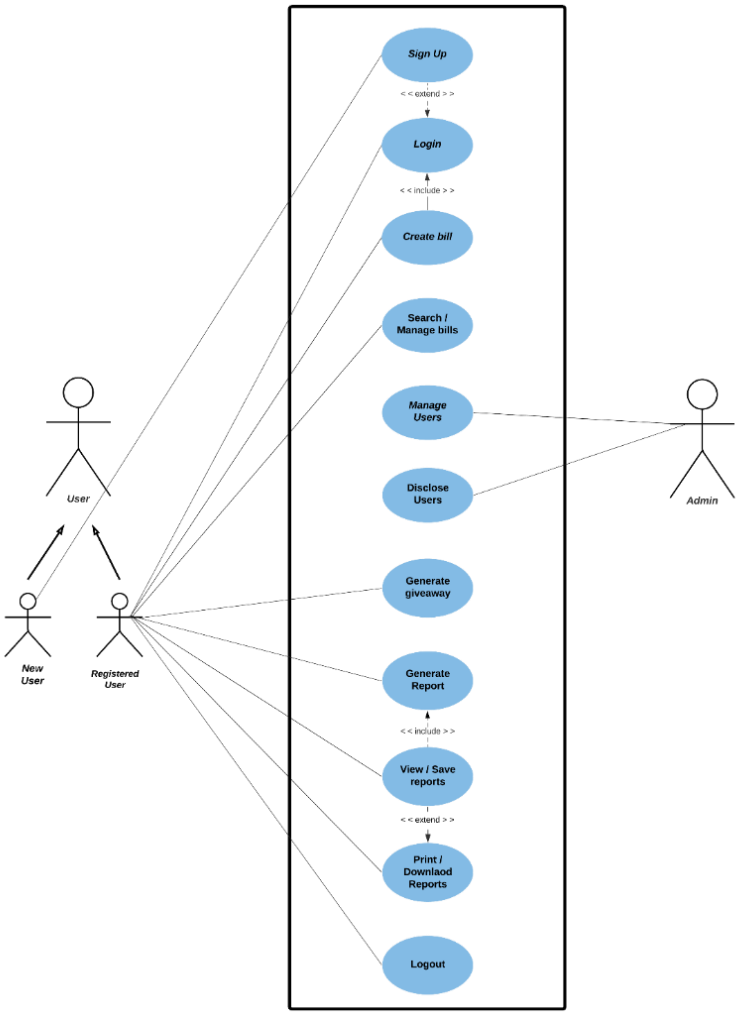


Fig2: USE CASE diagram

7.3: Class diagram:

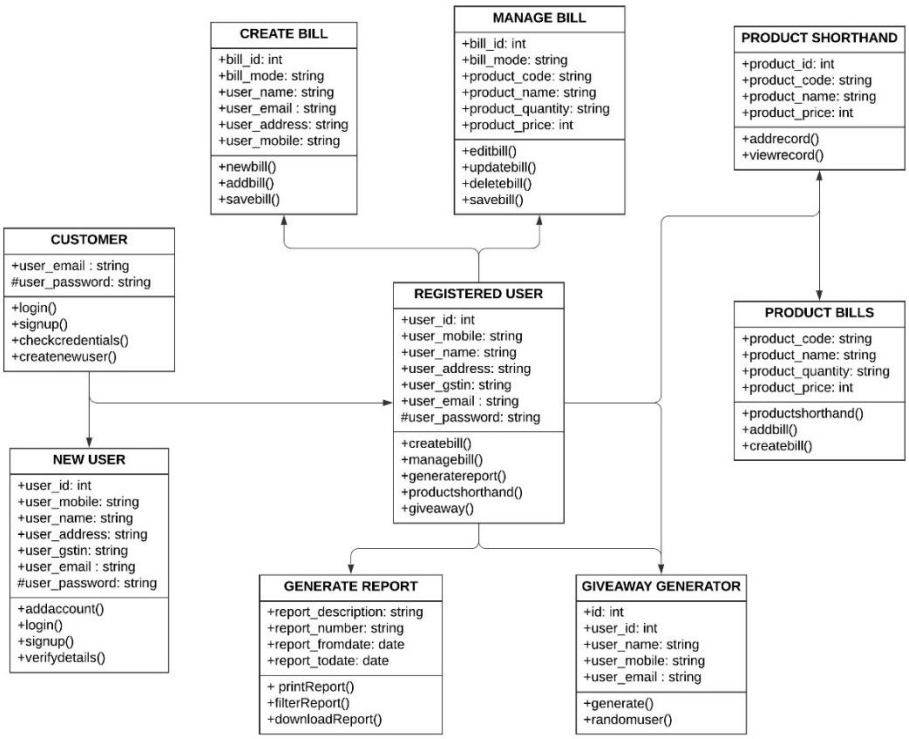


Fig3: CLASS diagram

7.4.Activity diagram:

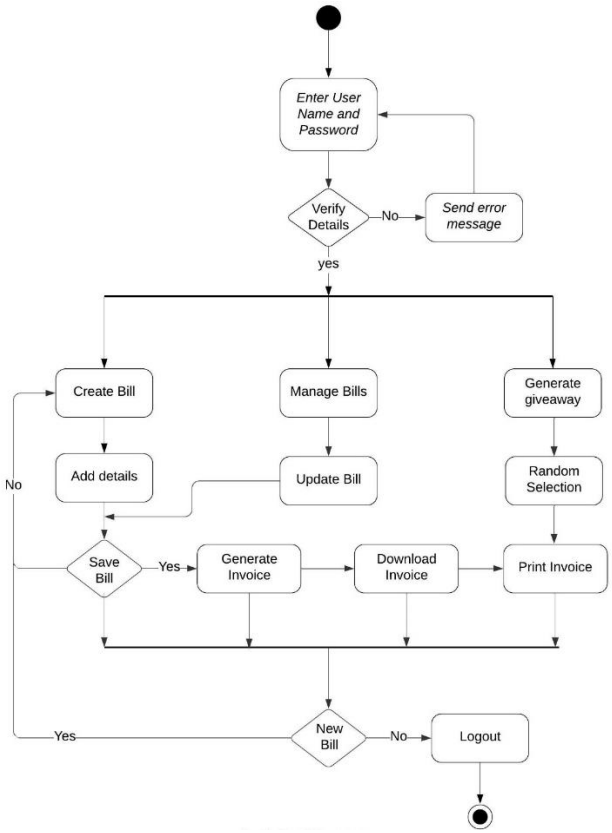


Fig4: ACTIVITY diagram

7.5.Sequence diagram:

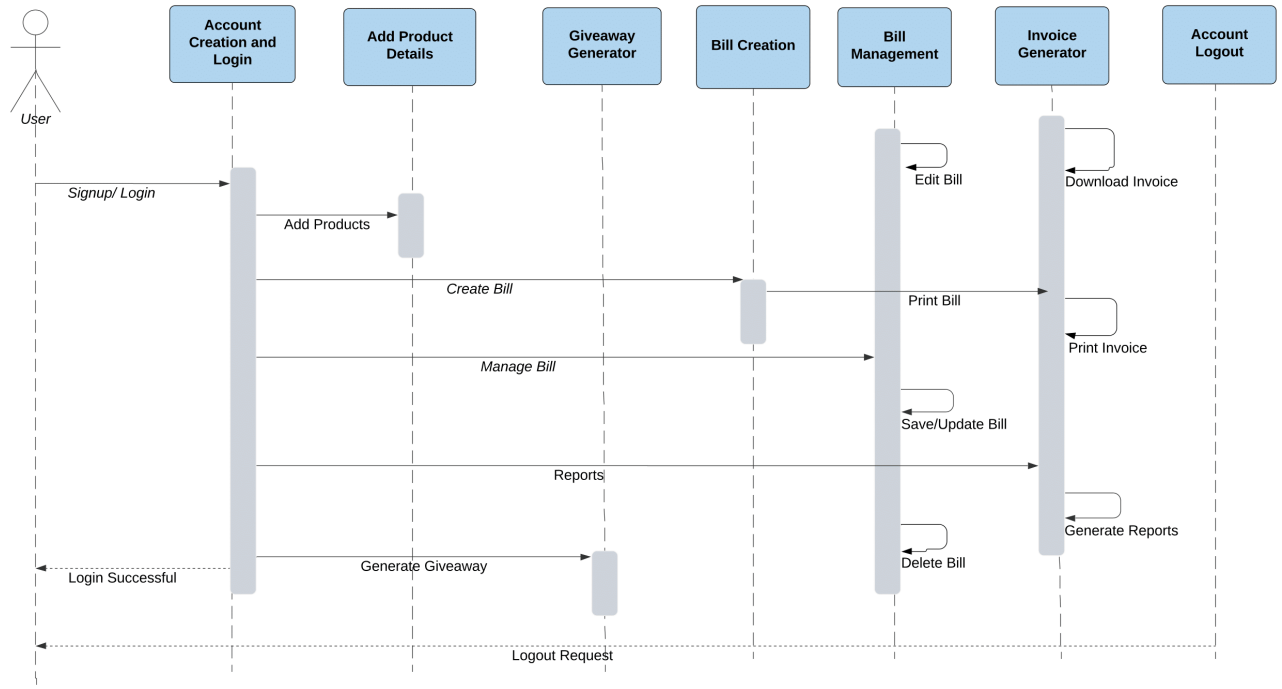


Fig5: SEQUENCE diagram

7.6. 1 level data flow diagram:

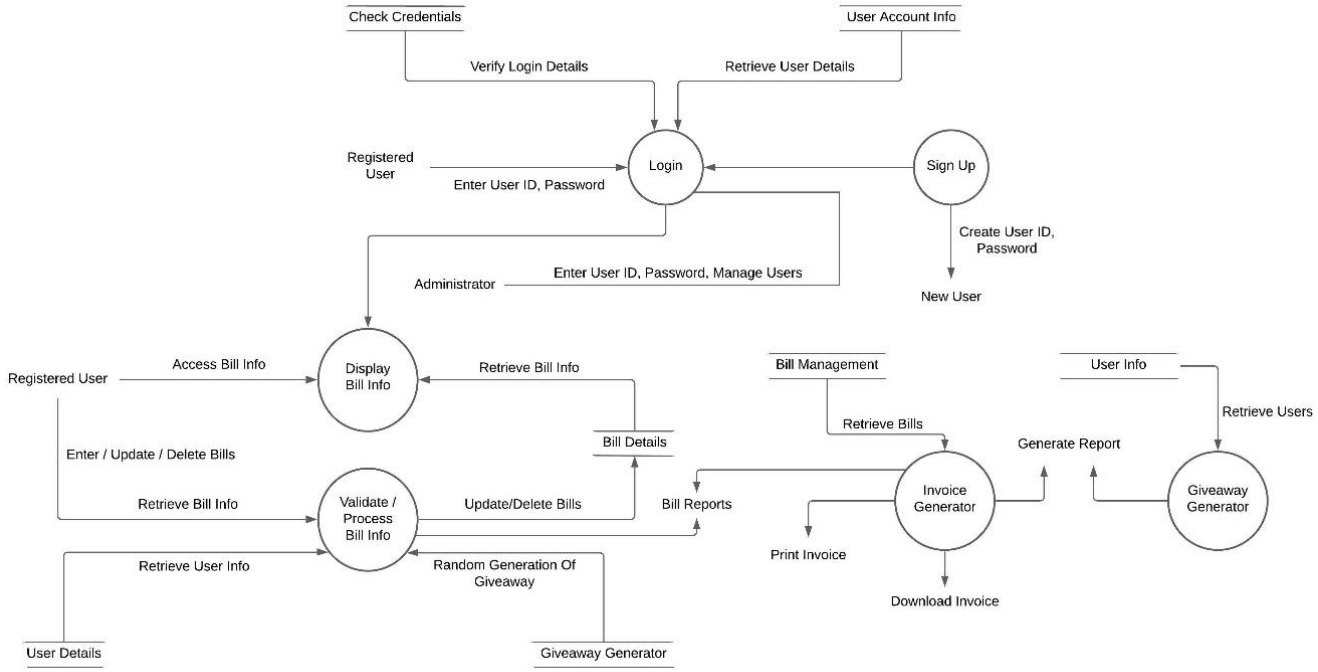


Fig6: 1 level data flow diagram

7.7. 2 level - Login data flow diagram:

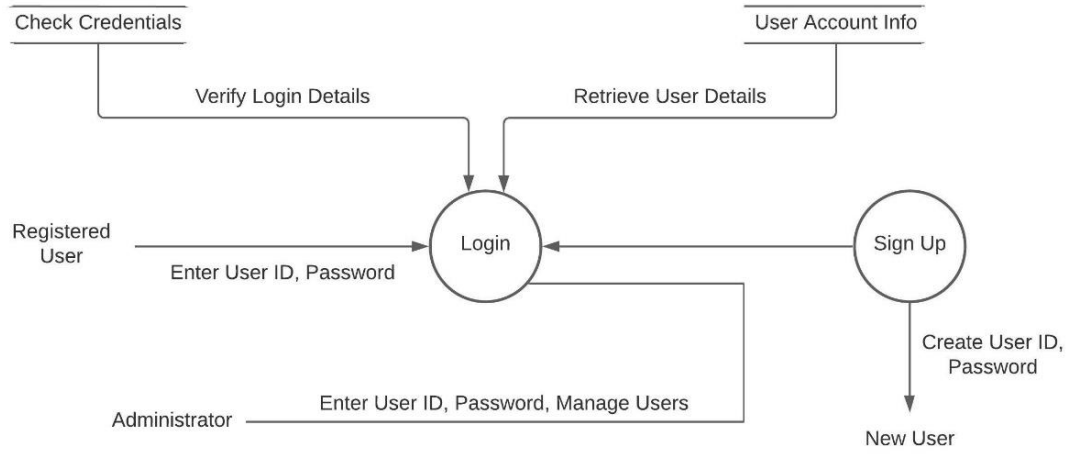


Fig7: 2 level LOGIN data flow diagram

7.8. 2 level- Bill Management data flow diagram:

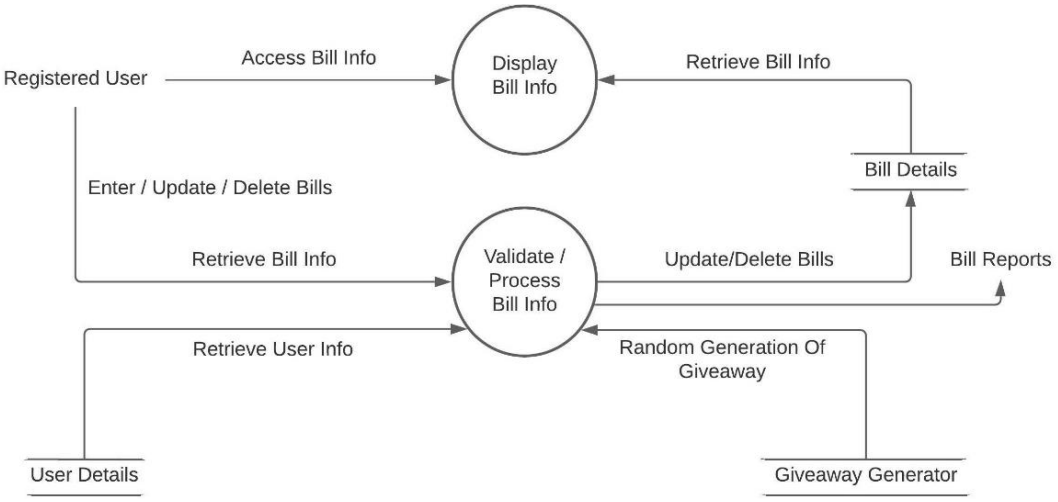


Fig8: 2 level BILL MANAGEMENT data flow diagram

7.9. 2 level- Report Give away data flow diagram

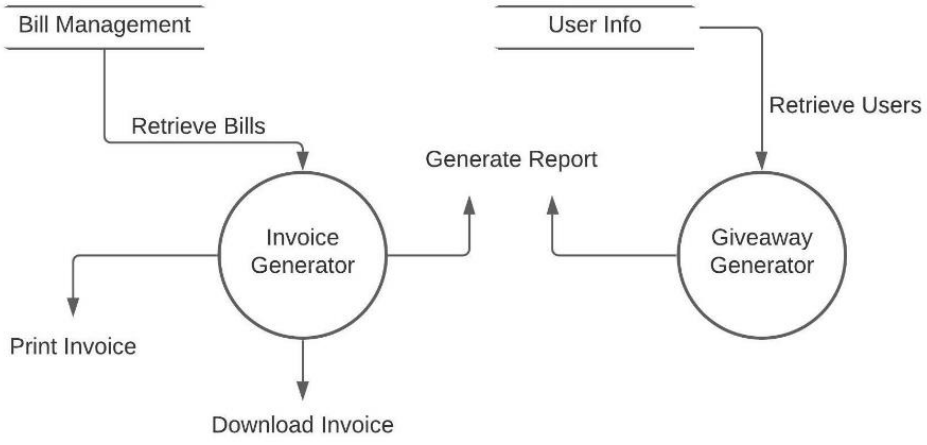


Fig9: 2 level REPORT GIVE AWAY data flow diagram

CODE SNAPSHOTS:

USER SIGN UP:

signup.php

```
<form method="post" action="" class="login">
    <header>SIGNUP</header>
    <div class="field">
        <span class="fa fa-user"></span>
        <input type="text" name="name" placeholder="Store Name" required>
    </div>
    <div class="field">
        <span class="fa fa-envelope"></span>
        <input type="email" name="email" placeholder="Email" required>
    </div>
    <div class="field">
      <span class="fa fa-phone"></span>
      <input type="phone" name="mob" placeholder="Mobile No." required>
    </div>
    <div class="field">
      <span class="fa fa-lock"></span>
      <input type="password" name="pass" placeholder="Password" required>
    </div>
    <div class="field">
      <span class="fa fa-vcard-o"></span>
      <input type="text" name="gst" placeholder="GSTIN / UIN" >
    </div>
    <input type="submit" class="submit" name="submit" value="CREATE ACCOUNT">

    <span class="logn-form-copy">Already have an account? <a href="login.php" class="login-form__sign-up">Login</a></span>
</form>
<?php
  if ($_SERVER["REQUEST_METHOD"] == "POST")
  {
      $name = $_POST['name'];
      $email=$_POST['email'];
      $mobile=$_POST['mob'];
      $pass=$_POST['pass'];
      $gst=$_POST['gst'];
      $query = "SELECT * FROM details WHERE email='$email'";
      $result = mysqli_query($conn, $query)or die($mysqli_error($conn));
      $num = mysqli_num_rows($result);
      if ($num!=0) {
        echo '<script>alert("Email Already exists")</script>';
      }else{
        $sql = "INSERT INTO details(name,email,mobile,password,gst) VALUES ('$name','$email','$mobile','$pass','$gst')";
        $result = mysqli_query($conn, $sql);
        header('location: login.html');
      }
  }
```

Fig10: User Signup

USER LOGIN:

login.php

```
<form method="post" action="" class="login">
    <header>LOGIN</header>
    <div class="field">
        <span class="fa fa-user"></span>
        <input type="email" name="email" placeholder="Email ID">
    </div>
    <div class="field">
      <span class="fa fa-lock"></span>
      <input type="password" name="pass" placeholder="Password">
    </div>
    <div class="forgot-password">
        <a href="#">Forgot password?</a>
    </div>
    <input type="submit" class="submit" value="LOGIN">
    <span class="logn-form-copy">Don't have an account? <a href="signup.php" class="login-form__sign-up">Sign up</a></span>
</form>
<?php
    if ($_SERVER["REQUEST_METHOD"] == "POST")
    {
        $email=$_POST['email'];
        $pass=$_POST['pass'];
        $query = "SELECT * FROM details WHERE email='$email' AND password='$pass'";
        $result = mysqli_query($conn, $query)or die($mysqli_error($conn));
        $num = mysqli_num_rows($result);
        if ($num==0) {
          echo '<script>alert("Invalid Credentials")</script>';
        } else {
          $row = mysqli_fetch_array($result);
          $_SESSION['email'] = $row['email'];
          $_SESSION['user_id'] = $row['id'];
          $_SESSION['name']=$row['name'];
          header('location: index.php');
        }
    }
?>
```

Fig11: User Login

USER LOGOUT:

logout.php

```php
<?php
  require("connect.php");
  session_unset();
  session_destroy();
  header('location: login.php');
?>
```

Fig12: User Logout

USER OPERATIONS:

```php
<!-- Vertical navbar -->
<div class="vertical-nav bg-white" id="sidebar">
  <div class="py-4 px-3 mb-4 bg-light">
    <div class="media d-flex align-items-center">
      <img loading="lazy" src="pic.jpg" alt="..." width="80" height="80" class="mr-3 rounded-circle img-thumbnail shadow-sm">
      <div class="media-body">
        <h4 class="m-0"><?php echo $_SESSION['name']; ?></h4>

      </div>
    </div>
  </div>

  <ul class="nav flex-column bg-white mb-0">
    <li class="nav-item">
      <a href="index.php" class="nav-link text-dark">
                <i class="fa fa-home mr-3 text-dark fa-fw"></i>
                Home
              </a>
    </li>
  </ul>
  <br>
<p class="text-dark font-weight-bold text-uppercase px-3 small pb-4 mb-0">Operations</p>

<ul class="nav flex-column bg-white mb-0">
  <li class="nav-item">
    <a href="createbill.php" class="nav-link text-dark bg-light">
              <i class="fa fa-pencil-square-o mr-3 text-dark fa-fw"></i>
              Create Bill
            </a>
  </li>
  <li class="nav-item">
    <a href="managebill.php" class="nav-link text-dark">
              <i class="fa fa-bar-chart mr-3 text-dark fa-fw"></i>
              Manage Bill
            </a>
  </li>

  <li class="nav-item">
    <a href="generatereport.php" class="nav-link text-dark">
              <i class="fa fa-window-restore mr-3 text-dark fa-fw"></i>
              Generate Report
            </a>
  </li>
<li class="nav-item">
  <a href="productshorthand.php" class="nav-link text-dark">
        <i class="fa fa-th-list mr-3 text-dark fa-fw"></i>
        Product Shorthand
      </a>
  </li>
  <li class="nav-item">
  <a href="giveawaygenerator.php" class="nav-link text-dark">
        <i class="fa fa-gift mr-3 text-dark fa-fw"></i>
        Giveaway Generator
      </a>
  </li>
</ul>

<div class="mt-5 text-center">
<button type="button" class="btn btn-dark"><a href="logout.php" style="color:white;text-decoration: none;">Logout</a></button>
```

Fig13,14: User Operations

CREATE BILL:

```php
<?php
  if (isset($_POST['addbill']))
  {
     $userid=$_SESSION['user_id'];
      $billid = $_SESSION['billno'];
      $mode=$_POST['mode'];
      $_SESSION['mode']=$mode;

      $cust_name=$_POST['name'];
      $_SESSION['cust_name']=$cust_name;

      $cust_addr=$_POST['addr'];
      $_SESSION['cust_addr']=$cust_addr;

      $cust_mob=$_POST['mob'];
      $_SESSION['cust_mob']=$cust_mob;

      $cust_email=$_POST['email'];
      $_SESSION['cust_email']=$cust_email;

      $prod_code=$_POST['pcode'];
      $prod_quantity=$_POST['quantity'];
      $prod_price=$_POST['price'];
      $query = "INSERT INTO bills(user_id,bill_id,mode,cust_name,cust_addr,cust_mob,cust_email,prod_code,prod_quantity,prod_price) VALUES ('$userid','$billi
      $result = mysqli_query($conn, $query)or die($mysqli_error($conn));
      echo("<meta http-equiv='refresh' content='1'>");
  }
  $user=$_SESSION['user_id'];
      $sql = "SELECT * FROM bills WHERE user_id='$user' AND bill_id='$val1'";
      $result = mysqli_query($conn, $sql);
      $sql1="SELECT SUM(prod_quantity*prod_price) AS TOTAL FROM bills WHERE user_id='$user' AND bill_id='$val1'";
      $result1=mysqli_query($conn, $sql1);
      $row1 = mysqli_fetch_assoc($result1);
      echo "<div class='container-fluid'><hr style='border-top: 1px dashed rgb(255, 255, 255);'><center><table class='table'><tr><th>ID</th><th>Item Name</t
              $count=1;
              while($row = mysqli_fetch_assoc($result))
              {
                  echo "<tr><td>".$count."</td><td>".$row['prod_code']."</td><td>".$row['prod_quantity']."</td><td>".$row['prod_price']."</td><td>".$row['pr
                  $count++;
              }
              echo "<tr><td></td><td></td><td></td><td></td><td></td><td><b>Total: RS ".$row1['TOTAL']."</b></td></tr>";
              echo "</table><center></div>";
?>
      $prod_code=$_POST['pcode'];
      $prod_quantity=$_POST['quantity'];
      $prod_price=$_POST['price'];
      $query = "INSERT INTO bills(user_id,bill_id,mode,cust_name,cust_addr,cust_mob,cust_email,prod_code,prod_quantity,prod_price) VALUES ('$userid','$billi
      $result = mysqli_query($conn, $query)or die($mysqli_error($conn));
      echo("<meta http-equiv='refresh' content='1'>");
  }
  $user=$_SESSION['user_id'];
      $sql = "SELECT * FROM bills WHERE user_id='$user' AND bill_id='$val1'";
      $result = mysqli_query($conn, $sql);
      $sql1="SELECT SUM(prod_quantity*prod_price) AS TOTAL FROM bills WHERE user_id='$user' AND bill_id='$val1'";
      $result1=mysqli_query($conn, $sql1);
      $row1 = mysqli_fetch_assoc($result1);
      echo "<div class='container-fluid'><hr style='border-top: 1px dashed rgb(255, 255, 255);'><center><table class='table'><tr><th>ID</th><th>Item Name</t
              $count=1;
              while($row = mysqli_fetch_assoc($result))
              {
                  echo "<tr><td>".$count."</td><td>".$row['prod_code']."</td><td>".$row['prod_quantity']."</td><td>".$row['prod_price']."</td><td>".$row['pr
                  $count++;
              }
              echo "<tr><td></td><td></td><td></td><td></td><td></td><td><b>Total: RS ".$row1['TOTAL']."</b></td></tr>";
              echo "</table><center></div>";
?>
      </div> <br>
                      <div style="display: flex; justify-content:center" class="text-center">
                          <form action="savebill.php" method="POST">
                      <button type="submit" name="savebill" class="btn btn-light">Save Bill</button>
                      </form>
                      </div>
```

