A

PROJECT REPORT

On

E Banking Solution

of

Submitted in partial fulfillment of their requirements for the award of the Three-Year Diploma in

Information Technology

Under the supervision of

Er. Brijesh Kumar Mishra

(IT-Manager)

Softpro India Computer Technologies (P) Ltd.

Lucknow (UP)

Submitted By: -

Akshat

Srivastava

Submitted To: -

Feroze Gandhi Polytechnic

PREFACE

"Necessity is Mother of All Inventions"

Summer training is an important part of the engineering curriculum. The Diploma course summer training helps a student in getting acquainted with the manner in which his/her knowledge is being practically used outside his/her institute and this is normally different from what he/she has learnt from books. Hence, when the student switches from the process of learning to that of implementing his/her knowledge, he/she finds an abrupt change. This is exactly why summer training session during the B.E curriculum becomes all the more important. Summer training is prescribed for the student of Technical College as a part of the four-year degree course of engineering by the AICET. We are required to undergo summer training for a period of 45 days after the completion of the 2nd year.

This training report describes in detail the training after the 3rd year session, which I completed at the ***Softpro India Computer Technology (P) Ltd****. . This report also gives the information about the organization and it's working along with the project undertaken in the training period.*

The fundamental step used in ***SDLC*** *process is based on the ISO 9001 guidelines. My aim was to follow the ISO guidelines and develop a perfect system.*

The system development was organized into 5 major parts:

1. ***Requirement Gathering***
2. ***Documentation/Design***
3. ***Development***
4. ***Coding***
5. ***Testing***

ACKNOWLEDGEMENT

I would like to express my deep and sincere gratitude to my supervisor Er. ***BRIJESH KUMAR MISHRA (****Softpro India Computer Technologies (P) Ltd.)****, who gave me his full support and encouraged me to work in an innovative and challenging project for Educational field. His wide knowledge and logical thinking gave me right direction all the time.*

I am deeply grateful my project coordinator for his/her help and support provided at every step of the project. Last but not the least, I thank to all employees of ***Softpro India Computer Technologies (P) Ltd****. for their support and co-operation.*

Akshat Srivastava

COMPLETION CERTIFICATE

This is to certify that Mr./Ms. ***Akshat Srivastava (Information Technology)*** *from* ***College Name*** *(Institute/University) was working on the project entitled "E Banking Solution" in Softpro India Computer Technologies Pvt. Ltd. She/he was engaged with us during* ***15 August*** *to* ***30 September*** *for a period of* ***45 days****.*

She/he has done an excellent job during his/her engagement with the Software Development & Testing Division of the company. She/he has completed his/her project during the training tenure. His/her performance has been good and satisfactory.

I would like to take this opportunity to express my appreciation to Mr./Ms. Akshat Srivastava for his/her work and wish him/her all the very best for his/her future endeavors.

Regards,

Ms. Yashi Asthana

CEO

Softpro India Computer Technologies Pvt. Ltd.

Lucknow (U.P.)

DECLARATION

This is to certify that the project report entitled ***"E Banking Solution"*** *is done by me is an authentic work carried out for the partial fulfillment of the requirements for the award of the Diploma in* ***"(Branch Name)"*** *under the guidance of Er.* ***BRIJESH KUMAR MISHRA****. The matter embodied in this project work has not been submitted earlier for award of any degree or diploma to the best of my knowledge and belief.*

Akshat Srivastava

INDEX

1. introduction.....

- *1.1 Overview of Organization*
- *1.2 Objectives*
- *1.3 Existing System Description*
- *1.4 Proposed System*
- *1.5 Tools Used*

2. System Analysis.....

- *2.1 Objective*
- *2.2 SDLC Phases*
  - *2.2.1 Feasibility Study*
  - *2.2.2 Report Approval*
  - *2.2.3 SystemAnalysis*
  - *2.2.4 System Design*
  - *2.2.5 Coding*
  - *2.2.6 Testing*
  - *2.2.7 Implementation*
  - *2.2.8 Maintenance*
- *2.3 Process Description*

2.4 *Project Model Used*
2.5 *ER-Diagram*
2.6 *Data Flow Diagram*

3. Software Requirement Specification.....

3.1 *Hardware Requirement*

3.2 *Software Requirement*

3.3 *Support Maintenance*

4. System Design Approach.....

4.1 *Top-Down Designing*

4.2 *Bottom –Up Designing*

4.3 *Following Approach*

5. Low Level Design.....

5.1 *Description of Classes and Methods*

6. Data Modeling.....

6.1 *List of Tables*
6.2 *Structure of Tables*

7. Testing.....

8. Input-Output Forms.....

8.1 *Modularization Details*
8.2 *User Screenshots*
8.3 *Admin Screenshots*

9. Future Scope.....

10. Conclusion.....

INTRODUCTION

This project is responding to the requirement of E Banking Solution. Developing a Web based Online Interface that manage banking operations like create account, deposit money, withdraw money, fund transfer and balance enquiry. Through this project we will provide the services for design, development and testing of web based E Banking Solution.

The objective of our efforts is to present a comprehensive and an iterative approach in building strong, flexible and full featured E Banking solution.

Organization of the report:

Softpro India Computer Technologies (P) Limited was founded in 2004 with the mission of "Bridging the Technology gap".

Provides programming services, Application Development and Integration Services, Web Design and Development Services, Search Engine Optimization Services (SEO) and Back-office services.

SYSTEM ANALYSIS:

Identification of Need:

Identification of need mainly concern with requirements of the project how much time it would be taken and how much cost it would require to complete. The hardware requirement of the project also concern. The member of the project team also is decided in this phase.

Here we focus our point towards organizing needs i.e. to organize the things which are scattered here and there.

Software Requirement Specification (SRS):

Introduction:

To develop a web-based application to improve the service to the customers of bank.

Constraints:

The main constrain there would be the checking the genuineness of the buyer, which is not always possible. There can be security risks involved.

