Chapter 1

INTRODUCTION

Introduction

1.1. Introduction to project

This project is designed to provide a flexible environment for its users. It is a non-sponsored website, developed as a web-based product using HTML/CSS and JavaScript technologies. The primary objective of this project is to practice and apply my technological learnings.

Users are required to sign up first to gain access to the website. The membership is divided into two categories: Demo and Users.

The website offers various functions categorized into two views: Demo View and Registered User View. In the Demo View, users can explore the entire website, including features like E-transfer, Fixed Deposit, International Transfer, etc. Registered users, on the other hand, have access to additional functionalities based on their requirements.

Registered users can sign up and log in to access the website's features. They can perform transactions to other accounts and update their profile details. Additionally, they can avail e-services such as bill payments and online shopping.

This project aims to provide an overview of how a banking website operates. It allows users to explore the functions provided by banks, thereby enabling new users to familiarize themselves with banking facilities through the website.

1.2. **Introduction to Visual Studio Code**

Visual Studio Code (VS Code) is a robust and flexible code editor crafted by Microsoft. It provides an intuitive and feature-rich environment for writing code in various programming languages, including HTML, CSS, and JavaScript.

With its lightweight design and extensive customization options, VS Code offers a seamless development experience for web developers. Its robust set of features, coupled with a vast ecosystem of extensions, makes it a preferred choice for building web applications and websites.

HTML, CSS, and JavaScript are fundamental languages used in web development, and VS Code offers excellent support for editing and debugging code written in these languages. The editor provides syntax highlighting, auto-completion, and code formatting features, enhancing productivity and code readability.

Developers can easily organize their projects, manage files, and collaborate with team members using VS Code's integrated version control systems, such as Git. The built-in terminal allows for seamless command-line interaction, enabling developers to run scripts and perform tasks without leaving the editor.

Furthermore, VS Code's live server extension enables developers to preview their web applications in real-time, making it easier to visualize changes and debug code efficiently.

In conclusion, Visual Studio Code provides a comprehensive and user-friendly environment for web development with HTML, CSS, and JavaScript. Its robust features and extensibility empower developers to create stunning web experiences with ease.

1.3.Description-

Online Banking-The main objective of this project is to provide knowledge about Banking services. It will help users about banking services like Login, Domestic/International Money Transfer.

For a user he has to enter details like Account no. ,IFSC Code , Branch etc.
There several modules that as a mediator between the jobseekers and companies.

To register, users are required to input their details into the registration form.. He has to select his own registration no through which he has to login.

There are several modules that worked once when the user logged in with his registration no and password. The user can choose any of the components displayed and can avail the services according to his need. Username and password is given for each user.

1.4.Objective

- Conduct banking services from the comfort of your home using just a computer or mobile device and an Internet connection.
- Access information about banking services and perform transactions conveniently.
- Online banking offers faster results compared to traditional methods of conducting transactions.
- Tasks like fund transfers, FD-redemptions, bill payments, and correspondence with potential users can be expedited with just a click of your mouse or a touch on your phone screen.
- The cost of conducting your transactions online is often free, saving you on transportation costs since you don't need to travel to a physical bank branch.

- Online banking also saves you time, allowing you to set up additional security features for added protection.
- Easily access mini-statements and search for specific transactions using our banking service.
- Online banking is user-friendly, interactive, and less time-consuming compared to traditional banking methods.
- Complete transactions without the need for assistance from others, as required in traditional banking.
- Access banking services 24/7, allowing you to manage your finances at any time of day or night.
- Enjoy greater flexibility in managing your accounts, including the ability to schedule payments and transfers in advance.
- Receive instant notifications for account activities, such as deposits, withdrawals, and bill payments.
- Access digital statements and documents, reducing the need for paper records and helping the environment.
- Securely store and manage important financial documents and information online.
- Easily track your spending and budgeting with tools and features available through online banking platforms.
- Access a wide range of banking services and products, including loans, investments, and insurance, all from the convenience of your home.
- Benefit from personalized recommendations and insights based on your banking history and preferences.

- Access customer support and assistance through online chat, email, or phone, without the need to visit a physical branch.
- Enjoy peace of mind with robust security features, including encryption, multi-factor authentication, and fraud detection systems.

Chapter 2

System Analysis

System Analysis

Accurate time estimation is a skill essential to good project management.it is important to get time estimates right for two main reasons:

1. Time estimates drive the setting of deadlines for delivery of projects and hence peoples assements of your reliability.
2. they often determine the pricing of contracts and hence their profitability

People often underestimate the time required to complete projects, especially when they lack familiarity with the task at hand.

The planning of project needs to be scheduled so that work may progress accordingly.project will also get completed in stipulated time.

Necessity:

- **Market Demand**: Online banking systems cater to the increasing demand for convenient and accessible banking services.
- **Customer Expectations**: Customers expect banks to provide online platforms for easy access to their accounts and transactions.
- **Competitive Edge**: Offering online banking services helps banks stay competitive in the market and attract tech-savvy customers.

Estimating Time:

- **Development Time**: The development of the online banking system may vary based on the complexity of features and functionalities.

- **Testing Time**: Rigorous testing is required to ensure the system functions seamlessly across different devices and platforms.
- **Deployment Time**: Deploying the system involves setting up servers, configuring security measures, and ensuring compatibility with existing banking infrastructure.

Security:

- **Encryption**: Implementing strong encryption protocols to secure data transmission and protect customer information.
- **Authentication**: Utilizing multi-factor authentication methods to verify user identities and prevent unauthorized access.
- **Firewalls and Intrusion Detection Systems**: Installing firewalls and intrusion detection systems to safeguard against cyber threats and attacks.
- **Regular Audits**: Conducting regular security audits to identify vulnerabilities and mitigate risks proactively.

Documentation:

- **User Manuals**: Providing comprehensive user manuals and guides to help customers navigate the online banking platform.
- **Developer Documentation**: Documenting the system architecture, APIs, and coding standards to facilitate maintenance and future enhancements.
- **Training Materials**: Developing training materials for bank staff to familiarize them with the system's features and functionalities.

Data Integrity and Accuracy:

- **Data Validation**: Implementing data validation mechanisms to ensure the accuracy and integrity of customer information and transactions.
- **Error Handling**: Incorporating robust error-handling mechanisms to prevent data corruption and ensure the reliability of the system.
- **Backup and Recovery**: Establishing regular data backup procedures and disaster recovery plans to mitigate the risk of data loss or corruption.

Synchronization of Usage:

- **Multi-Platform Compatibility**: Ensuring the online banking system is compatible with various devices and operating systems to accommodate different user preferences.
- **Real-Time Updates**: Providing real-time updates and synchronization of user data across multiple devices to maintain consistency and usability.
- **Offline Access**: Implementing features that allow users to access certain functionalities offline and synchronize data when online connectivity is restored.

Privacy:

- **Data Protection Laws**: Adhering to data protection laws and regulations to safeguard customer privacy and prevent unauthorized data sharing.
- **Privacy Policies**: Clearly communicating privacy policies and terms of service to customers to ensure transparency and trust.

- **Opt-In/Opt-Out Mechanisms**: Providing users with options to control the sharing of their personal information and preferences regarding targeted marketing and promotions.

Chapter 3

Feasibility Study

Feasibility Study

The feasibility study for the proposed online banking system encompasses various aspects, including economical, technical, and operational feasibility.

Economical Feasibility:

- **Cost Savings:** Reduction in overhead costs associated with physical branches through the adoption of online banking services.
- **Revenue Generation:** Potential increase in profits and customer retention by meeting the demand for online banking services.
- **Competitive Advantage:** Enhanced competitiveness and market share through the provision of convenient online banking solutions.
- **Market Analysis:** Assessment of the target market's willingness to adopt online banking services and their potential spending behavior.
- **Return on Investment (ROI):** Calculation of the anticipated ROI based on projected cost savings and revenue generation over time.
- **Risk Assessment:** Identification and mitigation of potential risks that may impact the financial viability of the online banking system.

Technical Feasibility:

- **Robust Technologies:** Utilization of modern web development frameworks and encryption protocols to ensure secure and scalable operations.

- **Integration:** Seamless integration with existing banking infrastructure and third-party services to facilitate smooth transactions and operations.
- **Data Security:** Implementation of stringent security measures to safeguard customer data and maintain regulatory compliance.
- **Scalability:** Evaluation of the system's ability to handle increasing user demand and transaction volumes without compromising performance.
- **Compatibility:** Ensuring compatibility with a wide range of devices, browsers, and operating systems to maximize accessibility for users.
- **Future Expansion:** Designing the system architecture with flexibility to accommodate future enhancements and technological advancements.

Operational Feasibility:

- **Process Streamlining:** Automation of routine banking tasks to reduce manual errors and processing times.
- **Customer Experience:** Improvement in customer experiences through faster service delivery and enhanced online banking features.
- **Training and Support:** Provision of comprehensive training and support to bank staff and customers for effective adoption and usage of the online banking platform.
- **Workflow Analysis:** Reviewing existing banking processes and workflows to identify areas for optimization and automation.
- **Resource Allocation:** Allocating resources effectively, including personnel, infrastructure, and technology, to support the implementation and operation of the online banking system.

- **User Feedback:** Gathering feedback from beta testers and pilot users to iterate and improve the system based on real-world usage and experiences.

Chapter 4

Methodology

Methodology

Development using the Waterfall Model

The Waterfall Model is a sequential software development process model that follows a linear and non-iterative approach. It consists of distinct phases, each building upon the outcomes of the previous phase. Below is the methodology for developing an online banking system using the Waterfall Model, along with a diagram illustrating the process:

1. Requirements Analysis:

- In this phase, the project requirements are gathered and analyzed in detail.
- Requirements are documented in a comprehensive manner, covering functional and non-functional aspects of the online banking system.
- Stakeholder meetings, interviews, and surveys are conducted to ensure all requirements are captured accurately.

2. System Design:

- Based on the requirements gathered, the system architecture and design are formulated.
- The design phase involves creating detailed specifications for the user interface, database structure, and system components.
- Design documents, such as system blueprints, wireframes, and data flow diagrams, are produced to guide the development process.

3. Implementation:

- Once the system design is finalized, the actual coding and development of the online banking system begin.
- Programmers write code according to the specifications outlined in the design documents.
- Development tasks are divided into modules, and coding standards and best practices are adhered to throughout the implementation phase.