Fig15,16: Create Bill

GENERATE REPORT:

```php
<?php
    $con = mysqli_connect("localhost","root","","billing");

    if(isset($_POST['from_date']) && isset($_POST['to_date']))
    {
        $from_date = $_POST['from_date'];
        $_SESSION['from_date'] = $_POST['from_date'];
        $to_date = $_POST['to_date'];
        $_SESSION['to_date'] = $_POST['to_date'];
        $sesid=$_SESSION['user_id'];

        $query = "SELECT bill_id, mode, cust_name, cust_mob, prod_quantity, prod_price*prod_quantity AS totalcost FROM
        bills WHERE user_id='$sesid' AND date BETWEEN '$from_date' AND '$to_date' GROUP BY cust_mob";
        $query_run = mysqli_query($con, $query);
        $countervar = 1;

        if(mysqli_num_rows($query_run) > 0)
        {
            foreach($query_run as $row)
            {
                ?>
                <tr>
                    <td><?php echo $countervar; ?></td>
                    <td><?= $row['bill_id']; ?></td>
                    <td><?= $row['mode']; ?></td>
                    <td><?= $row['cust_name']; ?></td>
                    <td><?= $row['cust_mob']; ?></td>
                    <td><?= $row['prod_quantity']; ?></td>
                    <td><?= $row['totalcost']; ?></td>
                </tr>
                <?php
                $countervar++;
            }
        }
        else
        {
            echo "<span class='text-white'>"."No Record Found"."</span> <br>";
        }
    }
?>
    </tbody>
</table> <br>
<div style="display: flex; justify-content:center" class="text-center">
    <form action="reportaspdf.php" method="POST">
```

reportaspdf.php > ...

```php
<?php
    include 'connect.php';
    require_once('./tcpdf/tcpdf.php');
    $obj_pdf = new TCPDF('P', PDF_UNIT, PDF_PAGE_FORMAT, true, 'UTF-8', false);
    $obj_pdf->SetCreator(PDF_CREATOR);
    $obj_pdf->SetTitle("YOUR BILL REPORT");
    $obj_pdf->SetHeaderData('', '', PDF_HEADER_TITLE, PDF_HEADER_STRING);
    $obj_pdf->setHeaderFont(array(PDF_FONT_NAME_MAIN, '', PDF_FONT_SIZE_MAIN));
    $obj_pdf->setFooterFont(array(PDF_FONT_NAME_DATA, '', PDF_FONT_SIZE_DATA));
    $obj_pdf->setDefaultMonospacedFont('helvetica');
    $obj_pdf->SetFooterMargin(PDF_MARGIN_FOOTER);

    $obj_pdf->SetMargins(PDF_MARGIN_LEFT,'0',PDF_MARGIN_RIGHT);
    $obj_pdf->setPrintHeader(false);
    $obj_pdf->setPrintFooter(false);
    $obj_pdf->SetAutoPageBreak(TRUE, 10);
    $obj_pdf->SetFont('helvetica', '', 12);
    $obj_pdf->setPrintHeader(false);
    $obj_pdf->AddPage();

    $content = "";
    $content .= "
    <center><h3>Your Bill Report</h3></center><br><hr><br><br>
    <table class='table text-white table-border'>
        <thead>
            <tr>
                <th>Id</th>
                <th>Bill Id</th>
                <th>Mode</th>
                <th>Name</th>
                <th>Mobile</th>
                <th>Quantity</th>
                <th>Total Cost</th>
            </tr>
        </thead>
        <tbody>";
```

Fig17,18: GENERATE REPORT

GENERATE GIVEAWAY:

```
        <div class="form-group row d-flex justify-content-center">
          <div class="btn-toolbar mr-3 ml-4" role="toolbar" aria-label="Toolbar with button groups">
            <div class="btn-group " role="group" aria-label="First group">
              <button type="submit" name="submit" class="btn btn-light"><b>Generate</b></button>
            </div>
          </div>
        </div>
        </div>
</form>
<?php
  if ($_SERVER["REQUEST_METHOD"] == "POST")
  {
      $id=$_SESSION['user_id'];
      $n=$_POST['rec'];
      $sql = "SELECT DISTINCT(cust_name),cust_mob,cust_email FROM bills WHERE user_id='$id' ORDER BY RAND() LIMIT ".$n;
      $result = mysqli_query($conn, $sql);
      echo "<div class='container'><center><table class='table'><tr>
      <th>ID</th><th>Name</th><th>Mobile</th><th>Email</th></tr>";
                $count=1;
                while($row = mysqli_fetch_assoc($result))
                {
                    echo "<tr><td>".$count."</td><td>".$row['cust_name']."</td><td>".$row['cust_mob'].
                    "</td><td>".$row['cust_email']."</td></tr>";
                    $count++;
                }
                echo "</table><center></div>";
  }
?>
  </div>
```

Fig19: Generate Giveaway

PRODUCT SHORTHAND:

```
<div class="col-sm-5">
  <?php
  $value=$_SESSION['user_id'];
      $sql="SELECT * FROM products WHERE user_id='$value'";
      $result = mysqli_query($conn,$sql);
      $num_rows= mysqli_num_rows($result);
      if ($num_rows>= 1) {
          echo "<center><table class='table'><tr><th>ID</th><th>Code</th><th>Item Name</th><th>Price</th></tr>";
          $count=1;
          while($num_rows!=0)
          {
              $row = mysqli_fetch_assoc($result);
              echo "<tr><td>".$count."</td><td>".$row['code']."</td><td>".$row['prod_name']."</td><td>".$row['price']."</td></tr>";
              $num_rows--;
              $count++;
          }
          echo "</table><center>";
      }else{
          echo "<center><h3>Database is empty!!<h3></center>";
      }
  ?>
    </div>
  </div>
</div>
```

Fig19: Generate Giveaway

RETRIEVE BILLS:

```
retrieve.php > ...
1   <?php
2       include 'connect.php';
3       if (!isset($_SESSION['email'])) {
4         header('location: login.php');
5       }
6       if(isset($_GET['codeval'])){
7           $item=$_GET['codeval'];
8           $query = "SELECT prod_name FROM products WHERE user_id=".$_SESSION['user_id']." AND code='$item'";
9           $result = mysqli_query($conn, $query)or die($mysqli_error($conn));
10          while($row = mysqli_fetch_assoc($result))
11          {
12              echo $row['prod_name'];
13          }
14      }
15      if(isset($_GET['priceval'])){
16          $item=$_GET['priceval'];
17          $query = "SELECT price FROM products WHERE user_id=".$_SESSION['user_id']." AND code='$item'";
18          $result = mysqli_query($conn, $query)or die($mysqli_error($conn));
19          while($row = mysqli_fetch_assoc($result))
20          {
21              echo $row['price'];
22          }
23      }
24  ?>
```

Fig20: Retrieve Bills

CONNECTION PHP CODE:

connect.php > ...

```php
<?php
$conn = mysqli_connect("127.0.0.1", "root", "", "billing")or die($mysqli_error($conn));
if(!isset($_SESSION))
{
    session_start();
}
?>
```