Assumptions and Dependencies:

- *The details related to the product, customer, payment and service transaction provided manually.*
- *Administrator is created in the system already.*
- *Roles and tasks are predefined.*

Software Specification:-

Programming Language Support

- *Python with Django Framework*
- *Java Script for client side validation*

Software Requirement for Development

- *IDE- PyCharm*
- *SQLite Database*

Client side Software Requirement

- *Google Chrome Browser*
- *Operating System*

Platform:

Windows platform like: 2000 Server, Professional, XP & Vista

Hardware Specification:

- *Intel Pentium and Celeron class processor*
- *RAM - 512M.B.(min)*
- *HDD - 40GB*
- *Monitor-14"SVGA*
- *Printer – Dot Matrix /Inkjet /Laser Printer*
- *Mouse & Keyboard-Normal*

For Client side: Web browser- IE 7 or above, Google chrome, Safari.

Functional Requirement:

The System must provide following functionalities—

- *Keeping records of customersaccount.*
- *Keeping the records of transaction.*
- *Administration*

Non Functional Requirement:

The System must provide following functionalities—

- *Secure access of confidential data (customer's details).*
- *Better component design to get better performance at peak time.*

Flexible service based architecture will be highly desirable for future extension Non functional requirements define system properties and constraints It arise through user needs, because of budget constraints or organizational policies, or due to the external factors such as safety regulations,

privacy registration and so on.

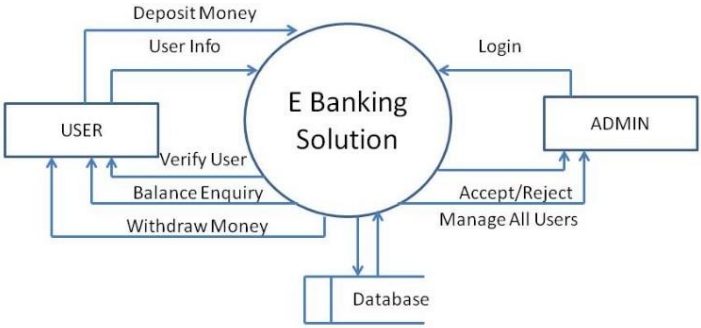
Various other Non-functional requirements are:

- *Security*
- *Reliability*
- *Maintainability*
- *Portability*
- *Extensibility*
- *Reusability*
- *Application Affinity/Compatibility*
- *Resource Utilization*

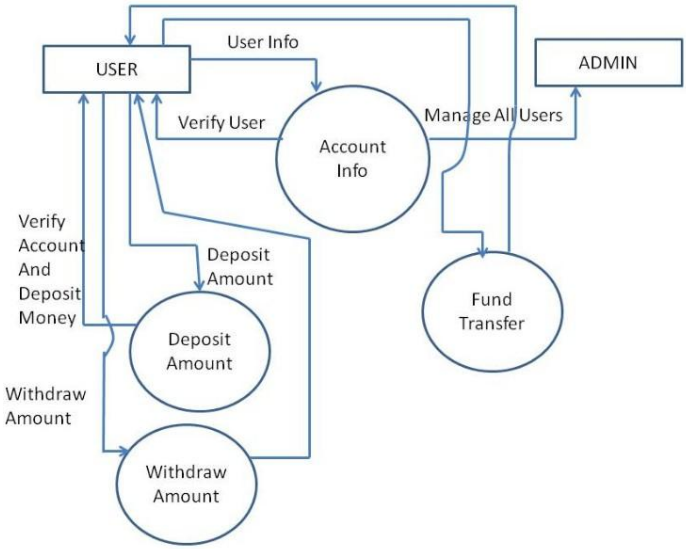
Data Models:

DFD (Data Flow Diagram):

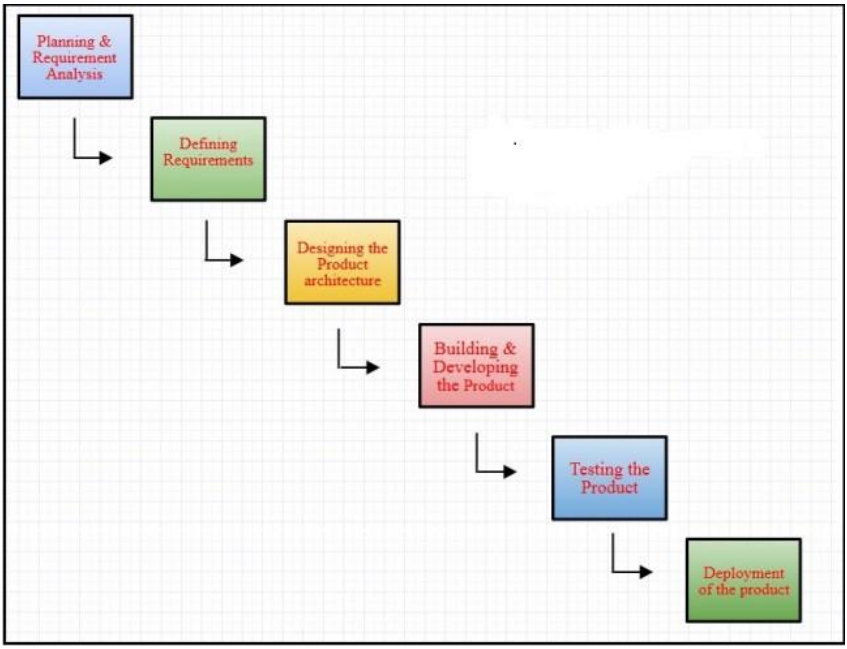
0 LEVEL DFD


Deposit Money
User Info
Login
E Banking Solution
USER
ADMIN
Verify User
Balance Enquiry
Accept/Reject
Manage All Users
Withdraw Money
Database

1 LEVEL DFD



SYSTEM DEVELOPMENT LIFE CYCLE



System Design

Modularisation details:

- *Customer Module*
- *Deposit Amount*
- *Withdraw Amount*
- *Fund Transfer*
- *Balance Enquiry*
- *Login Management*
- *Administration*

Customer Module:

This module performs the registration of account holder and further allows them to perform bank operations.