4. Testing:

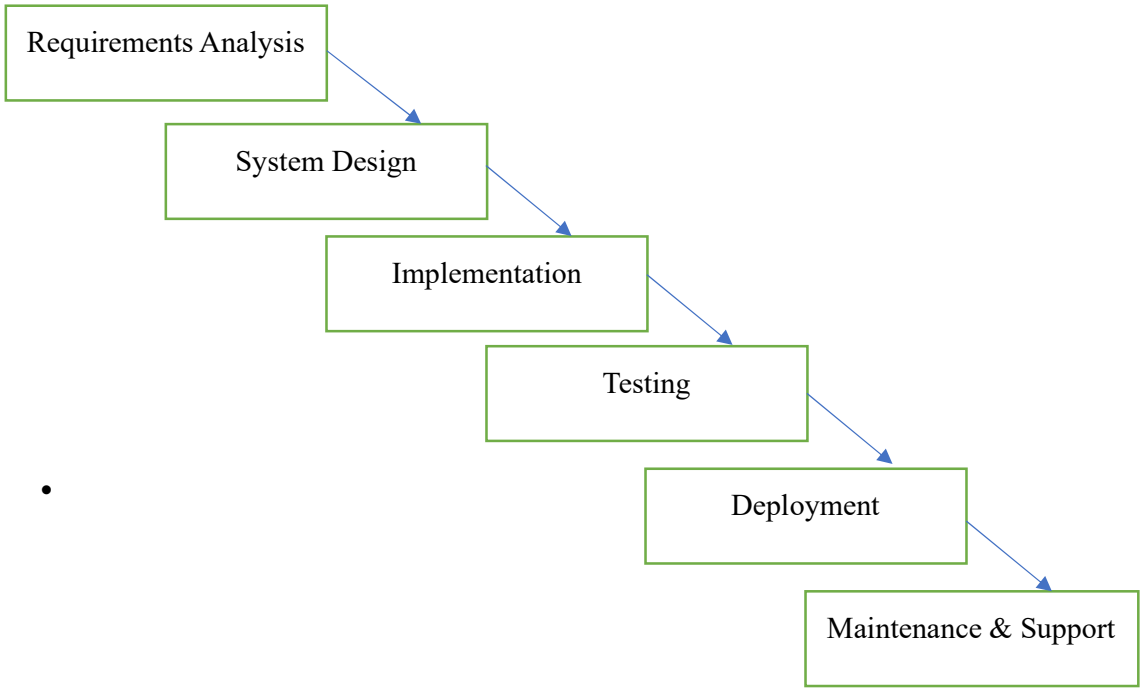
- The testing phase involves the verification and validation of the developed system to ensure it meets the specified requirements.
- Various types of testing, including unit testing, integration testing, system testing, and user acceptance testing (UAT), are performed.
- Test cases are designed to cover all functional and non-functional aspects of the online banking system, and any defects or issues identified are addressed promptly.

5. Deployment:

- Once testing is complete and the system is deemed ready for production, it is deployed to the live environment.
- Deployment involves installing the online banking system on the server infrastructure and configuring it for use by end-users.
- User accounts and data are migrated from existing systems, and users are provided with access to the new online banking platform.

6. Maintenance and Support:

- After deployment, the online banking system enters the maintenance phase, where it is monitored and supported on an ongoing basis.
- Routine maintenance tasks, such as software updates, bug fixes, and performance optimizations, are performed as needed.
- User feedback and system usage data are collected to identify areas for improvement and future enhancements.



Chapter 5

System Requirements

System Requirements

Software Requirements

Processor	Pentium 2
Clock Speed	800 MHZ
System BUS	64Bit
RAM	256mb
HDD	3gb
Monitor	SVGA color
Keyboard	108 keys
Modem	56kbps`
Mouse	Logitech

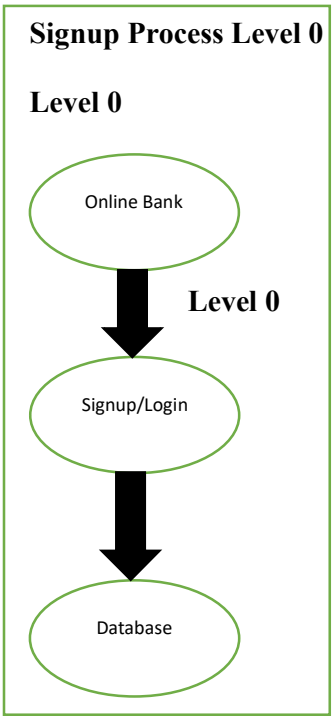
Software Requirements

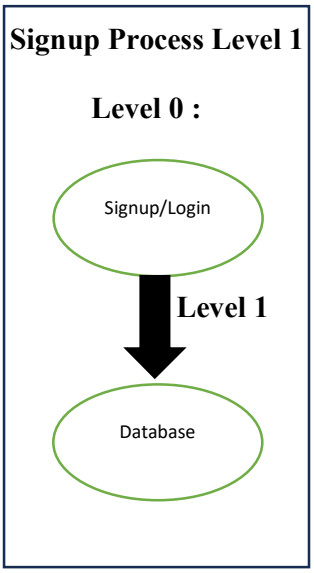
Operating System	Windows 7
Browser	Internet Explorer 5.5 or any HTTP
Frontend	HTML/CSS & JS
DTABASELAYER	SQL/LOCAL DATABASE
CONNECTION	TCP/IP
PROTOCOL	HTTP,SMTP,POP3,WAP

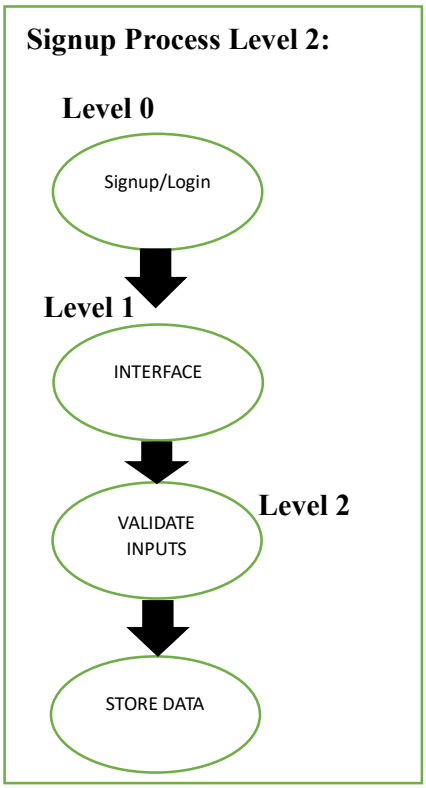
Chapter 6

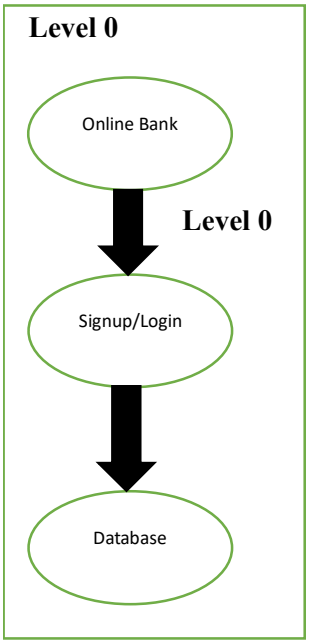
System Design

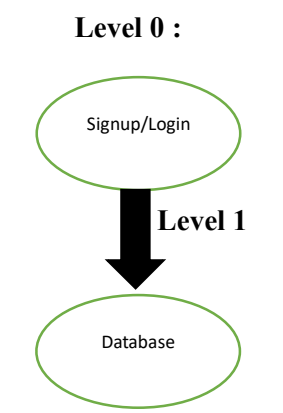
DFD

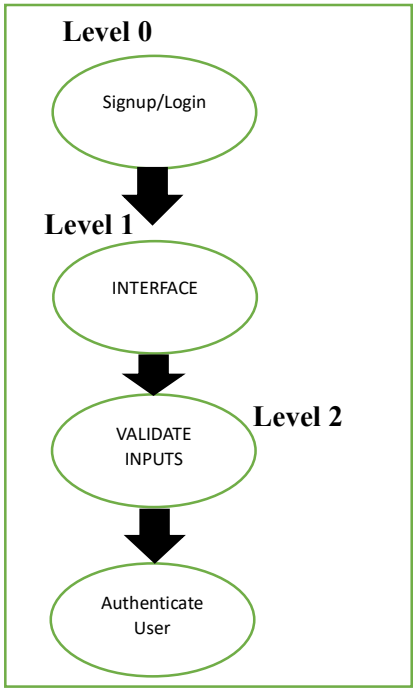

Signup Process Level 0
Level 0
Online Bank
Level 0
Signup/Login
Database


Signup Process Level 1
Level 0 :
Signup/Login
Level 1
Database


Signup Process Level 2:
Level 0
Signup/Login
Level 1
INTERFACE
VALIDATE INPUTS
Level 2
STORE DATA


Login Process Level 1
Level 0
Online Bank
Level 0
Signup/Login
Database


Login Process Level 1
Level 0 :
Signup/Login
Level 1
Database


Login Process Level 2:
Level 0
Signup/Login
Level 1
INTERFACE
VALIDATE INPUTS
Level 2
Authenticate User

In these diagrams:

- **0 Level**: Represents the core system where all transactions and operations occur.
- **Level 1**: Illustrates the distinct processes, such as signup or login, nested within the primary system.
- **Level 2**: Further refines the processes depicted at Level 1, offering a more detailed breakdown of sub-processes.
- **Signup/Login**: Denotes the interface or operation where users either register for a new account or authenticate into an existing one.
- **Database**: Serves as the repository for all user-related data, encompassing registration and login credentials.

These diagrams provide a hierarchical representation of the signup and login procedures within the online banking system, showcasing the flow of data among diverse components across various levels of abstraction. They offer insight into the granular functionalities and interactions within the system's architecture.

ER Diagram

An Entity-Relationship (E-R) diagram illustrates the entities within a system and the relationships between them. For an online banking system, the E-R diagram might include entities such as User, Account, Transaction, and Beneficiary. Here's a simplified E-R diagram:

Entities:

1. **User:**
    - Attributes: UserID, Username, Password, Email, FirstName, LastName, etc.

2. **Account:**

    - Attributes: AccountID, AccountNumber, Balance, AccountType, UserID (foreign key), etc.

3. **Transaction:**

    - Attributes: TransactionID, Amount, TransactionType, Timestamp, FromAccount, ToAccount, etc.

4. **Beneficiary:**

    - Attributes: BeneficiaryID, BeneficiaryName, AccountNumber, UserID (foreign key), etc.

Relationships:

1. **User - Account:**

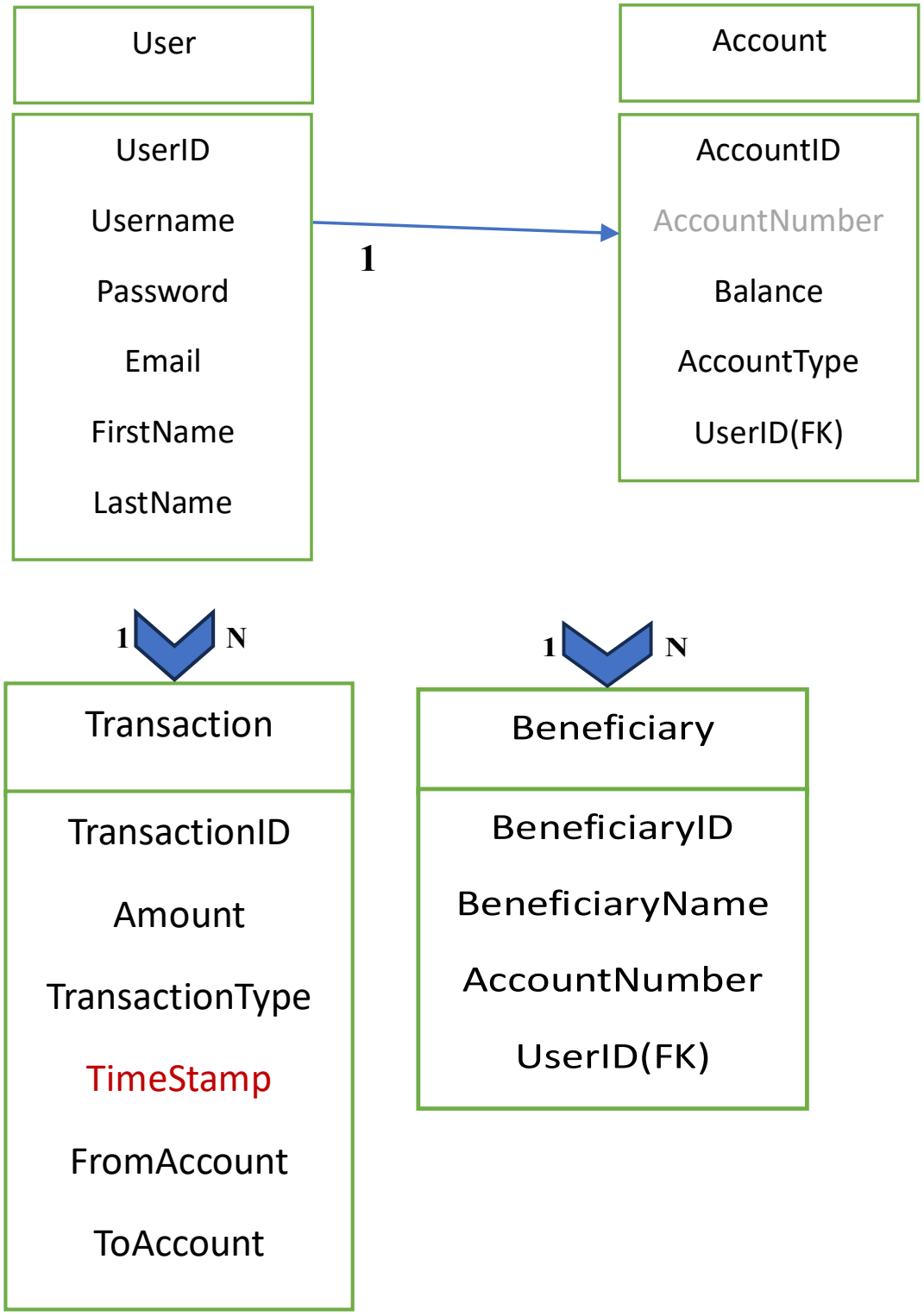
    - One user can have multiple accounts.
    - One account belongs to one user.
    - Relationship: One-to-Many (1:N)