Fig21: Connection PHP code

REQUIRED CSS:

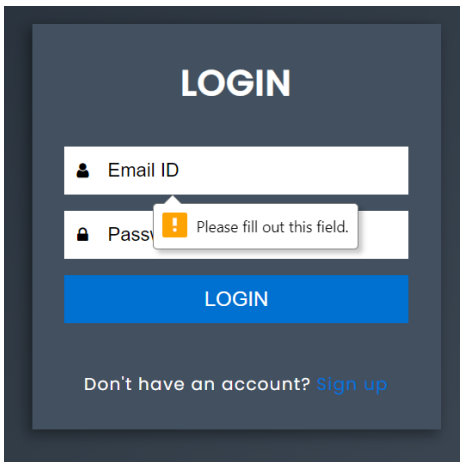
```css
body {
  background-color: #485461;
  background-image: linear-gradient(315deg, #485461 0%, #28313b 74%);
  min-height: 100vh;
  overflow-x: hidden;
}
.vertical-nav {
  min-width: 17rem;
  width: 17rem;
  height: 100vh;
  position: fixed;
  top: 0;
  left: 0;
  box-shadow: 3px 3px 10px rgba(0, 0, 0, 0.1);
  transition: all 0.4s;
}
.page-content {
  width: calc(100% - 17rem);
  margin-left: 17rem;
  transition: all 0.4s;
}

#sidebar.active {
  margin-left: -17rem;
}

#content.active {
  width: 100%;
  margin: 0;
}

.separator {
  margin: 3rem 0;
  border-bottom: 1px dashed #fff;
}

.text-uppercase {
  letter-spacing: 0.1em;
}
.text-gray {
  color: #aaa;
}
```


```css
body {
  background-color: #485461;
  background-image: linear-gradient(315deg, #485461 0%, #28313b 74%);
  min-height: 100vh;
  overflow-x: hidden;
}
.vertical-nav {
  min-width: 17rem;
  width: 17rem;
  height: 100vh;
  position: fixed;
  top: 0;
  left: 0;
  box-shadow: 3px 3px 10px rgba(0, 0, 0, 0.1);
  transition: all 0.4s;
}
.page-content {
  width: calc(100% - 17rem);
  margin-left: 17rem;
  transition: all 0.4s;
}

#sidebar.active {
  margin-left: -17rem;
}

#content.active {
  width: 100%;
  margin: 0;
}

.separator {
  margin: 3rem 0;
  border-bottom: 1px dashed #fff;
}

.text-uppercase {
  letter-spacing: 0.1em;
}
.text-gray {
  color: #aaa;
}
```


Fig22: Required CSS

TESTING:

1. Testing Login Page, The user credentials are properly validated. Expected output is matched with actual output.



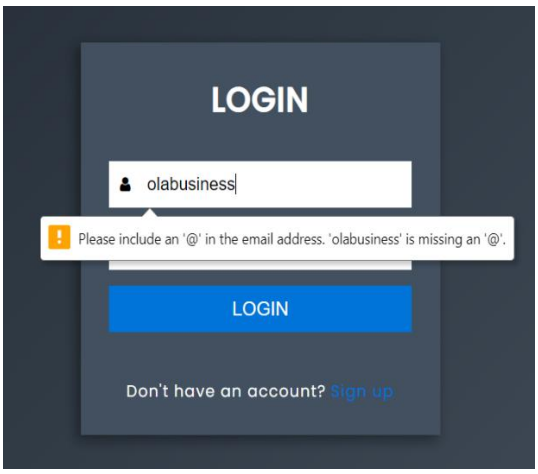


Fig23,24: Testing Login Page

2. When a user credentials and actual credentials don't match, handling the error and displaying with proper error message.

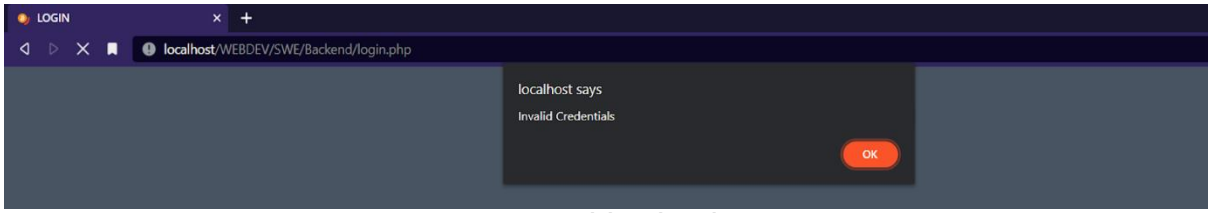


Fig25: Invalid Credentials

3. When an empty product details are added to the product list, It is not being handled properly. Expected output is not matching with actual output.

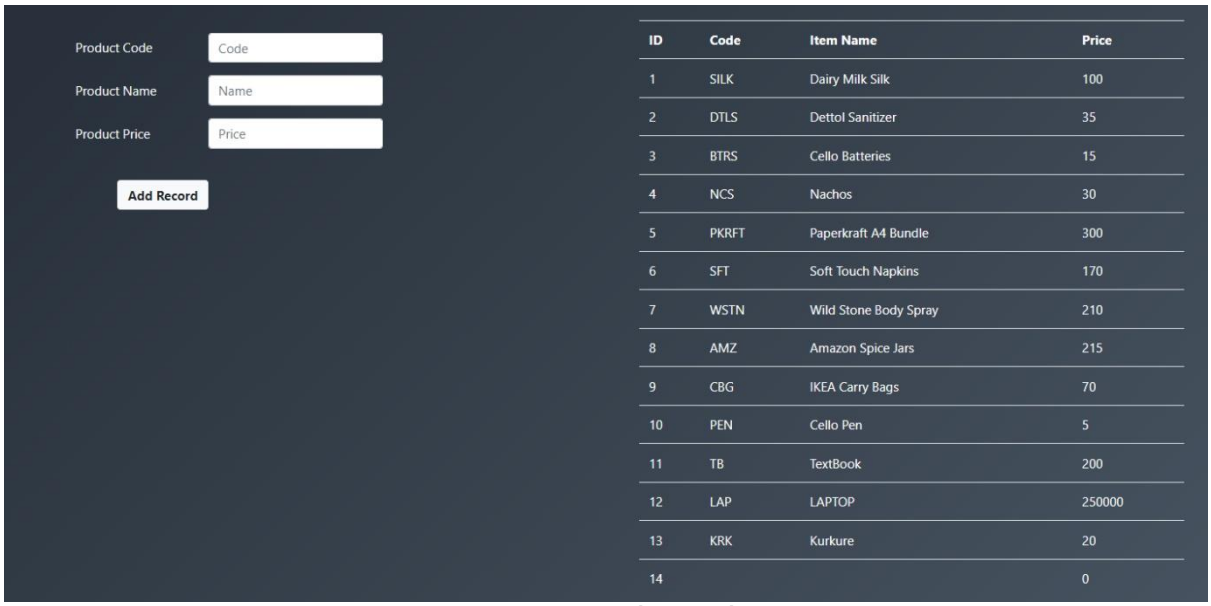


Fig26: Empty Product Details

4. Billing for an empty product details. It is not being handled properly. Expected output is not matching with actual output.

Fig27: Empty Product Details

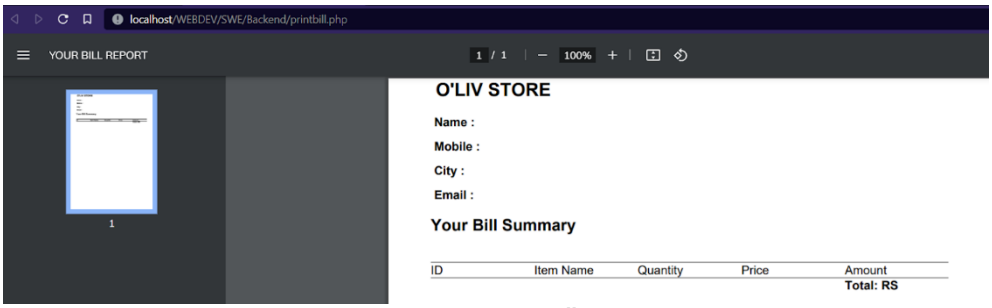


Fig28: Empty Bill

5. The list of random user records that is generated for no. of records is accurate. Expected output matches with actual output.