Deposit Amount:

In this module user first login with account no and password if user is valid then he/she will able to deposit amount.

Withdraw Amount:

In this module user first login with account no and password if user is valid then he/she will able to withdraw amount. In this module first the amount is compare with balance, if balance is not available it will display insufficient balance.

Fund Transfer:

In this module user first login with account no and password if user is valid then he/she will able to transfer amount. In this module first the amount is compare with balance, if balance is not available it will display insufficient balance.

Balance Enquiry:

In this module user first login with account no and password if user is valid then he/she will able to check balance.

Login Management:

In this module we validate the user.

Administration:

In this module admin do login and manage customer..

Data Integrity and Constraints:

Account Table

Name	*Constraint*	*Data Type*
acno	*Primary Key*	*int*
name		*varchar(50)*
address		*varchar(255)*
contactno		*varchar(15)*
emailaddress		*varchar(50)*
panno		*varchar(10)*
aadharno		*varchar(12)*
balance		*int*
password		*varchar(20)*

Login Table

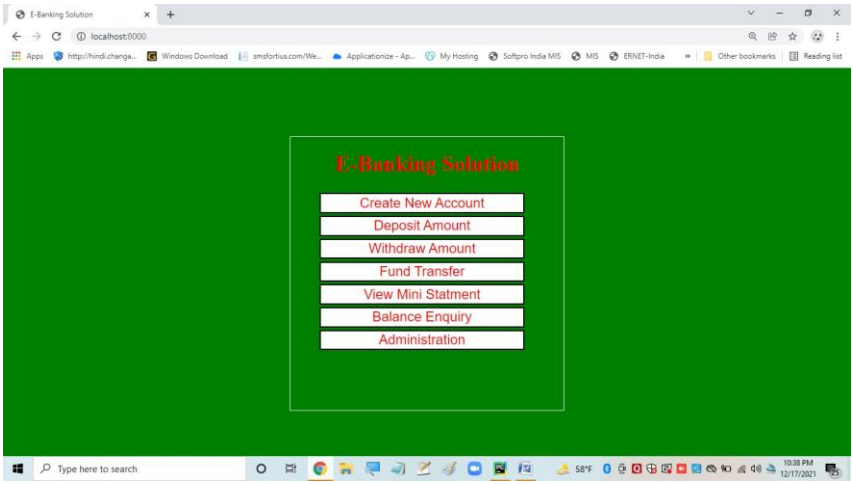
Name	*Constraint*	*Data Type*
userid	*Primary Key*	*varchar(50)*
password		*varchar(20)*

Statement Table

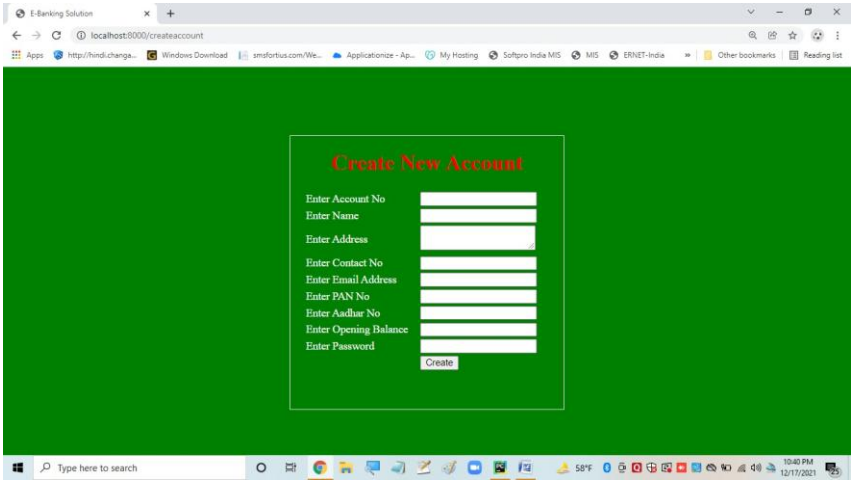
Name	*Constraint*	*Data Type*
fromaccount	*Primary Key*	*int*
toaccount		*int*
operation		*varchar(20)*
amount		*int*
opdate		*varchar(20)*