2. **Account - Transaction:**

    - One account can have multiple transactions.
    - One transaction is associated with one account.
    - Relationship: One-to-Many (1:N)

3. **User - Beneficiary:**

- One user can have multiple beneficiaries.
- One beneficiary belongs to one user.
- Relationship: One-to-Many (1:N)



In the diagram:

- **Primary keys** are denoted by underlining the attribute name.
- **Foreign keys** are indicated with (FK).
- Relationships are depicted using cardinality notation (1:N), where "1" indicates one entity and "N" indicates many entities.
- Each entity has its own set of attributes representing its properties.

This E-R diagram provides a visual representation of the entities and their relationships within the online banking system, aiding in the understanding and design of the database structure.

Modules

Modules for Demo Users:

1. **Homepage:**
    - Overview of the banking services available.
    - Access to basic information about the bank and its services.
2. **Demo Transactions:**
    - Simulated transactions for exploring the system.
    - Examples include fund transfers, bill payments, and balance inquiries.
3. **Demo Account Management:**

- Allows users to create a mock account.
- Features like adding beneficiaries, updating personal information, and setting preferences for a simulated account.

4. **Demo Customer Support:**
    - Provides access to FAQ, tutorials, and guides.
    - Simulated customer support chat or messaging system for inquiries.

Modules for Registered Users:

1. **Login/Authentication:**
    - Secure login process for registered users.
    - Authentication mechanisms like username/password, biometrics, or OTP.
2. **Account Overview:**
    - Dashboard displaying account balances, recent transactions, and account summary.
    - Access to account statements and transaction history.
3. **Fund Transfer:**
    - Facilitates transfers between user's accounts or to other bank accounts.
    - Options for one-time transfers, recurring transfers, and scheduled payments.
4. **Bill Payment:**

- Enables users to pay bills for utilities, credit cards, loans, etc.
- Integration with biller networks for seamless payments.

5. **Account Management:**
    - Allows users to update personal information, change passwords, and manage account settings.
    - Options for adding, editing, or deleting beneficiaries.

Chapter 7

Brief Description of Pages

Brief Description of pages

In this academic project, we are developing an Online Banking Website using basic HTML, CSS, and JavaScript. We will adhere to the design provided by the instructor, which includes the following features:

- A homepage featuring essential views and links to various pages such as login/signup, services, fixed deposit, e-services, account details, and about us.
- The top bars will contain links to all linked pages.
- Services offered include Personal internet banking, Corporate banking, Merchant business, and Online tax.
- Payment transfer functionalities include International transfer, National transfer, and Bill payment.
- The My Account page will display Account Number, Bank/Branch, and IFSC Code details.
- An ABOUT US page will showcase our contact details and headquarters.

We have employed HTML programming concepts such as Hyperlink projection and utilized different tags for improved visual presentation, including font, table, listing, hyperlink, and styling. We've incorporated elements such as Combo Box, Search Box, Button, Text Field, Label, Fieldset, and Legend to enhance user interaction.

This project serves as an online banking platform allowing users to create bank accounts and access various banking services.

CONCEPTS USED

This website is made up of HTML, CSS, and JavaScript.

The concept used in HTML:

1. Using the link to add an external CSS file to it.
2. Using the “src” attribute with the “script” tag we link external JavaScript.
3. Using the “div” tag to separate various parts of the webpage to give functionality and style separately.
4. Using “h1” tags to give a heading.
5. Using “img” tag to give images on the webpage.
6. Using anchor (“a”) tags to link multiple HTML files to each other.
7. Using “id” and “class” attributes with different tags to give them style and functionality accordingly.
8. Using the “form” tag to use the form for the login and sign-up page.
9. Using the “method” attribute’s value as “POST” to give the form data securely.
10. Using “input” tags to make an input field for the users. (With its different types.)
11. Using “onload”, “onclick”, “onblur”, “onchange” with its value to call the functions which are in the respective JavaScript files on loading the webpage, on clicking a button, on losing focus, and changing the state respectively.
12. Using “label” tags to create labels for the input fields.
13. Using “select” tags to create a drop-down menu or combo box in the form.
14. Using “option” tags inside the “select” tag to give options in the drop-down menu.
15. Using “button” tags to create a button for submitting and resetting the form.

The concept used in CSS:

1. Using “#” and “.” to access elements with their id and class values respectively.
2. Using some pseudo-classes to make a somewhat interactive web page. (Without using JS.)
3. Using media queries to make the page responsive for some specific display ratios.
4. Using “@import” for importing another CSS file to a specific file.

The concept used in JavaScript:

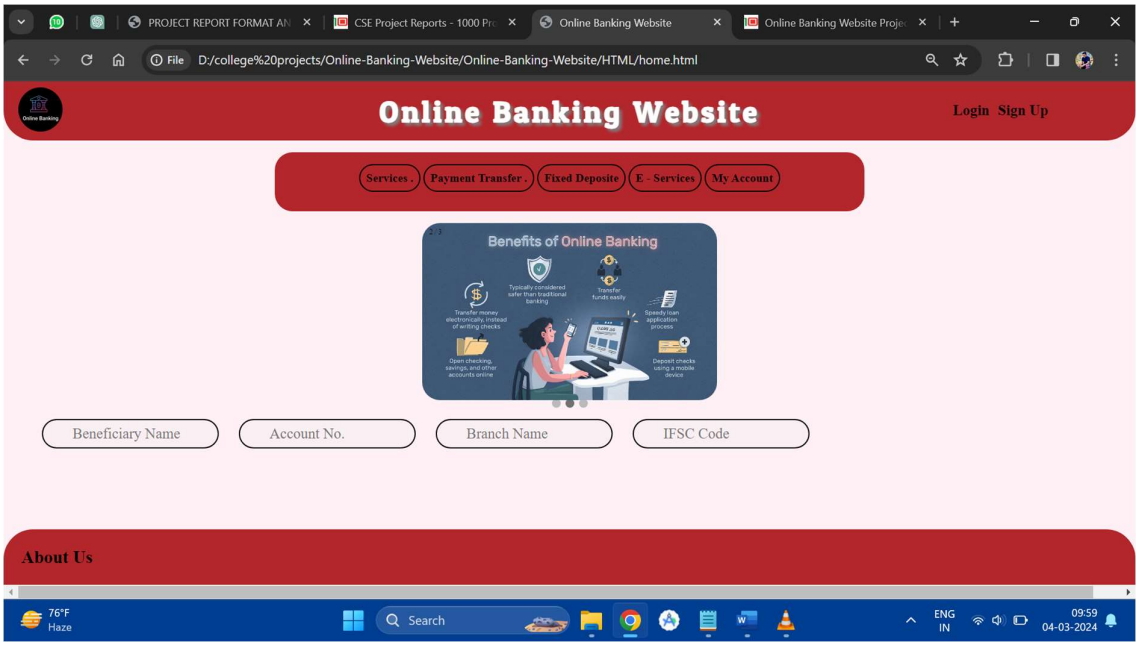
(In this project, JavaScript is used mainly for form validation and for interactive buttons.)

1. Using Regular Expression (RegEx) for creating a pattern for email and password validations as they have a specific pattern in them.
2. Using “window.location.replace()” to go to a specific location on calling it.
3. And using other pre-defined functions to create our user-defined functions.

Chapter 8

Coding with Screenshots

Home Page



HOME.html

```
<!DOCTYPE html>
<html>
   <head>
      <title>Online Banking Website</title>
      <link href="../CSS/homeCSS.css" rel="stylesheet">
      <link href="https://fonts.googleapis.com/css2?family=Patua+One&display=swap"
rel="stylesheet">
      <!-- <script src="../Java Script/homeJS.js"></script> -->
   </head>
   <body>
      <div id="header" class="head">
         <h1>Online Banking Website</h1>
        <img src="../Images/logo.png" alt="Online Banking System"
              title="Online Banking Website">
      </div>
      <div id="log_sign">
         <li><a href="login.html">Login</a></li>
         <li><a href="signup.html">Sign Up</a></li>
      </div>
      <div id="navbar">
         <div id="nav">
            <div id="services">
               <li><a href="#">Services .</a></li>
               <nav class="content">
```

```
                <li><a href="perIntBank.html">Personal Internet Banking</a></li>
                <li><a href="corpBank.html">Corprorate Banking</a></li>
                <li><a href="merBusiness.html">Merchant Bussiness</a></li>
                <li><a href="onlTax.html">Online Tax</a></li>
            </nav>
        </div>
        <div id="payment_transfer">
            <li><a href="#">Payment Transfer .</a></li>
            <nav class="content">
                <li><a href="intTransfer.html">International Transfer</a></li>
                <li><a href="natTransfer.html">National Transfer</a></li>
                <li><a href="billPay.html">Bill Payment</a></li>
            </nav>
        </div>
        <div id="fixed_deposite">
            <li><a href="fixDep.html">Fixed Deposite</a></li>
        </div>
        <div id="e_service">
            <li><a href="eServ.html">E - Services</a></li>
        </div>
        <div id="acc">
            <li><a href="acc.html">My Account</a></li>
        </div>
    </div>
</div>
<div class="slideshow-container">
    <div class="mySlides fade">
        <div class="numbertext">1 / 3</div>
        <img src="../Images/online banking-1.png">
    </div>
    <div class="mySlides fade">
        <div class="numbertext">2 / 3</div>
        <img src="../Images/online banking-2.png">
    </div>
    <div class="mySlides fade">
        <div class="numbertext">3 / 3</div>
        <img src="../Images/online banking-3.png">
    </div>
</div>
<br>
<div style="text-align:center">
    <span class="dot"></span>
    <span class="dot"></span>
    <span class="dot"></span>
</div>
<script>
    var slideIndex = 0;
    showSlides();
    function showSlides()
    {
```

```
            var i;
            var slides = document.getElementsByClassName( "mySlides" );
            var dots = document.getElementsByClassName( "dot" );
            for ( i = 0; i < slides.length; i++ ) {
               slides[ i ].style.display = "none";
            }
            slideIndex++;
            if ( slideIndex > slides.length ) { slideIndex = 1 }
            for ( i = 0; i < dots.length; i++ ) {
               dots[ i ].className = dots[ i ].className.replace( " active", "" );
            }
            slides[ slideIndex - 1 ].style.display = "block";
            dots[ slideIndex - 1 ].className += " active";
            setTimeout( showSlides, 2000 ); // Change image every 2 seconds
         }
      </script>
      <div id="disp">
         <p>Login to see your Details here:</p>
         <div>
            <input type="text" disabled name="name" id="name" placeholder="Beneficiary
Name" value="">
            <input type="number" disabled name="accno" id="accno" value=""
placeholder="Account No.">
            <input type="text" disabled name="branch" id="branch" value=""
placeholder="Branch Name">
            <input type="text" disabled name="ifsc" id="ifsc" value="" placeholder="IFSC
Code">
         </div>
      </div>
      <div id="foot">
         <a href="aboutUs.html">About Us</a>
      </div>
   </body>
</html>
```