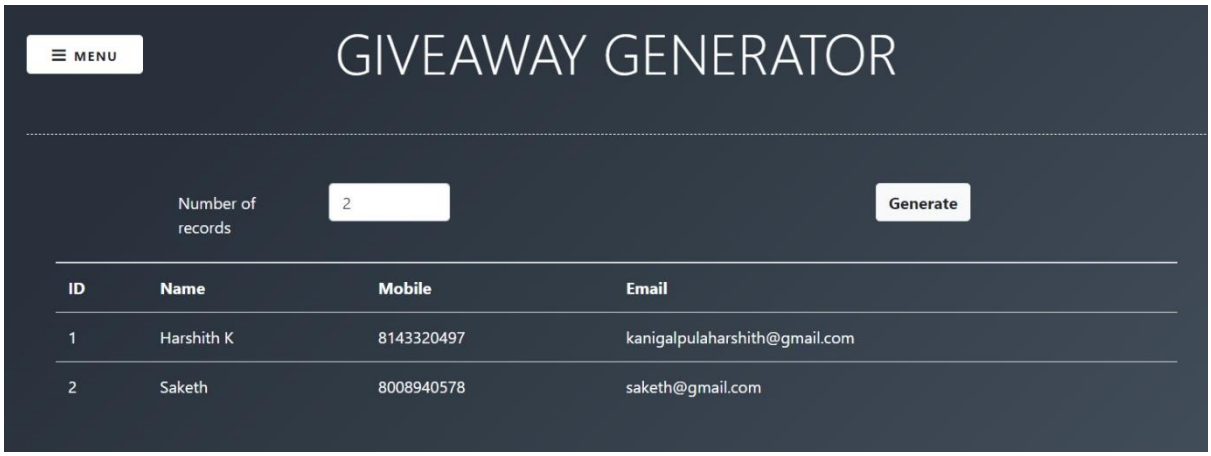


Fig29: Giveaway generator

6. The list of random user records that is generated for no. of records entered is not validated. Expected output is not matching with actual output.

Fig30: Giveaway generator

7. A working modal (popup) for Manage bills - Edit button. Expected Output matches with actual output.

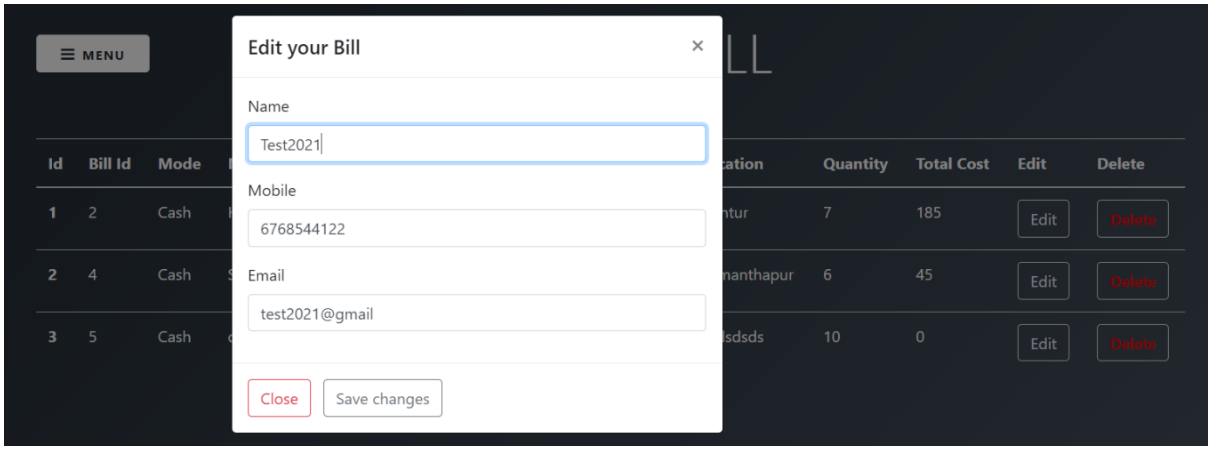


Fig31: Giveaway generator

RESULTS DISCUSSION:

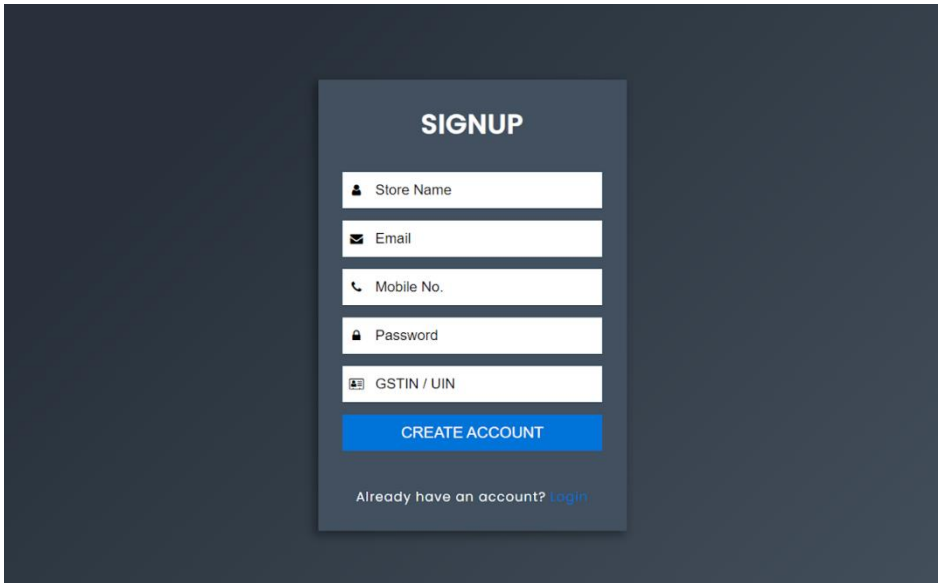


Fig32: Signup Page

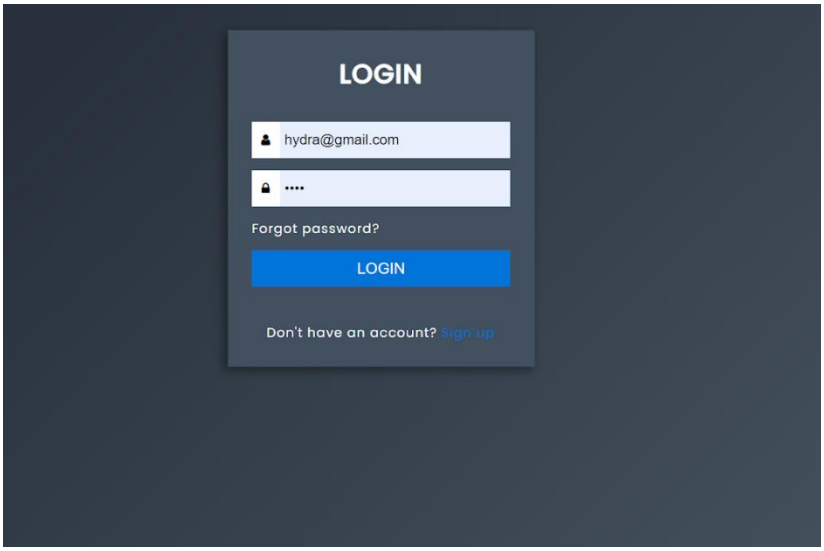


Fig33: Login Page

Fig34: Homepage

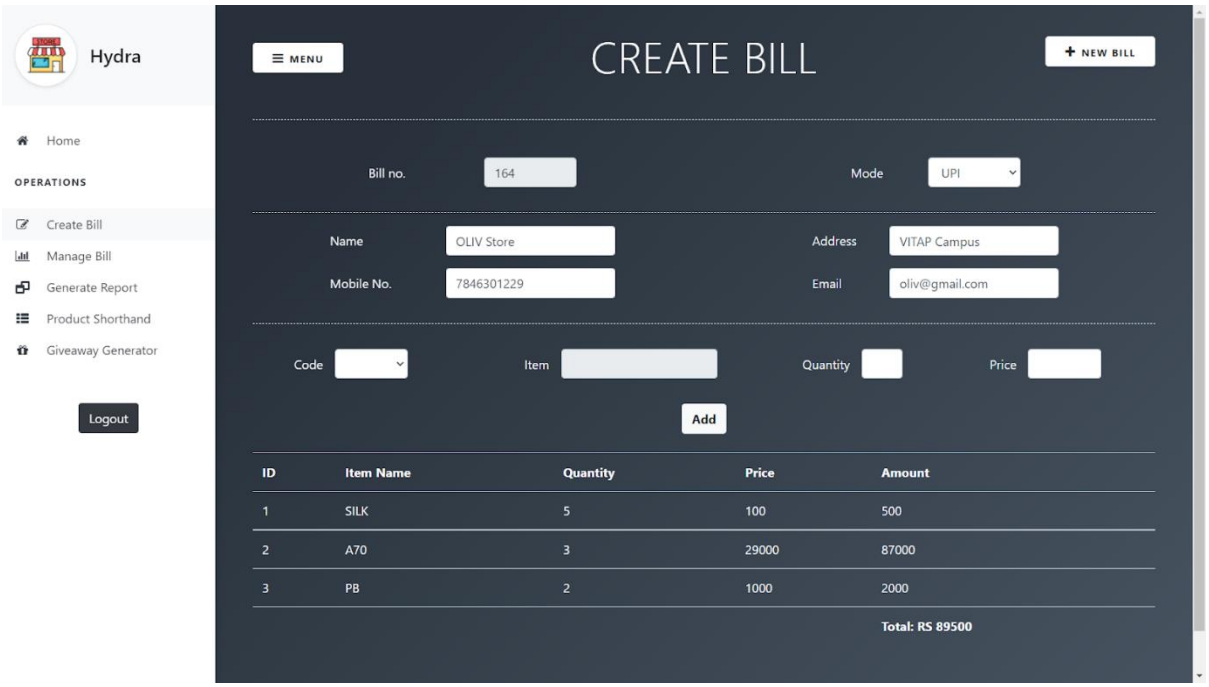


Fig35: Create Bill Page

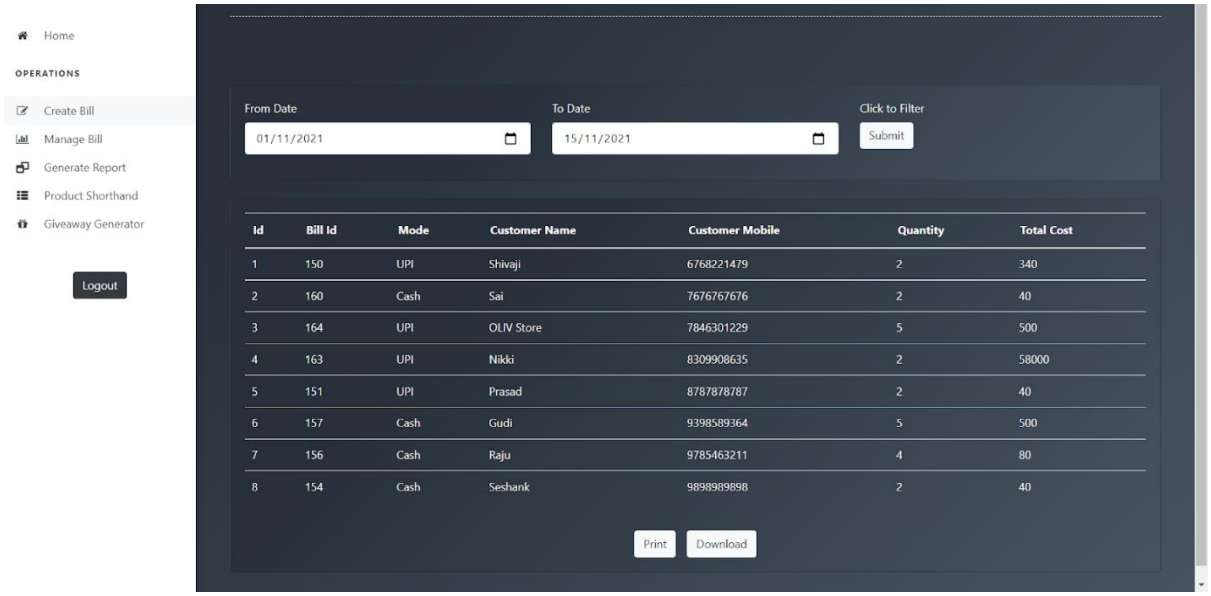


Fig36: Manage Bill Page

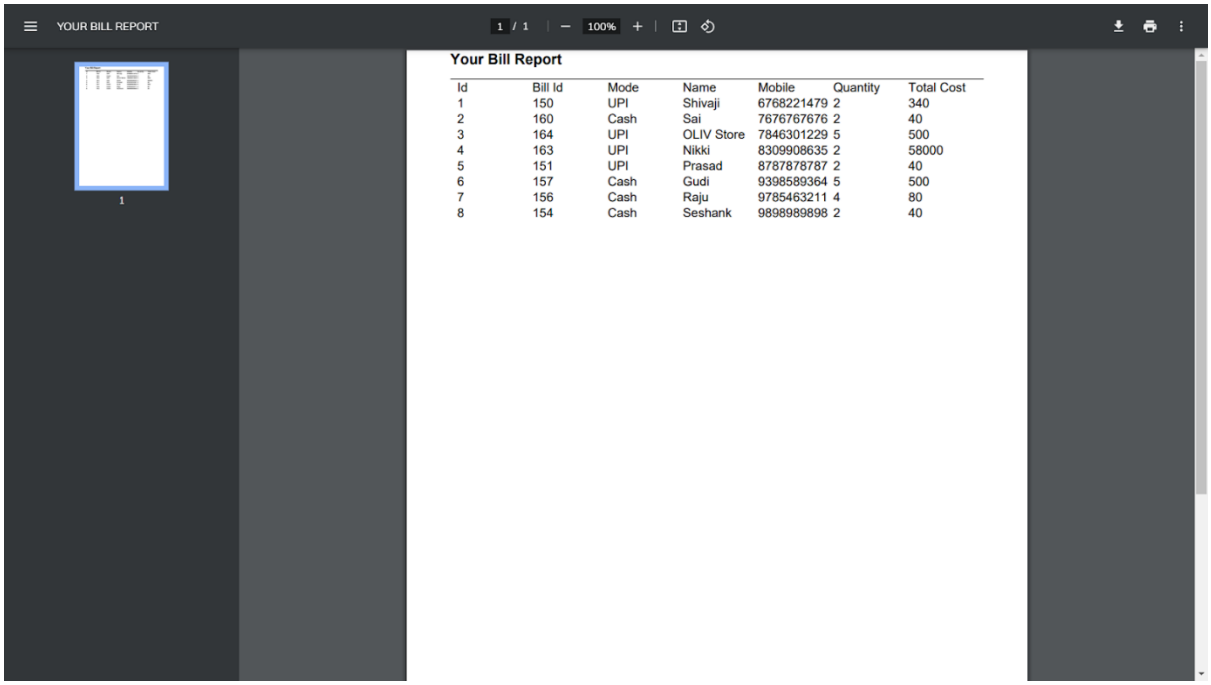


Fig37: Report Generation page for print

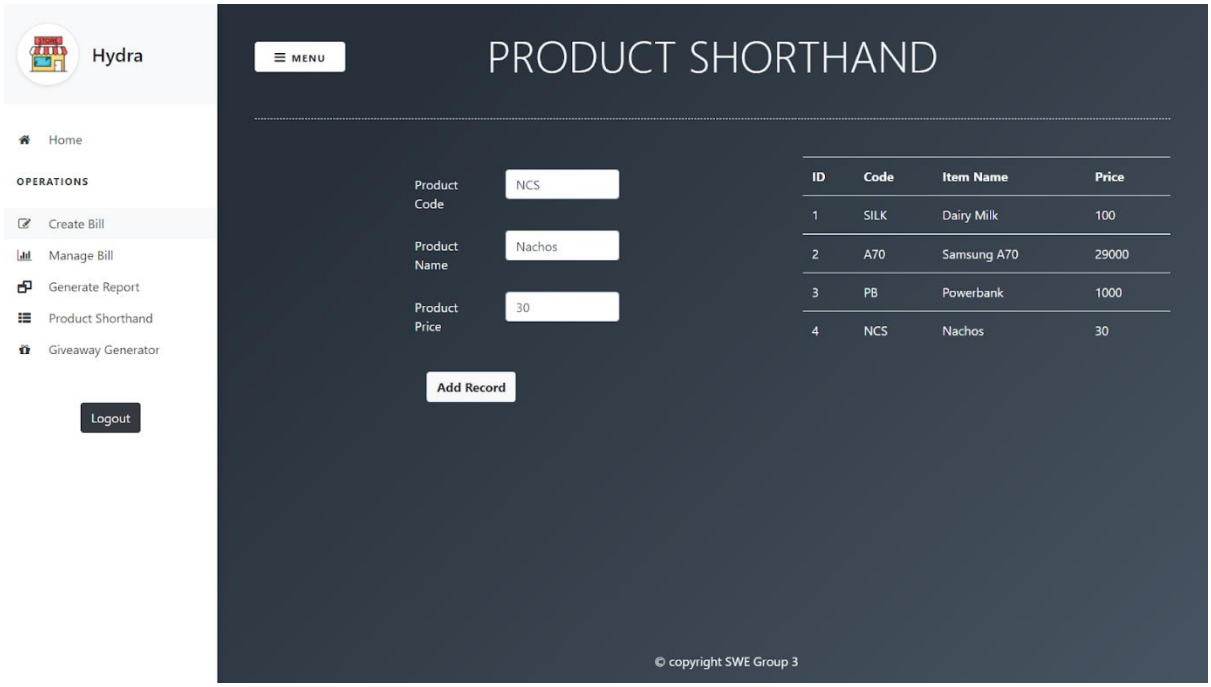


Fig38: Product ShortHand

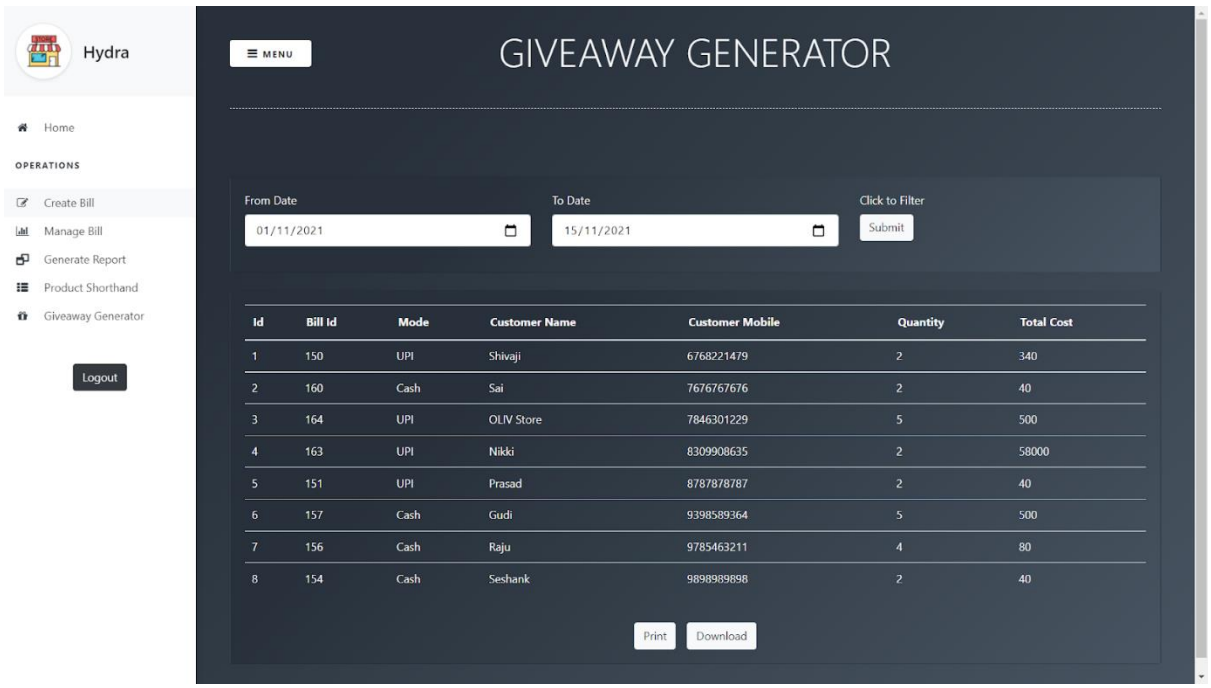


Fig39: Giveaway Generator

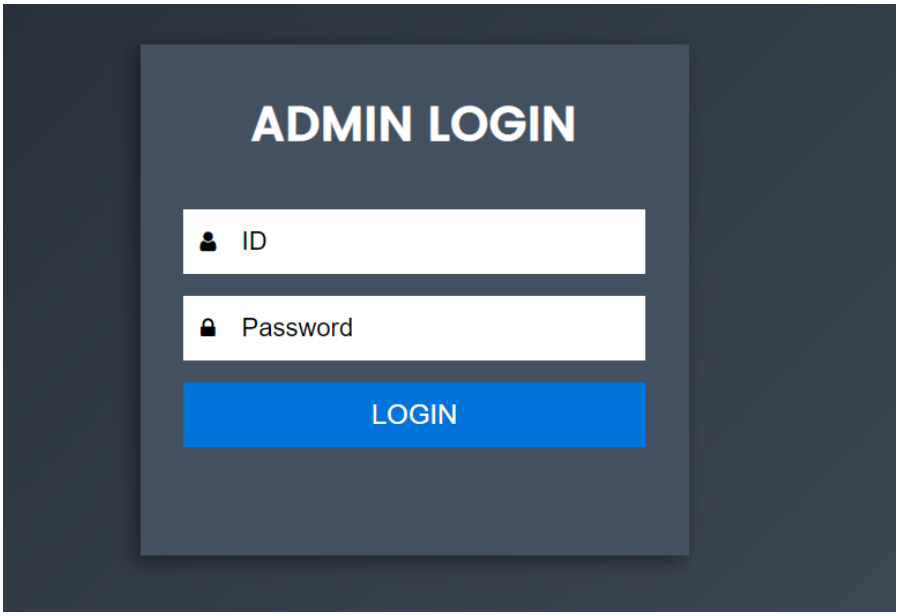


Fig40: Admin login page

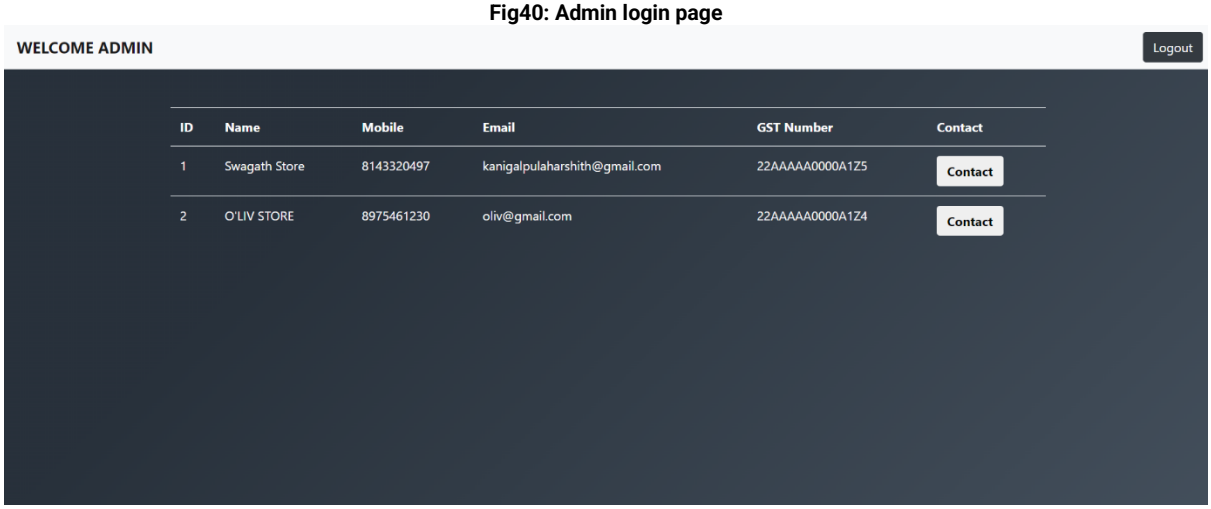


Fig41: Admin page

FUTURE SCOPE:

Though our project is itself matured enough but still betterment is always an open door. In this case, also we can add some features to this project in future to make this software more reliable. Such as

- Showing billing summary, profits/ loss with a visual interface by using Plots or graphs
- Sending their billing details, upcoming offers of that particular store as SMS to customer's registered mobile number.
- Additional details of each store in Admin Module and options to manage each store through admin module.

REFERENCES:

[1] Software Engineering: A Practitioner's Approach by Roger S. Pressman.
[2] Importance of billing software for small businesses.
[3] E-Billing Management System UML Diagram.
[4] How to Write a Software Requirements Specification (SRS Document).
[5] W3Schools - PHP Tutorial.
[6] techBriefers - How To Create PDF In PHP With TCPDF.

THANK YOU