Snap Shots:-

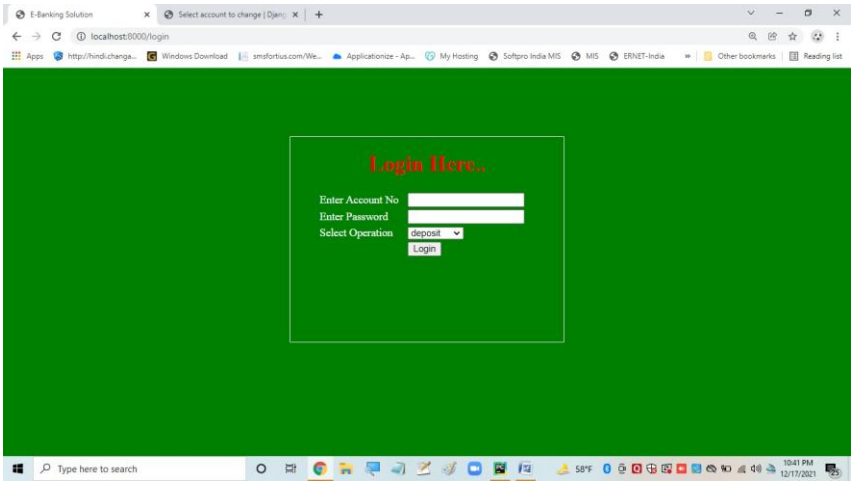
Index Page:-



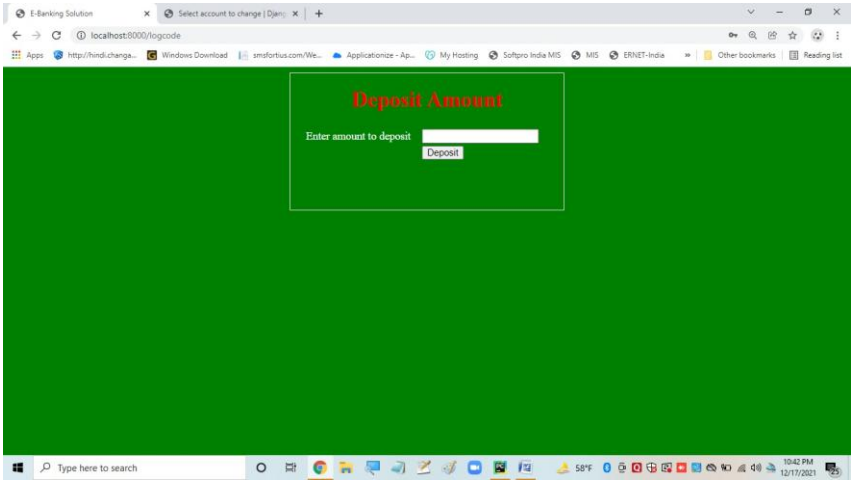
Create Account Page:-



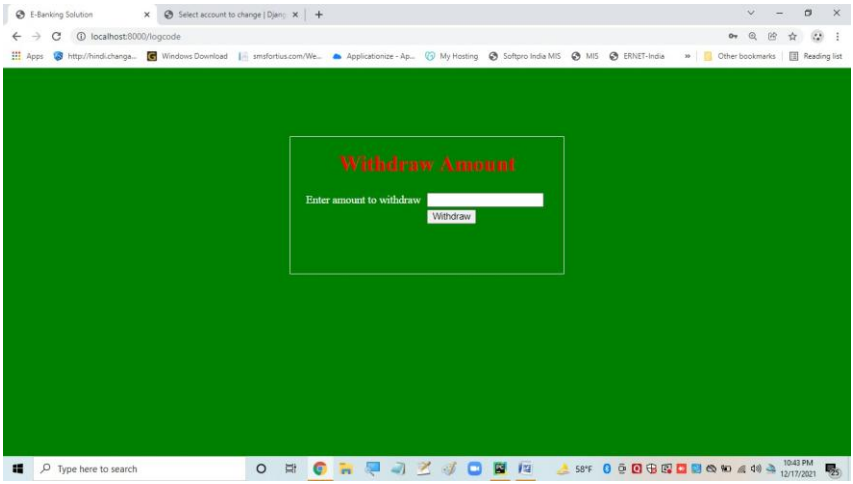
Login Page:-



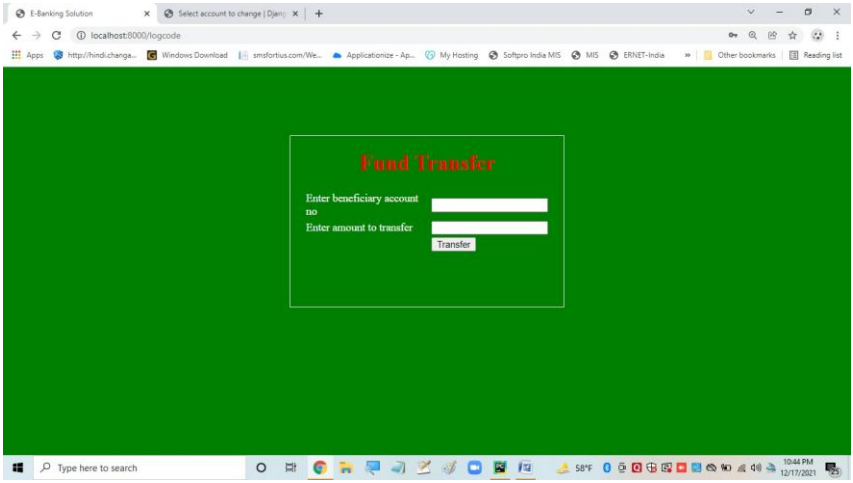
Deposit Amount:-



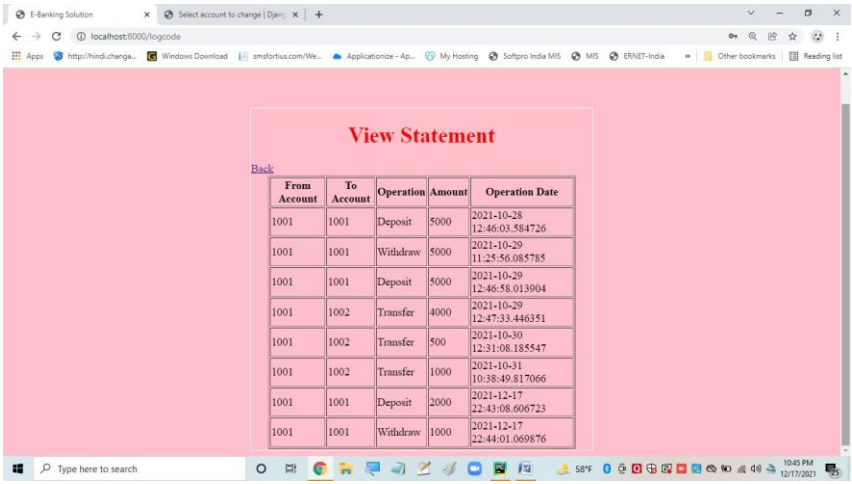
Withdraw Amount:-



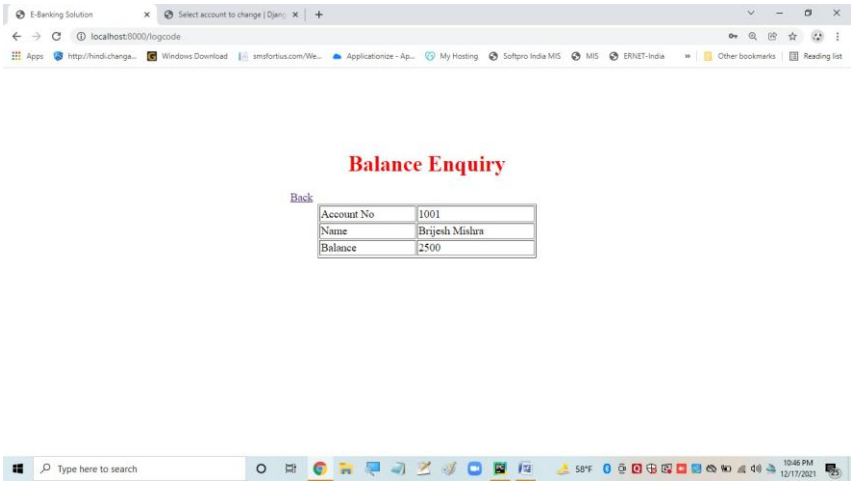
Fund Transfer Page:-



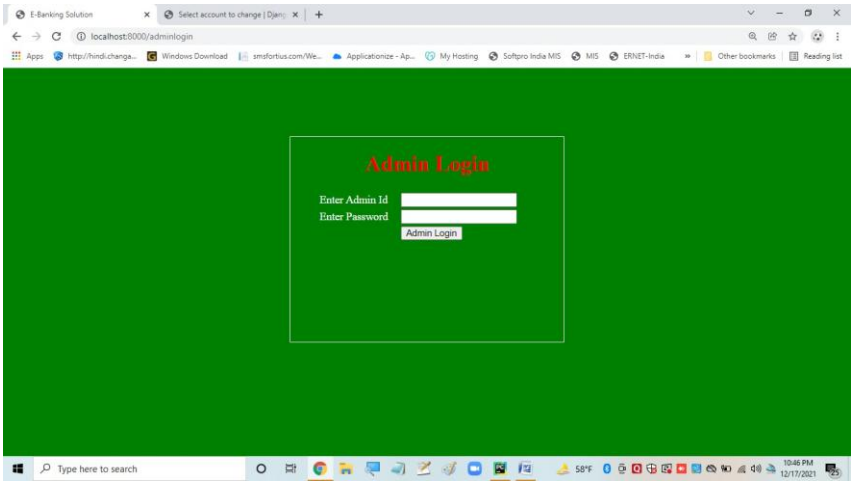
View Statement Page:-



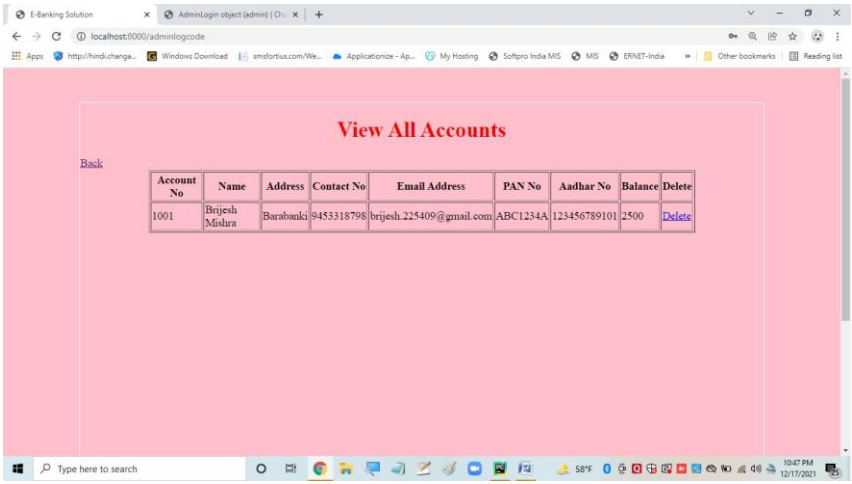
Balance Enquiry Page:-



Admin Login Page:-



Administration Page:-



Coding:-

index.html template:-

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>E-Banking Solution</title>
    <style>
        #main
        {
            width:400px;
            height:400px;
            border:1px solid white;
            margin:auto;
            margin-top:100px;
        }
```

```
        #main h1
        {
            text-align:center;
            color:red;
        }
        .mybtn
        {
            background-color:white;
            border:2px solid black;
            color:red;
            text-align:center;
            width:300px;
            font-size:20px;
            cursor:pointer;
        }
    </style>
</head>
<body bgcolor="green">
    <div id="main">
        <h1>E-Banking Solution</h1>
        {% if msg %}
            <p style="color:red;text-
align:center;">{{msg}}</p>
        {% endif %}
        <table style="margin:0 auto;width:80%;">
            <tr>
                <td>
                    <button class="mybtn"
onclick="window.location.href='{% url
"createaccount" %}'">Create New Account</button>
                </td>
```

```
        </tr>
        <tr>
          <td>
            <button class="mybtn"
onclick="window.location.href='{% url "login"
 %}'">Deposit Amount</button>
          </td>
        </tr>
        <tr>
          <td>
            <button class="mybtn"
onclick="window.location.href='{% url "login"
 %}'">Withdraw Amount</button>
          </td>
        </tr>
        <tr>
          <td>
            <button class="mybtn"
onclick="window.location.href='{% url "login"
 %}'">Fund Transfer</button>
          </td>
        </tr>
        <tr>
          <td>
            <button class="mybtn"
onclick="window.location.href='{% url "login"
 %}'">View Mini Statment</button>
          </td>
        </tr>
        <tr>
          <td>
```