Home.css

```
/*for Header and Login and Signup*/
*{
   box-sizing:border-box;
   font-family: ';
}

body{
   background-color: lavenderblush;
   margin: 0px;
}

h1{
   text-align: center;
   font-family: 'Patua One', cursive;
```

```
    font-weight:bolder;
    height:75px;
    word-spacing: 1.5px;
    letter-spacing: 5px;
    padding-top:20px;
    margin: 0px;
    font-size:50px;
    text-shadow: 5px 5px 5px grey;
}
.head{
    position: fixed;
    top:0px;
}
#header img{
    height:75px;
    border-radius:20px;
    position:absolute;
    top:10px;
    left:25px;
}
#header{
    background-color: #b3262c;
    color: #fff;
    height:100px;
    width:100%;
    margin-bottom: 10px;
    border-bottom-left-radius: 50px;
    border-bottom-right-radius: 50px;
}
#log_sign{
    padding-top:25px;
    width:300px;
    position:absolute;
    top:10px;
    right:10px;
    font-weight:bolder;
}
#log_sign li{
    list-style: none;
    display: inline;
    margin-right: 10px;
}
#log_sign li a{
    text-decoration:none;
    color:black;
    font-size:25px;
}
#log_sign li a:hover{
    text-decoration: underline;
    border-radius:30px;
```

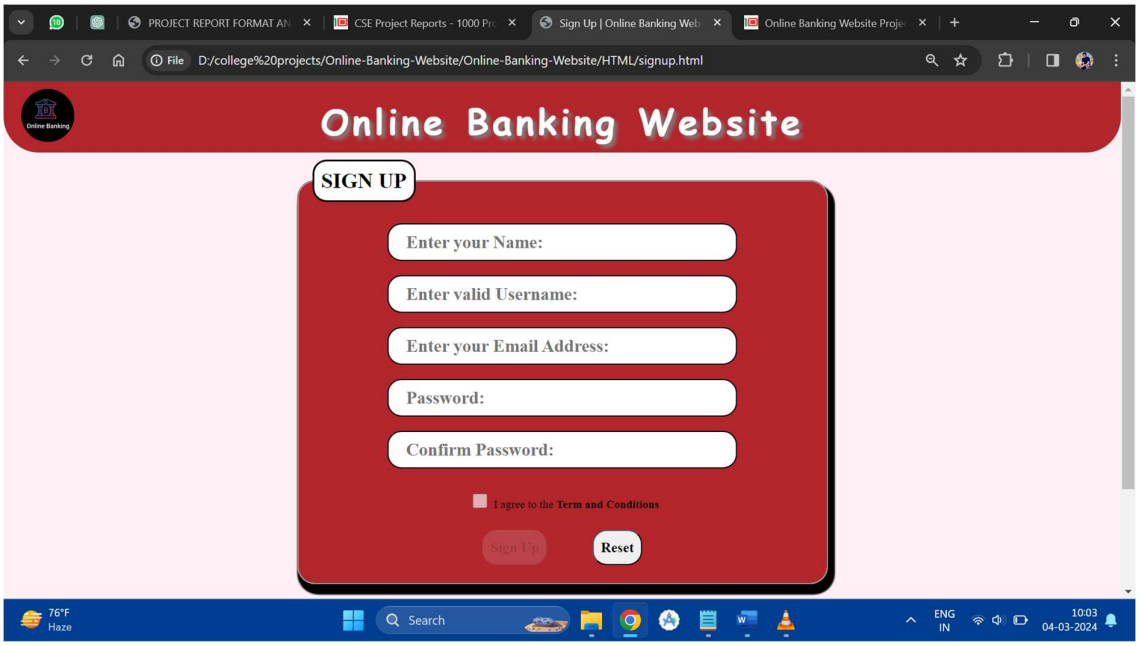
```
    border:3px solid black;
    padding:10px;
    transition-duration: 0.20s;
}
a{
    color: black;
    font-weight: bolder;
    text-decoration: none;
}
/*
    for small screen
*/
@media(max-width:1340px) {
    #header h1{
        display: none;
    }
    #header{
        width:100%;
    }
    img{
        position:static;
        margin:0px 0px;
        display: inline-block;
        width:300px;
    }
    #log_sign{
        position:absolute;
        top:10px;
        right:25px;
        display: inline-block;
        width:200px;
    }
}

/*For Nav Bar*/
.content{
    display: none;
}
div li{
    list-style: none;
    display: block;
}
#navbar div{
    display: inline-block;
}
#navbar{
    width:100%;
    display: flex;
    justify-content:space-evenly;
    flex-direction: row;
```

```
    background-color: #b3262c;
    padding:20px;
    margin:120px auto 0px;
    width:1000px;
    font-size: 20px;
    border-radius:30px;
    height:100px;
}
#nav div{
    border:2px solid black;
    border-radius: 30px;
    text-align: center;
    padding:10px;
}

#nav div:hover{
    background-color: #fff;
    transition-property: "background-image";
    transition-duration: 0.25s;
}

/*For content drop down*/
.content{
    display: none;
    position:absolute;
    margin-top:15px;
    z-index: 1;
    text-align: left;
    border:2px solid black;
    border-radius: 20px;
    padding:10px;
   background-color: #fff;
    color:white;
}
.content li{
    border-bottom: 2px dashed lightslategray;
    padding:0px 10px;
}
.content li:last-child{
    border-bottom:none;
}
.content li:hover{
    background-color: #b3262c;
    border-radius:30px;
}
#nav div:hover .content{
    display: block;
}

/*for about us */
```

```
#foot{
    position: fixed;
    bottom:0px;
    background-color: #b3262c;
    border-top-left-radius: 50px;
    border-top-right-radius: 50px;
    display: block;
    width: 100%;
    padding:30px;
}
#foot a{
    font-size:30px;
    font-weight: bolder;
}
#foot a:hover{
    border-radius:30px;
    border:3px solid black;
    padding:5px;
}
div#disp{
    position: relative;
    left:50px;
}
@media(max-height:700px){
    div#disp{
        position: relative;
        top:50px;
        left:50px
    }
}
#disp dev{
    display: flex;
    width:100%;
    justify-content: space-evenly;
}
#disp p{
    display: none;
}
input{
    box-sizing:border-box;
    height:50px;
    border-radius:30px;
    border:2px solid black;
    padding:20px;
    font-size:25px;
    padding-left:50px;
    width:300px;
    text-overflow: ellipsis;
    margin:15px;
    color:black;
```

```
}

/* Slideshow pictures */
.mySlides
{
   display: none;
   z-index:0;
}
.slideshow-container img
{
   vertical-align: middle;
   height:300px;
   width:500px;
   border-radius: 30px;
   z-index: 0;
}

/* Slideshow container */
.slideshow-container {
   position: relative;
   max-width: 1000px;
   top:20px;
   left:250px;
   margin: 0px auto;
   z-index:0;
}

/* Caption text */
.text {
   color: #f2f2f2;
   font-size: 15px;
   padding: 8px 12px;
   position: absolute;
   bottom: 8px;
   width: 100%;
   text-align: center;
   z-index: 0;
   z-index:0;
}

/* Number text (1/3 etc) */
.numbertext {
   color:black;
   font-size: 12px;
   padding: 8px 12px;
   position: absolute;
   top: 0;
   z-index:0;
}
```

```
/* The dots/bullets/indicators */
.dot {
   height: 15px;
   width: 15px;
   margin: 0 2px;
   background-color: #bbb;
   border-radius: 50%;
   display: inline-block;
   transition: background-color 0.6s ease;
   z-index:0;
}

.active {
   background-color: #717171;
}
/* Fading animation */
.fade {
   -webkit-animation-name: fade;
   -webkit-animation-duration: 1.5s;
   animation-name: fade;
   animation-duration: 1.5s;
   z-index:0;
}

@-webkit-keyframes fade {
   from {opacity: .4}
   to {opacity: 1}
}

@keyframes fade {
   from {opacity: .4}
   to {opacity: 1}
}

/* On smaller screens, decrease text size */
@media only screen and (max-width: 300px) {
   .text {font-size: 11px}
}
```



Signup.html

```
<!DOCTYPE html>
<html lang="en">

  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Sign Up | Online Banking Website</title>
    <link href="../CSS/signupCSS.css" rel="stylesheet">
    <script src="../Java Script/signupJS.js"></script>
  </head>

  <body onload="load();">
    <div id="header">
      <h1>Online Banking Website</h1>
      <a href="home.html"><img src="..//Images//logo.png" alt="Online Banking Website" title="Online Banking Website"></a>
    </div>
```

```
        <form  method="POST">
            <fieldset>
                <legend>Sign Up</legend>
                <input type="text" id="name" placeholder="Enter your Name:"><br />
                <input type="text" id="username" placeholder="Enter valid Username:"
                    onblur="return validUserName();"><br />
                <input type="email" id="email" placeholder="Enter your Email Address:"><br />
                <input type="password" id="password" placeholder="Password:" onblur="return
validPassword();"><br />
                <input type="password" id="confirmPassword" placeholder="Confirm Password:"
                    onblur="return cnfrmPass();"><br />
                <p id="result"></p>
                <p><input type="checkbox" name="tnc" id="tnc" onchange="return validForm();"><span>I agree to
the <a
                        href="#">Term and Conditions</a></span></p>
                <button type="button" id="signupbtn" value="0" onclick="goto();">Sign Up</button>
                <button type="reset" id="resetbtn">Reset</button>
            </fieldset>
        </form>
    </body>


</html>
```

Signup.css

```
@import url("../CSS/homeCSS.css");
fieldset{
    border-radius:25px;
    padding:20px;
    background-color: #b3262c;
    text-align: center;
    width:750px;
```

```
    box-shadow:5px 10px 2px 5px;
    margin:10px auto;
}
form{
    height:800px;
}
legend{
    text-transform: uppercase;
    font-weight:bolder;
    font-size: 30px;
    text-align: end;
    border-radius:20px;
    border:3px solid black;
    background-color: #fff;
    padding:10px;
}
input{
    height:30px;
    border-radius:20px;
    padding:25px;
    font-size:25px;
    font-weight:bolder;
    margin:10px;
    width:70%;
}
input:focus{
    border:2px solid blue;
}
button{
    height:50px;
```

```
    padding:10px;
    border-radius:20px;
    border-style:outset;
    margin:5px 30px;
    font-size: 20px;
    font-weight: bolder;
}
/* span{
    position: relative;
    bottom:20px;
    margin:0px;
} */
input[type="checkbox"]{
    height:20px;
    width:20px;
}
span{

}
input[type="checkbox"]:checked{
    width:20px;
    transition: width 0.5s;
}

input[type="email"]:invalid{
    border:2px solid red;
}
input[type="number"]:invalid{
    border:2px solid red;
}
```

```
#result{
    width:100%;
    color:red;
    text-decoration: underline;
}
span{
    display: inline-block;
    margin-top:10px;
}
```

Signup.js

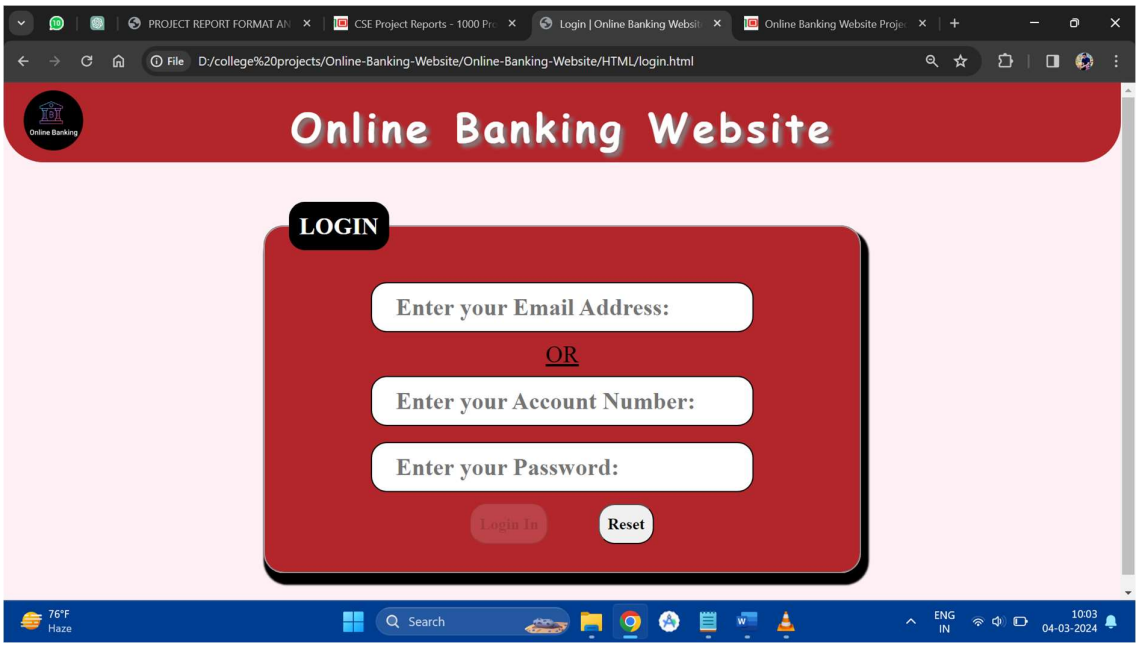
```
function load () {
    wrong();
    document.getElementById( "name" ).focus();
}
function wrong () {
    document.getElementById( "signupbtn" ).disabled = true;
    document.getElementById( "signupbtn" ).style.opacity = "0.5";
    document.getElementById( "tnc" ).disabled = true;
    document.getElementById( "tnc" ).checked = false;

}
function right () {
    document.getElementById( "result" ).innerHTML = "";
}
function validUserName () {
    var username = document.getElementById( "username" ).value;
    if ( /\s/.test( username ) )
    {
        wrong();
        document.getElementById( "result" ).innerHTML = "User name must not contain any space.";
```

```
        document.getElementById( "username" ).focus();
        return false;
    }
    else
    {
        right();
        return true;
    }
}

function validPassword () {
    var pass = document.getElementById( "password" ).value;
    if ( pass.length < 6 )
    {
        wrong();
        document.getElementById( "result" ).innerHTML = "Password should contain atleast 6 characters.";
        document.getElementById( "password" ).focus();
        document.getElementById( "password" ).style.border = "2px solid red";
        return false;
    }
    else if ( pass.length >= 6 )
    {
        if ( /(?=.*[\d])(?=.*[A-Z])(?=.*[a-z])(?=.*[!@#$%^&*])[\w!@#$%^&*]{8,}$/.test( pass ) == false )
        {
            wrong();
            document.getElementById( "result" ).innerHTML = "Password should contain Uppercase, lowercase,
and numeric charcaters. ";
            return false;
        }
        else
```

```
        {
            right();
            document.getElementById( "password" ).style.border = "none";
            return true;
        }
    }
}

function cnfrmPass () {
    var pass1 = document.getElementById( "password" ).value;
    var pass2 = document.getElementById( "confirmPassword" ).value;
    if ( pass1 != pass2 )
    {
        wrong();
        document.getElementById( "result" ).innerHTML = "Confirm Password must be same as the Password.";
        document.getElementById( "confirmPassword" ).focus();
        document.getElementById( "confirmPassword" ).blur();
        return false;
    }
    else
    {
        right();
        if ( validUserName() && validPassword() )
        {
            document.getElementById( "tnc" ).disabled = false;
            return true;
        }
        else
        {
            document.getElementById( "result" ).innerHTML = "There is something wrong in the form.";
```

```
            document.getElementById( "tnc" ).checked = false;
            return false;
        }
    }
}
function validForm () {
    if ( cnfrmPass() )
    {
        if ( document.getElementById( "tnc" ).checked == false )
        {
            document.getElementById( "tnc" ).disabled = false;
            document.getElementById( "signupbtn" ).disabled = true;
            document.getElementById( "signupbtn" ).style.opacity = "0.5";
            document.getElementById( "tnc" ).focus();
            return false;
        }
        else
        {
            document.getElementById( "signupbtn" ).style.opacity = "1";
            document.getElementById( "signupbtn" ).disabled = false;
            return true;
        }
    }
    else
    {
        wrong();
    }
}
function goto () {
    document.getElementById( "signupbtn" ).style.borderStyle = "inset";
```

```
    window.location.replace( "home.html" );
    return true;
}
```



Login.html

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8">
        <title>Login | Online Banking Website</title>
        <link href="../CSS/loginCSS.css" rel="stylesheet">
        <script src="../Java Script/loginJS.js"></script>
    </head>

    <body onload="load();">
        <div id="header">
            <h1>Online Banking Website</h1>
            <a href="home.html"><img src="..//Images//logo.png" alt="Online Banking Website"
                    title="Online Banking Website"></a>
        </div>
```

```
        <form method="POST">
            <fieldset>
                <legend>Login</legend>
                <input type="email" placeholder="Enter your Email Address:" id="email"><br />
                <p>OR</p>
                <input type="text" placeholder="Enter your Account Number:" id="accno"><br />
                <input type="password" accept="number" placeholder="Enter your Password:" id="password"
                    onblur="return validForm();"><br />
                <p id="result"></p>
                <button type="button" id="loginbtn" value="0" onclick="goto();">Login In</button>
                <button type="reset" id="resetbtn">Reset</button>
            </fieldset>
        </form>
    </body>

</html>
```

Login.css

```
@import url("../CSS/homeCSS.css");
fieldset{
    border-radius:25px;
    padding:30px;
    background-color: #b3262c;
    text-align: center;
    width:750px;
    box-shadow:5px 10px 2px 5px;
    margin:50px auto;
}
legend{
    text-transform: uppercase;
    font-weight:bolder;
```

```
    font-size: 30px;
    text-align: end;
    border-radius:20px;
    border:3px solid black;
    background-color: #000;
    color: #fff;
    padding:10px
}
input{
    height:50px;
    border-radius:20px;
    padding:30px;
    font-size:30px;
    font-weight:bolder;
    margin:10px;
    width: 70%;
}
input:focus{
    border:2px solid blue;
}
form p{
    font-size:30px;
    margin:0px;
    text-decoration: underline;
}
button{
    height:50px;
    padding:10px;
    border-radius:20px;
    border-style: outset;
```

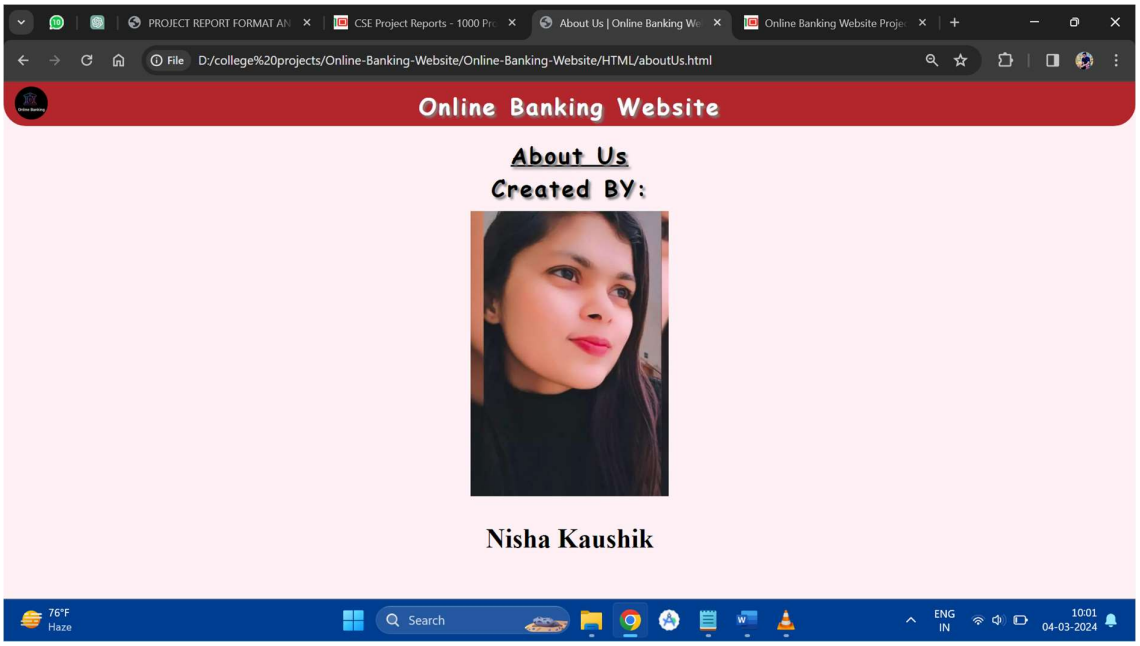
```
    margin:5px 30px;
    font-size: 20px;
    font-weight: bolder;
}
#result{
    font-size:10px;
    color:red;
    font-style:oblique;
}

#accno:invalid{
    border:2px solid red;
}
input[type="email"]:invalid{
    border:2px solid red;
}
```

Login.js

```
function load () {
    wrong();
    document.getElementById( "accno" ).value = "";
    document.getElementById( "email" ).value = "";
}
function retEmail () {
    return document.getElementById( "email" ).value;
}
function wrong () {
    document.getElementById( "loginbtn" ).disabled = true;
    document.getElementById( "accno" ).focus();
    document.getElementById( "loginbtn" ).style.opacity = "0.5";
}
```

```
function validForm () {
    var email = document.getElementById( "email" ).value;
    var accno = document.getElementById( "accno" ).value;
    var pass = document.getElementById( "password" ).value;
    if ( email.length == 0 )
    {
        if ( /[0-9]/.test( accno ) == true && ( accno.length != 16 ) )
        {
            document.getElementById( "result" ).innerHTML = "Account No. must be of 16 numeric characters.";
            wrong();
            return false;
        }
        else
        {
            if ( pass.length < 6 )
            {
                document.getElementById( "result" ).innerHTML = "Password must contain more than 6 charcaters.";
                wrong();
                return false;
            }
            else if ( pass.length >= 6 )
            {
                document.getElementById( "loginbtn" ).disabled = false;
                document.getElementById( "result" ).innerHTML = "";
                document.getElementById( "loginbtn" ).style.opacity = "1";
                return true;
            }
        }
    }
    else
```