```
                <button class="mybtn"
onclick="window.location.href='{% url "login"
 %}'">Balance Enquiry</button>
            </td>
          </tr>
          <tr>
            <td>
                <button class="mybtn"
onclick="window.location.href='{% url "adminlogin"
 %}'">Administration</button>
            </td>
          </tr>
        </table>
    </div>
</body>
</html>
```

adminlogin.html template:-

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>E-Banking Solution</title>
    <style>
        #main
        {
            width:400px;
            height:300px;
            margin:0 auto;
            margin-top:100px;
```

```
        border:1px solid white;
      }
      #main h1
      {
        text-align:center;
        color:red;
      }
    </style>
</head>
<body bgcolor="green">
    <div id="main">
      <h1>Admin Login</h1>
      {% if msg %}
        <p style="text-align:center;">{{msg}}</p>
      {% endif %}
      <form action="{% url 'adminlogcode' %}"
method="post">
        {% csrf_token %}
        <table style="width:80%;margin:0 auto;">
          <tr>
            <td style="color:white;">
              Enter Admin Id
            </td>
            <td>
              <input type="text" name="adminid"/>
            </td>
          </tr>
          <tr>
            <td style="color:white;">
              Enter Password
            </td>
```

```
            <td>
                <input type="password"
name="password"/>
            </td>
        </tr>
        <tr>
            <td></td>
            <td>
                <input type="submit" value="Admin
Login"/>
            </td>
        </tr>
    </table>
  </form>
 </div>
</body>
</html>
```

createaccount.html template:-

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>E-Banking Solution</title>
  <style>
    #main
    {
      width:400px;
      height:400px;
      border:1px solid white;
```

```
            margin:0 auto;
            margin-top:100px;
        }
        #main h1
        {
            text-align:center;
            color:red;
        }
    </style>
</head>
<body bgcolor="green">
    <div id="main">
        <h1>Create New Account</h1>
        <form action="{% url 'create' %}"
method="post">
            {% csrf_token %}
        <table style="width:90%;margin:0 auto;">
            <tr>
                <td style="color:white;">Enter Account
No</td>
                <td>
                    <input type="number" name="acno"/>
                </td>
            </tr>
            <tr>
                <td style="color:white;">EnterName</td>
                <td>
                    <input type="text" name="name"/>
                </td>
            </tr>
            <tr>
```

```
            <td style="color:white;">Enter Address</td>
            <td>
              <textarea name="address"></textarea>
            </td>
          </tr>
          <tr>
            <td style="color:white;">Enter Contact
No</td>
            <td>
              <input type="number"
name="contactno"/>
            </td>
          </tr>
          <tr>
            <td style="color:white;">Enter Email
Address</td>
            <td>
              <input type="email"
name="emailaddress"/>
            </td>
          </tr>
          <tr>
            <td style="color:white">Enter PAN No</td>
            <td>
              <input type="text" name="panno"/>
            </td>
          </tr>
          <tr>
            <td style="color:white;">Enter Aadhar
No</td>
            <td>
```

```
            <input type="number"
name="aadharno"/>
          </td>
        </tr>
        <tr>
          <td style="color:white;">Enter Opening
Balance</td>
          <td>
            <input type="number"name="balance"/>
          </td>
        </tr>
        <tr>
          <td style="color:white;">Enter
Password</td>
          <td>
            <input type="password"
name="password"/>
          </td>
        </tr>
        <tr>
          <td></td>
          <td>
            <input type="submit"value="Create"/>
          </td>
        </tr>
      </table>
    </form>
  </div>
</body>
</html>
```

deposit.html template:-

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <title>E-Banking Solution</title>
   <style>
      #main
      {
         width:400px;
         height:200px;
         border:1px solid white;
         margin-top:100px;
         margin:0 auto;
      }
      #main h1
      {
         text-align:center;
         color:red;
      }
   </style>
</head>
<body bgcolor="green">
   <div id="main">
      <h1>Deposit Amount</h1>
      <form action="{% url 'depositamt' %}"
method="post">
         {% csrf_token %}
         <table style="margin:0 auto;width:90%;">
            <tr>
```

```
                <td style="color:white;">
                    Enter amount to deposit
                </td>
                <td>
                    <input type="number" name="amt"/>
                </td>
            </tr>
            <tr>
                <td></td>
                <td>
                    <input type="submit" value="Deposit"/>
                </td>
            </tr>
        </table>
    </form>
  </div>
</body>
</html>
```

enquiry.html template:-

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>E-Banking Solution</title>
  <style>
    #main
    {
      width:400px;
      height:300px;
```

```
            margin:0 auto;
            margin-top:100px;
            border:1px solid white;
        }
        #main h1
        {
            text-align:center;
            color:red;
        }
    </style>
</head>
<body>
    <div id="main">
        <h1>Balance Enquiry</h1>
        <a href="{% url 'back' %}">Back</a><br/>
        {% if enq %}
            <table style="width:80%;margin:0 auto;"
border="1">
                <tr>
                    <td>Account No</td>
                    <td>{{enq.acno}}</td>
                </tr>
                <tr>
                    <td>Name</td>
                    <td>{{enq.name}}</td>
                </tr>
                <tr>
                    <td>Balance</td>
                    <td>{{enq.balance}}</td>
                </tr>
            </table>
```

```
        {% endif %}
    </div>
</body>
</html>
```

login.html template:-

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>E-Banking Solution</title>
    <style>
        #main
        {
            width:400px;
            height:300px;
            border:1px solid white;
            margin:0 auto;
            margin-top:100px;
        }
        #main h1
        {
            text-align:center;
            color:red;
        }
    </style>
</head>
<body bgcolor="green">
    <div id="main">
        <h1>Login Here..</h1>
```

```
        {% if msg %}
         <p style="text-align:center">{{msg}}</p>
        {% endif %}
        <form action="{% url 'logcode' %}"
method="post">
          {% csrf_token %}
          <table style="width:80%;margin:0 auto;">
            <tr>
              <td style="color:white;">
                Enter Account No
              </td>
              <td>
                <input type="number" name="acno"/>
              </td>
            </tr>
            <tr>
              <td style="color:white;">
                Enter Password
              </td>
              <td>
                <input type="password"
name="password"/>
              </td>
            </tr>
            <tr>
              <td style="color:white;">
                Select Operation
              </td>
              <td>
                <select name="op">
                  <option>deposit</option>
```

```
                <option>withdraw</option>
                <option>transfer</option>
                <option>statement</option>
                <option>enquiry</option>
            </select>
        </td>
    </tr>
    <tr>
        <td></td>
        <td>
            <input type="submit"value="Login"/>
        </td>
    </tr>
</table>
</form>
</div>
</body>
</html>
```

statement.html template:-

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>E-Banking Solution</title>
    <style>
        #main
        {
            width:500px;
            height:auto;
```

```
            min-height:500px;
            margin:0 auto;
            margin-top:100px;
            border:1px solid white;
        }
        #main h1
        {
            text-align:center;
            color:red;
        }
    </style>
</head>
<body bgcolor="pink">
    <div id="main">
        <h1>View Statement</h1>
        <a style="margin:0 auto;" href="{% url 'back'
 %}">Back</a>
        <table border="1" style="margin:0
auto;width:90%;">
            <tr>
                <th>From Account</th>
                <th>To Account</th>
                <th>Operation</th>
                <th>Amount</th>
                <th>Operation Date</th>
            </tr>
            {% for s in stmt %}
                <tr>
                    <td>{{s.fromaccount}}</td>
                    <td>{{s.toaccount}}</td>
                    <td>{{s.operation}}</td>
```

```
                <td>{{s.amount}}</td>
                <td>{{s.opdate}}</td>
            </tr>
        {% endfor %}
    </table>
  </div>
</body>
</html>
```

transfer.html template:-

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>E-Banking Solution</title>
  <style>
    #main
    {
      width:400px;
      height:250px;
      margin:0 auto;
      margin-top:100px;
      border:1px solid white;
    }
    #main h1
    {
      text-align:center;
      color:red;
    }
  </style>
```

```
</head>
<body bgcolor="green">
   <div id="main">
      <h1>Fund Transfer</h1>
      <form action="{% url 'transferamt' %}"
method="post">
         {% csrf_token %}
         <table style="margin:0 auto;width:90%;">
            <tr>
               <td style="color:white;">
                  Enter beneficiary account no
               </td>
               <td>
                  <input type="number"
name="toaccount"/>
               </td>
            </tr>
            <tr>
               <td style="color:white;">
                  Enter amount totransfer
               </td>
               <td>
                  <input type="number" name="amt"/>
               </td>
            </tr>
            <tr>
               <td></td>
               <td>
                  <input type="submit"
value="Transfer"/>
               </td>
```

```
            </tr>
          </table>
        </form>
    </div>
</body>
</html>
```

viewaccounts.html template:-

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>E-Banking Solution</title>
    <style>
        #main
        {
            width:1000px;
            height:auto;
            min-height:800px;
            margin:0 auto;
            margin-top:50px;
            border:1px solid white;
        }
        #main h1
        {
            text-align:center;
            color:red;
        }
    </style>
</head>
```

```
<body bgcolor="pink">
   <div id="main">
      <h1>View All Accounts</h1>
      <a href="{% url 'back' %}">Back</a>
      <table border="1" style="width:80%;margin:0
auto;">
         <tr>
            <th>Account No</th>
            <th>Name</th>
            <th>Address</th>
            <th>Contact No</th>
            <th>Email Address</th>
            <th>PAN No</th>
            <th>Aadhar No</th>
            <th>Balance</th>
            <th>Delete</th>
         </tr>
         {% for a in ac %}
            <tr>
               <td>{{a.acno}}</td>
               <td>{{a.name}}</td>
               <td>{{a.address}}</td>
               <td>{{a.contactno}}</td>
               <td>{{a.emailaddress}}</td>
               <td>{{a.panno}}</td>
               <td>{{a.aadharno}}</td>
               <td>{{a.balance}}</td>
               <td>
                  <a href="{% url 'deleteaccount' a.acno
 %}">Delete</a>
               </td>
```

```
            </tr>
          {% endfor %}
        </table>
    </div>
</body>
</html>
```

withdraw.html template:-

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>E-Banking Solution</title>
    <style>
        #main
        {
            width:400px;
            height:200px;
            margin:0 auto;
            margin-top:100px;
            border:1px solid white;
        }
        #main h1
        {
            text-align:center;
            color:red;
        }
    </style>
</head>
<body bgcolor="green">
```

```
    <div id="main">
      <h1>Withdraw Amount</h1>
      <form action="{% url 'withdrawamt' %}"
method="post">
        {% csrf_token %}
        <table style="margin:0 auto;width:90%;">
          <tr>
            <td style="color:white">
              Enter amount to withdraw
            </td>
            <td>
              <input type="number" name="amt"/>
            </td>
          </tr>
          <tr>
            <td></td>
            <td>
              <input type="submit"
value="Withdraw"/>
            </td>
          </tr>
        </table>
      </form>
    </div>
</body>
</html>
```