```
    {
        if ( pass.length < 6 )
        {
            document.getElementById( "result" ).innerHTML = "Password must contain more than 6 charcaters.";
            wrong();
            return false;
        }
        else if ( pass.length >= 6 )
        {
            document.getElementById( "loginbtn" ).disabled = false;
            document.getElementById( "result" ).innerHTML = "";
            document.getElementById( "loginbtn" ).style.opacity = "1";
            return true;
        }
    }
}

function goto () {
    document.getElementById( "loginbtn" ).style.borderStyle = "inset";
    window.location.replace( "home.html" );
    return true;
}
```



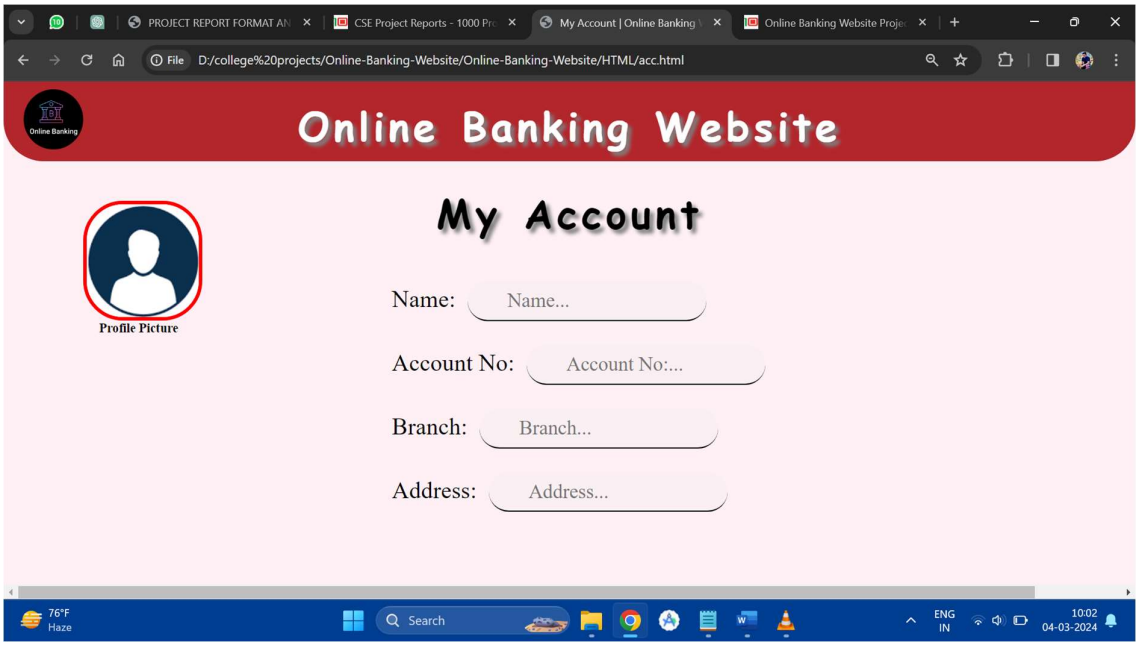
AboutUs.html

```
<!DOCTYPE html>
<html lang="en">


  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>About Us | Online Banking Website</title>
    <link href="../CSS/aboutUsCSS.css" rel="stylesheet">
  </head>


  <body>
    <div id="header">
      <h1>Online Banking Website</h1>
      <a href="home.html"><img src="../Images/logo.png" alt="Online Banking System"
          title="Online Banking Website"></a>
    </div>
    <div>
      <h1 style="text-decoration: underline;">About Us</h1>
```

```
        <h1>Created BY:</h1>
        <div id="creator">
            <img src="https://i.imgur.com/I8QHisi.jpeg">
            <h2>Nisha Kaushik</h2>

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

</html>
```

AboutUs.css

```
@import url("homeCSS.css");
#creator{
   margin:30px auto;
   width:1000px;
   text-align: center;
   font-size:40px;
   font-family: 'Courier New', Courier, monospace;
}
```



Acc.html

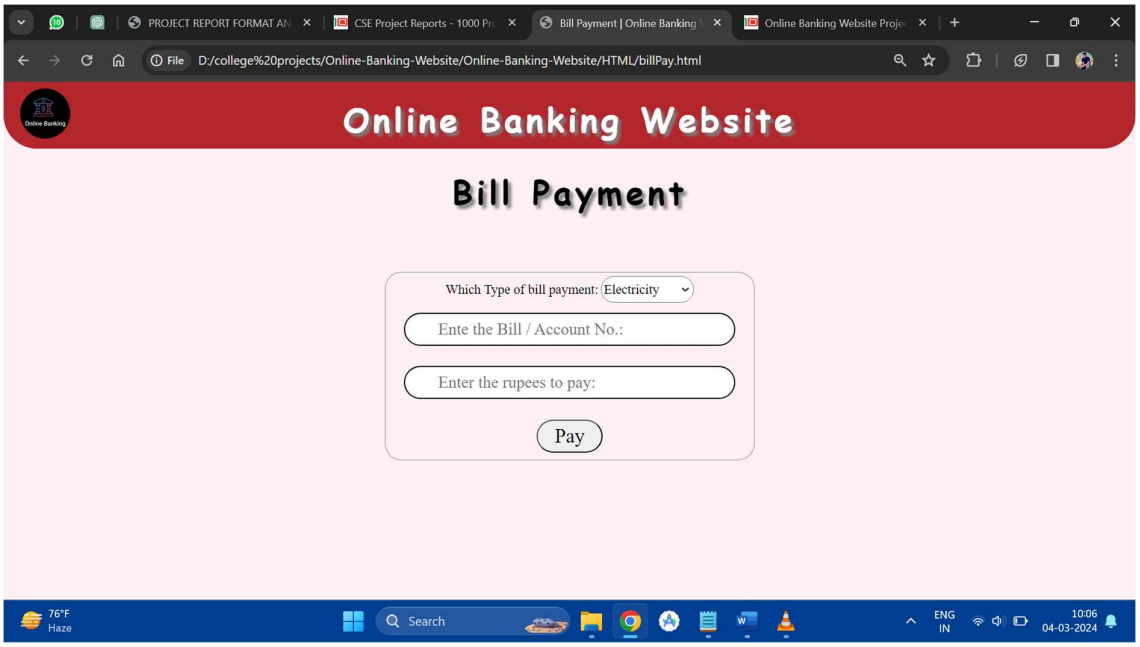
```
<!DOCTYPE html>
<html lang="en">

  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>My Account | Online Banking Website</title>
    <link href="../CSS/accCSS.css" rel="stylesheet">
  </head>

  <body>
    <div id="header">
      <h1>Online Banking Website</h1>
      <a href="home.html"><img src="../Images/logo.png" alt="Online Banking System"
          title="Online Banking Website"></a>
    </div>
    <div>
      <h1>My Account</h1>
    </div>
    <div id="pic">
      <img src="../Images/man.png">
      <p>Profile Picture</p>
    </div>
    <div id="info">
      <label>Name:</label><input type="text" disabled placeholder="Name..."><br/>
      <label>Account No:</label><input type="text" disabled placeholder="Account No:..."><br/>
      <label>Branch:</label><input type="text" disabled placeholder="Branch..."><br/>
      <label>Address:</label><input type="text" disabled placeholder="Address..."><br/>
    </div>
```

```
    </body>


</html>
```

Acc.css

```
@import url("homeCSS.css");
#pic img{
    position: absolute;
    top:150px;
    left:100px;
    border:5px solid red;
    border-radius:50px;
    height: 150px;
    width: 150px;
    text-align: center;
}
#pic img{
    height:150px;
}
#pic p{
    position: relative;
    top:100px;
    left:120px;
    font-weight: bolder;
}
#info{
    position: absolute;
    right:10px;
    width:65%;
}
input{
```

```
    border:none;
    border-radius:30px;
    border-bottom: 2px solid black;

}
label{
    font-size: 30px;
}
```



Billpay.html

```
<!DOCTYPE html>
<html lang="en">

    <head>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <title>Bill Payment | Online Banking Website</title>
        <script src="../Java Script/billPayJS.js"></script>
        <link rel="stylesheet" href="../CSS/billPayCSS.css">
    </head>
```

```
<body>
    <div id="header">
        <h1>Online Banking Website</h1>
        <a href="home.html"><img src="../Images/logo.png" alt="Online Banking System"
            title="Online Banking Website"></a>
    </div>

    <div>
        <h1>Bill Payment</h1>
        <form action="" method="POST">
            <fieldset>
                <label>Which Type of bill payment:</label>
                <select name="type">
                    <div>
                        <option value="Electricity">Electricity</option>
                        <option value="Water">Water</option>
                        <option value="Gas">Gas</option>
                    </div>
                </select><br />
                <input type="text" size="16" id="billno" placeholder="Ente the Bill / Account No.:"><br />
                <input type="number" id="rupee" placeholder="Enter the rupees to pay:" min="1"
max="20000"><br />
                <p id="error"></p>
                <button type="button" id="paybtn" onclick="goto();">Pay</button>
            </fieldset>
        </form>
    </div>
</body>
```

```
</html>
```

Billpay.css

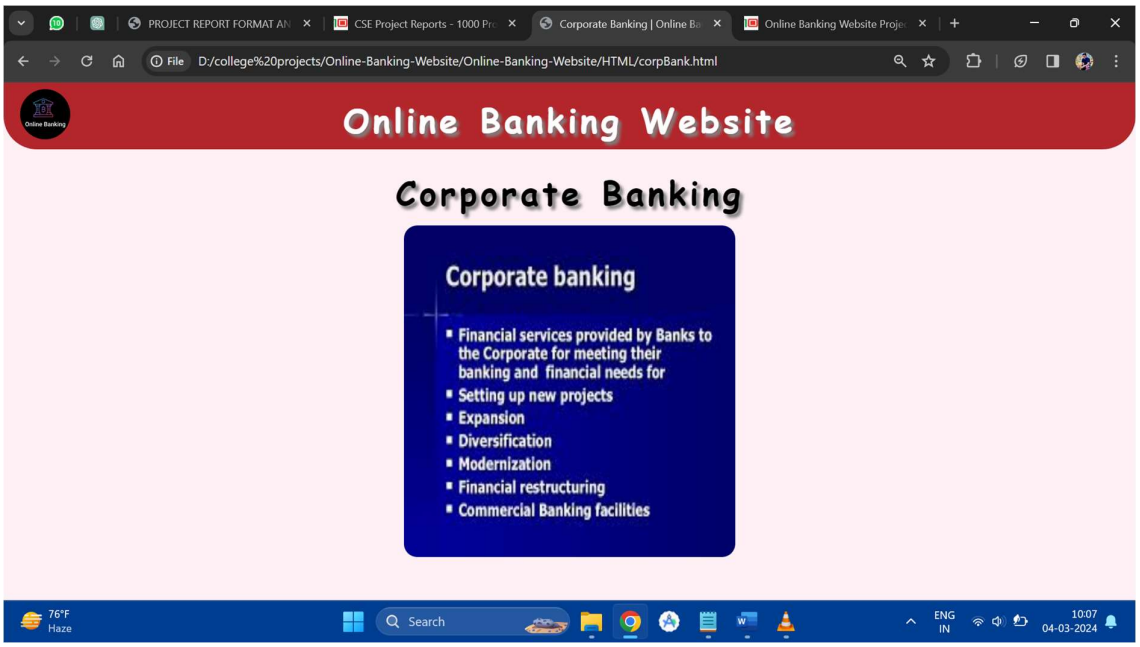
```
@import url("homeCSS.css");
fieldset{
    width: 500px;
    border-radius:25px;
    margin:100px auto;
    text-align: center;
    border:2px ridge;
}
label{
    font-size:20px;