views.py code:-

```
from django.shortcuts import render, redirect
from . models import Account, AdminLogin,
```

```
Statement
from django.core.exceptions import
ObjectDoesNotExist
import datetime

# Create your views here.
def index(request):
    return render(request,"index.html")
def createaccount(request):
    return render(request,"createaccount.html")
def create(request):
    acno=request.POST['acno']
    name=request.POST['name']
    address=request.POST['address']
    contactno=request.POST['contactno']
    emailaddress=request.POST['emailaddress']
    panno=request.POST['panno']
    aadharno=request.POST['aadharno']
    balance=request.POST['balance']
    password=request.POST['password']

ac=Account(acno=acno,name=name,address=addres
s,contactno=contactno,emailaddress=emailaddress,p
anno=panno,aadharno=aadharno,balance=balance,pa
ssword=password)
    ac.save()
    return redirect('index')
def login(request):
    return render(request,"login.html")
def logcode(request):
    acno=request.POST['acno']
```

```
    password=request.POST['password']
    op=request.POST['op']
    msg=''
    try:

obj=Account.objects.get(acno=acno,password=pass
word)
        if obj is not None:
            if op=="deposit":
                acno=obj.acno
                request.session['acno']=acno
                return render(request,"deposit.html")
            elif op=="withdraw":
                acno=obj.acno
                request.session['acno']=acno
                return render(request,"withdraw.html")
            elif op=="transfer":
                acno=obj.acno
                request.session['acno']=acno
                return render(request,"transfer.html")
            elif op=="statement":
                acno=obj.acno

stmt=Statement.objects.filter(fromaccount=acno)
                return
render(request,"statement.html",{'stmt':stmt})
            elif op=="enquiry":
                acno=obj.acno
                enq=Account.objects.get(acno=acno)
                return
render(request,"enquiry.html",{'enq':enq})
```

```
    except ObjectDoesNotExist:
        msg='Invalid account'
    return render(request,"login.html",{'msg':msg})
def depositamt(request):
    amt=int(request.POST['amt'])

obj=Account.objects.get(acno=request.session['acno
'])
    balance=obj.balance
    balance=balance+amt
    acno=obj.acno

Account.objects.filter(pk=acno).update(balance=bala
nce)
    opdate=datetime.datetime.today()
    operation="Deposit"

stmt=Statement(fromaccount=acno,toaccount=acno,
operation=operation,amount=amt,opdate=opdate)
    stmt.save()
    request.session['acno']=None
    return redirect('index')
def withdrawamt(request):
    amt=int(request.POST['amt'])

obj=Account.objects.get(acno=request.session['acno
'])
    balance=obj.balance
    if amt>balance:
        return
render(request,"index.html",{'msg':'Insufficient
```

```
balance'})
    balance=balance-amt
    acno=obj.acno

Account.objects.filter(pk=acno).update(balance=bala
nce)
    operation="Withdraw"
    opdate=datetime.datetime.today()

stmt=Statement(fromaccount=acno,toaccount=acno,
operation=operation,amount=amt,opdate=opdate)
    stmt.save()
    request.session['acno']=None
    return redirect('index')
def transferamt(request):
    toaccount=request.POST['toaccount']
    amt=int(request.POST['amt'])
    msg=''
    try:
        obj2=Account.objects.get(acno=toaccount)
        if obj2 is not None:

obj1=Account.objects.get(acno=request.session['acn
o'])
            balance1=obj1.balance
            if amt>balance1:
                return
render(request,"index.html",{'msg':'Insufficient
Balance'})
            balance1=balance1-amt
            balance2=obj2.balance
```

```
            balance2=balance2+amt

Account.objects.filter(pk=obj1.acno).update(balance
=balance1)

Account.objects.filter(pk=obj2.acno).update(balance
=balance2)
            operation="Transfer"
            opdate=datetime.datetime.today()

stmt1=Statement(fromaccount=obj1.acno,toaccount
=obj2.acno,operation=operation,amount=amt,opdate
=opdate)
            stmt2=Statement(fromaccount=obj2.acno,
toaccount=obj1.acno,
operation=operation,amount=amt,opdate=opdate)
            stmt1.save()
            stmt2.save()
            request.session['acno']=None
            return redirect('index')
        except ObjectDoesNotExist:
            request.session['acno']=None
            return render(request,"index.html",{'msg':'Invalid
Account No'})
def back(request):
    request.session['acno']=None
    return redirect('index')
def adminlogin(request):
    return render(request,"adminlogin.html")
def adminlogcode(request):
    adminid=request.POST['adminid']
```

```
    password=request.POST['password']
    msg=''
    try:

obj=AdminLogin.objects.get(userid=adminid,passwor
d=password)
        if obj is not None:
            ac=Account.objects.all()
            return
render(request,"viewaccounts.html",{'ac':ac})
    except ObjectDoesNotExist:
        msg='Invalid user'
    return
render(request,"adminlogin.html",{'msg':msg})
def deleteaccount(request,acno):
    obj=Account.objects.get(acno=acno)
    obj.delete()
    ac=Account.objects.all()
    return render(request,"viewaccounts.html",{'ac':ac})
```

admin.py code:-

```
from django.contrib import admin
```

```
from . models import
Account,AdminLogin,Statement

# Register your models here.
admin.site.register(Account)
admin.site.register(AdminLogin)
admin.site.register(Statement)
```

models.py code:-

```
from django.db import models

# Create your models here.
class Account(models.Model):
    acno=models.IntegerField(primary_key=True)
    name=models.CharField(max_length=50)
    address=models.TextField()
    contactno=models.CharField(max_length=15)
    emailaddress=models.CharField(max_length=50)
    panno=models.CharField(max_length=10)
    aadharno=models.CharField(max_length=12)
    balance=models.IntegerField()
    password=models.CharField(max_length=20)
class AdminLogin(models.Model):

userid=models.CharField(max_length=50,primary_key
=True)
    password=models.CharField(max_length=20)
class Statement(models.Model):
    fromaccount=models.IntegerField()
    toaccount=models.IntegerField()
```

```
    operation=models.CharField(max_length=20)
    amount=models.IntegerField()
    opdate=models.CharField(max_length=30)
```

bankappurls.py code:-

```
from django.conf.urls import url
from . import views

urlpatterns=[
    url(r'^$',views.index,name="index"),

url(r'^createaccount',views.createaccount,name="cre
ateaccount"),
    url(r'^create',views.create,name="create"),
    url(r'^login',views.login,name="login"),
    url(r'^logcode',views.logcode,name="logcode"),

url(r'^depositamt',views.depositamt,name="deposita
mt"),

url(r'^withdrawamt',views.withdrawamt,name="withdr
awamt"),
```

```
url(r'^transferamt',views.transferamt,name="transferamt"),
    url(r'^back',views.back,name="back"),

url(r'^adminlogin',views.adminlogin,name="adminlogin"),

url(r'^adminlogcode',views.adminlogcode,name="adminlogcode"),

url(r'^deleteaccount/(?P<acno>\d+)$',views.deleteaccount,name="deleteaccount"),
]
```