}
select{
    height: 40px;
    width:140px;
    font-size:20px;
    border-radius:30px;
}
input{
    width:500px;
}
option{
    padding:30px;
    border-radius: 30px;
    background-color:black;
    color:white;
}
```

```
button{
    height:50px;
    width:100px;
    border-radius:30px;
    border-style: outset;
    font-size:30px;
}
input:focus{
    border:2px solid blue;
}
input[type="number"]:out-of-range{
    border:2px solid red;
}
```

Billpay.js

```
function goto () {
    document.getElementById( "paybtn" ).style.borderStyle = "inset";
    if
(document.getElementById("billno").value.length==0||document.getElementById("rupee").value.length==0)
    {
        document.getElementById( "error" ).innerHTML = "Above Fields are blank.";
        document.getElementById( "error" ).style.color = "red";
        document.getElementById( "error" ).style.textDecoration = "underline";
        document.getElementById( "paybtn" ).style.borderStyle = "outset";
    }
    else
    {
        document.getElementById( "error" ).innerHTML = "";
        alert( "Bill had paid successfully!!." );
        window.location.replace( "home.html" );
```

```
    }
}
```



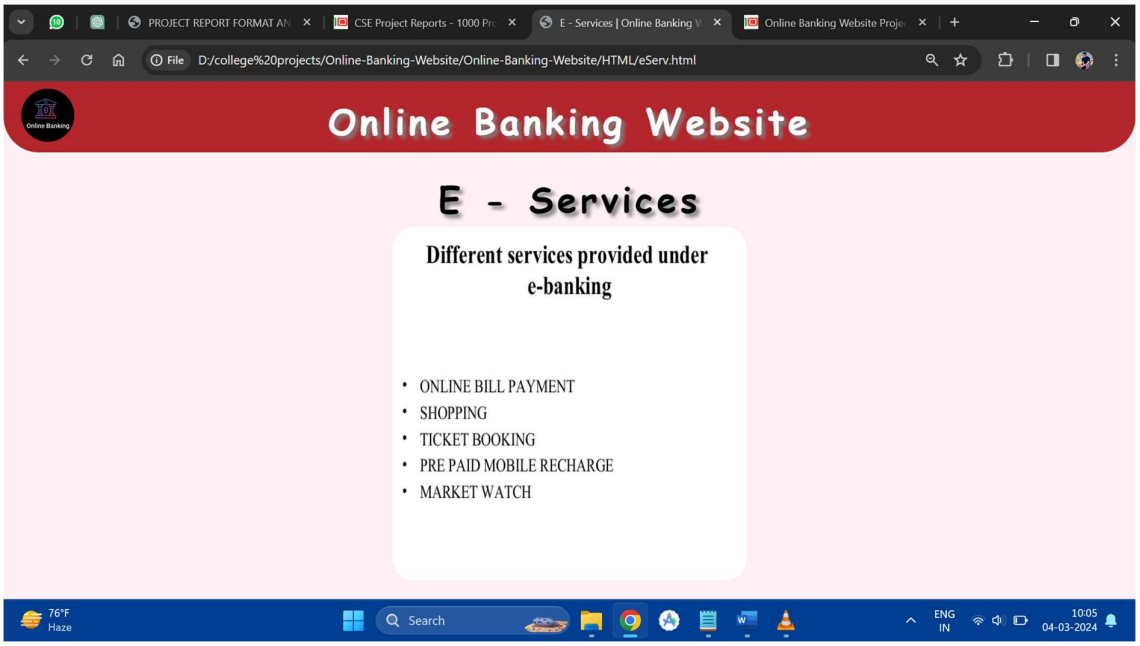
Corpbank.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Corporate Banking | Online Banking Website</title>
    <link href="../CSS/corpBankCSS.css" rel="stylesheet">
    <script src="../Java Script/corpBankJS.js"></script>
</head>
<body>
    <div id="header">
        <h1>Online Banking Website</h1>
        <a href="home.html"><img src="..//Images//logo.png" alt="Online Banking Website" title="Online Banking Website"></a>
    </div>
```

```
    <div>
        <h1>Corporate Banking</h1>
        <div id="corp">
            <!-- <img src="../Images/corporate.jpg" alt="Corporate Banking" title="Corportae Banking"> -->
        </div>
    </div>
</body>
</html>
```

Corpbank.css

```
@import url("../CSS/homeCSS.css");
#corp{
    background-image: url("../Images/corporate.jpg");
    background-size: 500px 500px;
    overflow: auto;
    margin:30px auto;
    border-radius:20px;
    height:500px;
    width:500px;
}
```



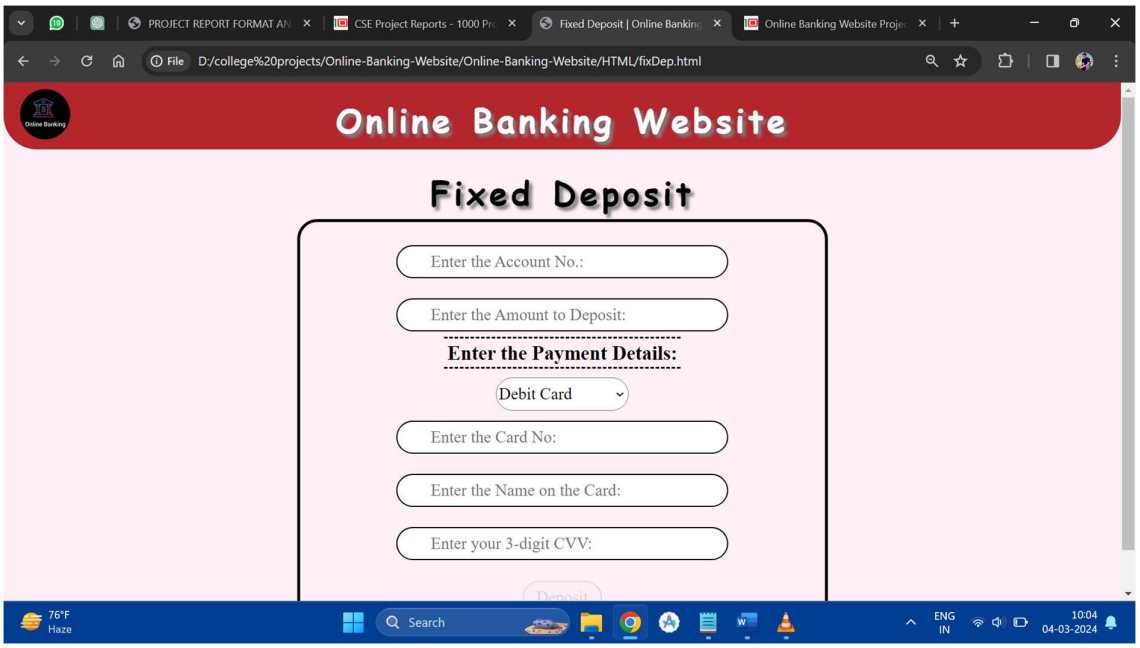
Eserv.html

```
<!DOCTYPE html>
<html lang="en">

  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>E - Services | Online Banking Website</title>
    <link href="../CSS/eServCSS.css" rel="stylesheet">
  </head>

  <body>
    <div id="header">
      <h1>Online Banking Website</h1>
      <a href="home.html"><img src="../Images/logo.png" alt="Online Banking System"
          title="Online Banking Website"></a>
    </div>
    <div>
      <h1>E - Services</h1>
      <div id="eserv"></div>
    </div>
  </body>

</html>
```

Eserv.css

```
@import url("homeCSS.css");

#eserv{
  background-image: url("../Images/e-services.jpg");
  height:500px;
```

```
    width:500px;
    background-size: 500px 500px;
    margin: 20px auto;
    border-radius: 30px;
}
```



Fixdep.html

```
<!DOCTYPE html>
<html lang="en">

    <head>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <title>Fixed Deposit | Online Banking Website</title>
        <link href="../CSS/fixDepCSS.css" rel="stylesheet">
        <script src="../Java Script/fixDepJS.js"></script>
    </head>

    <body onload="load();">
        <div id="header">
```

```
        <h1>Online Banking Website</h1>
        <a href="home.html"><img src="../Images/logo.png" alt="Online Banking System"
            title="Online Banking Website"></a>
    </div>
    <div>
        <h1>Fixed Deposit</h1>
        <form method="POST">
            <input type="text" id="accno" maxlength="16" placeholder="Enter the Account No.:" onblur="return
validAcc();"><br />
            <input type="number" id="rupee" max="50000" placeholder="Enter the Amount to Deposit:"><br />
            <label>Enter the Payment Details:</label><br />
            <select name="cards">
                <option value="credit">Credit Card</option>
                <option value="debit" selected>Debit Card</option>
            </select><br />
            <input type="text" id="cardno" maxlength="16" placeholder="Enter the Card No:" onblur="return
validCard();"><br />
            <input type="text" id="name" placeholder="Enter the Name on the Card:"><br />
            <input type="password" maxlength="3" id="cvv" placeholder="Enter your 3-digit CVV:"
onblur="right();"><br />
            <p id="result"></p>
            <button type="button" id="paybtn" onclick="goto();">Deposit</button>
        </form>
    </div>
</body>

</html>
```

Fixdep.css

```
@import url("homeCSS.css");
form{
    /* display: inline-block; */
```

```
    margin:20px auto;
    width:800px;
    border: 5px solid black;
    border-radius: 30px;
    padding:20px;
    text-align: center;
}
input{
    width:500px;
}
label{
    font-weight: bolder;
    font-size: 30px;
    border-top:3px dashed black;
    border-bottom: 3px dashed black;
    padding:5px;
}
select{
    margin-top:20px;
    width:200px;
    height:50px;
    font-size:25px;
    border-radius:30px;
}
button{
    height:50px;
    width:120px;
    font-size: 25px;
    padding:5px;
    border-radius:30px;
```

```
    border-style: outset;
}
p{
    color:red;
    text-decoration: underline;
}
```

Fixdep.js

```
function load () {
    wrong();
}
function wrong () {
    document.getElementById( "paybtn" ).disabled = true;
    document.getElementById( "paybtn" ).style.opacity = "0.5";
}
function right () {
    document.getElementById( "paybtn" ).disabled = false;
    document.getElementById( "result" ).innerHTML = "";
    document.getElementById( "paybtn" ).style.opacity = "1";
}
function validAcc () {
    var accno = document.getElementById( "accno" ).value;
    if ( /[0-9]/.test( accno ) == true && ( accno.length != 16 ) )
    {
        document.getElementById( "result" ).innerHTML = "Account No. should be of 16 numeric characters.";
        wrong();
        document.getElementById( "accno" ).focus();
        return false;
    }
    else
    {
```

```
        document.getElementById( "result" ).innerHTML = "";
        // right();
        return true;
    }
}
function validCard () {
    var card = document.getElementById( "cardno" ).value;
    if ( /[0-9]/.test( card ) == true && ( card.length != 16 ) )
    {
        document.getElementById( "result" ).innerHTML = "Card No. should be of 16 numeric characters.";
        wrong();
        document.getElementById( "cardno" ).focus();
        return false;
    }
    else
    {
        document.getElementById( "result" ).innerHTML = "";
        // right();
        return true;
    }
}

function goto () {
    var rupee = document.getElementById( "rupee" ).value.length;
    var name = document.getElementById( "name" ).value.length;
    if ( validAcc() == true && validCard() == true && rupee == 0 && name == 0 )
    {
        document.getElementById( "result" ).innerHTML = "Some input fields have undesired values.";

        return false;
```

```
    }
    else
    {
        alert( "Deposited..." );
        window.location.replace( "home.html" );
        return true;
    }
}
```

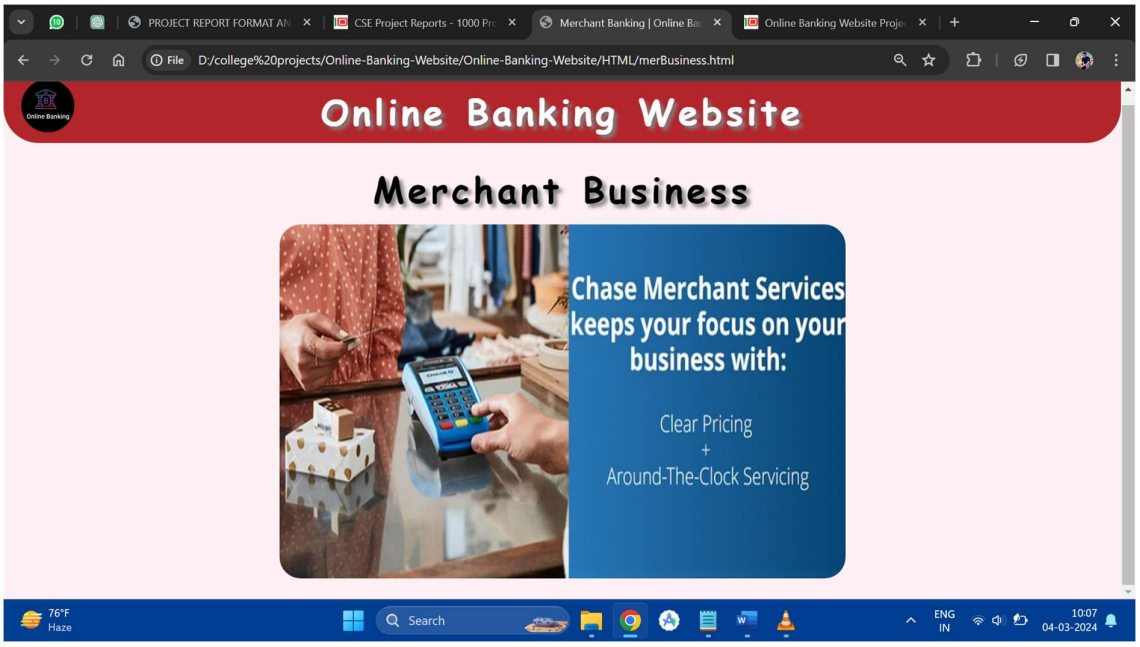
IntTransfer.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>International Transfer | Online Banking Website</title>
    <link href="../CSS/intTransferCSS.css" rel="stylesheet">
</head>
<body>
    <div id="header">
        <h1>Online Banking Website</h1>
        <a href="home.html"><img src="..//Images//logo.png" alt="Online Banking Website" title="Online
Banking Website"></a>
    </div>
    <div>
        <h1>International Money Transfer</h1>
        <div id="int">

        </div>
    </div>
</body>
```

```
</html>
```

IntTransfer.html

```
@import url("../CSS/homeCSS.css");

#int{
    background-image: url("../Images/intMoney.jpg");
    height:600px;
    width:800px;
    background-size:800px 600px;
    margin: 30px auto;
}
```



MerBusiness.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Merchant Banking | Online Banking Website</title>
    <link href="../CSS/merBusinessCSS.css" rel="stylesheet">
```

```
</head>
<body>
    <div id="header">
        <h1>Online Banking Website</h1>
        <a href="home.html"><img src="..//Images//logo.png" alt="Online Banking Website" title="Online
Banking Website"></a>
    </div>
    <div>
        <h1>Merchant Business</h1>
        <div id="mer">


        </div>
    </div>
</body>
</html>
```

MerBusiness.css

```
@import url("../CSS/homeCSS.css");
#mer{
    background-image: url("../Images/merchant.jpg");
    height:500px;
    width:800px;
    border-radius:30px;
    background-size:800px 500px;
    margin:30px auto;
}
```

NetTransfer.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
```

```
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>National Transfer | Online Banking Website</title>
    <link rel="stylesheet" href="../CSS/natTransferCSS.css">
</head>
<body>
    <div id="header">
        <h1>Online Banking Website</h1>
        <a href="home.html"><img src="..//Images//logo.png" alt="Online Banking Website" title="Online Banking Website"></a>
    </div>
    <div>
        <h1>National Money Transfer</h1>
        <div id="nat"></div>
    </div>
</body>
</html>
```

NetTransfer.css

```
@import url("../CSS/homeCSS.css");
#nat{
    background-image: url("../Images/natMoney.jpg");
    height:500px;
    width:700px;
    background-size:700px 500px;
    border-radius:30px;
    margin:30px auto;
}
```

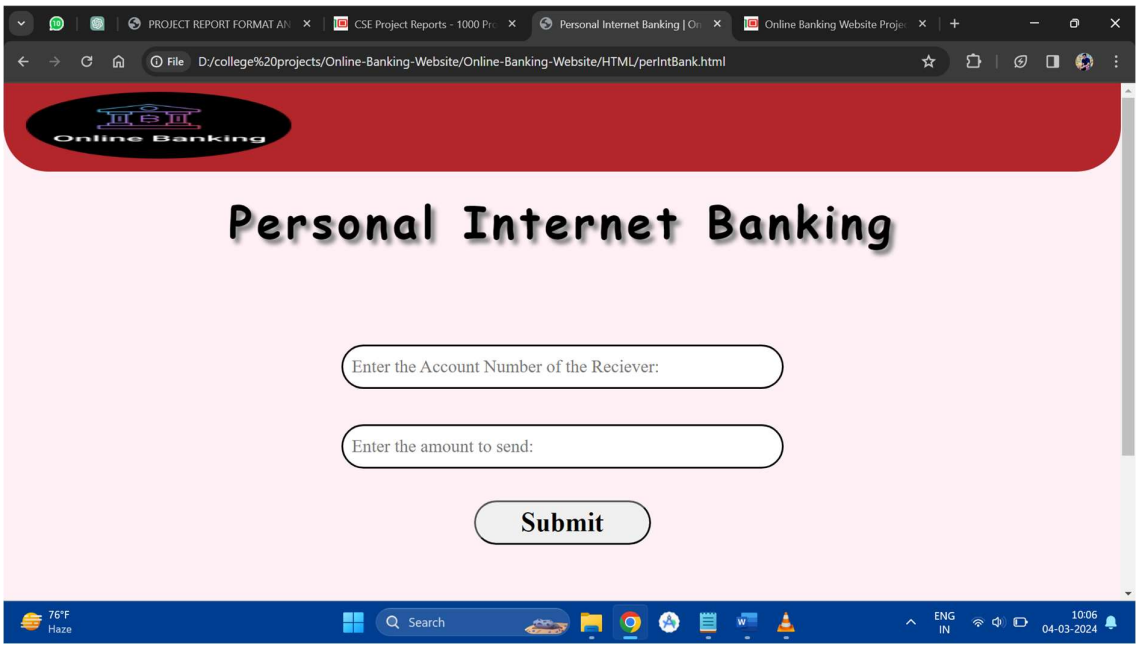
OnlTax.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Online Tax | Online Banking Website</title>
    <link rel="stylesheet" href="../CSS/onlTaxCSS.css">
</head>
<body>
    <div id="header">
        <h1>Online Banking Website</h1>
        <a href="home.html"><img src="../Images/logo.png" alt="Online Banking Website" title="Online Banking Website"></a>
    </div>
    <div>
        <h1>Online Tax</h1>
        <div id="on">
```

```
        </div>
    </div>
</body>
</html>
```

OnlTax.css

```
@import url("../CSS/homeCSS.css");
#on{
    height:500px;
    width:700px;
    background-image: url(../Images/tax.jpg);
    border-radius:30px;
    background-size: 700px 500px;
    margin:30px auto;
}
```



PerIntBank.html

```
<!DOCTYPE html>
<html lang="en">

    <head>
```

```
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Personal Internet Banking | Online Banking Website</title>
    <link href="../CSS/perIntBankCSS.css" rel="stylesheet">
    <script src="../Java Script/perIntBankJS.js"></script>
  </head>

  <body onload="load();">
    <div id="header">
      <h1>Online Banking Website</h1>
      <a href="home.html"><img src="..//Images//logo.png" alt="Online Banking Website"
          title="Online Banking Website"></a>
    </div>
    <div id="menu">
      <h1>Personal Internet Banking</h1>
      <form method="POST">
        <input type="text" name="accno" id="accno" size="16"
          placeholder="Enter the Account Number of the Reciever:"><br />
        <input type="number" min="0" max="20000" name="amnt" id="amnt"
          placeholder="Enter the amount to send:"><br />
        <p id="error"></p>
        <button type="button" id="submitbtn" onclick="goto();">Submit</button>
      </form>
    </div>
  </body>

</html>
```

PerIntBank.css

```
@import url("../CSS/homeCSS.css");
#menu{
```

```
    margin:100px auto;
    height:500px;
    width:1000px;
    text-align: center;
}
div#menu h1{
    position: relative;
    bottom: 100px;
}
input{
    height:50px;
    margin:20px;
    border-radius:30px;
    width:500px;
    font-size:20px;
    padding-left:10px;
}
button{
    border-radius:30px;
    height:50px;
    width:200px;
    font-size:30px;
    font-weight:bolder;
    border-style:outset;
}
input[type="number"]:invalid{
    border:2px solid red;
}
input:focus{
    border:2px solid blue;
```

```
}
#error{
    text-decoration: underline;
    color:red;}
```

Chapter 9

Future Enhancements & Limitations

Future Enhancements:

- Integration of additional payment gateways to expand payment options.
- Implementation of biometric authentication to bolster security measures.
- Development of a mobile banking application for convenient access on smartphones and tablets.
- Introduction of personalized financial management tools based on user spending patterns.
- Integration of AI-powered chatbots for improved customer support and assistance.
- Investigation into blockchain technology for secure and transparent transactions.
- Enhancement of data analytics capabilities to offer personalized banking recommendations.
- Introduction of voice-based commands for hands-free banking experiences.

Limitations:

- Dependency on internet connectivity for accessing banking services may limit accessibility in remote areas with poor connectivity.
- Security risks associated with online transactions, including phishing attacks, malware, and data breaches.
- Compatibility issues with older web browsers or operating systems may affect user experience.
- Potential delays or disruptions in service due to system maintenance or technical issues.
- Regulatory compliance requirements may restrict certain features or services in specific regions.
- Limited availability of customer support outside of standard business hours may inconvenience users requiring immediate assistance.
- Possibility of errors or discrepancies in account information due to system glitches or processing issues.
- Risks associated with unauthorized access or account takeover despite security measures in place.

Chapter 10

Conclusions & References

Conclusions:

- The online banking system provides users with convenient and secure access to a wide range of financial services from anywhere, at any time.
- Despite certain limitations and challenges, the system offers significant advantages in terms of accessibility, efficiency, and flexibility.
- Continuous monitoring, updates, and enhancements are essential to address evolving user needs, technological advancements, and regulatory requirements.
- Overall, the online banking system represents a valuable tool for modern banking customers, offering a seamless and user-friendly banking experience.

References

- https://www.youtube.com/playlist?list=PLu0W_9lII9agiCUZYRsvtGTXdxkzPyItg

Source Code & All Reports Availability

https://github.com/Nisha22204/OnlineCollegeProject