

**VISVESVARAYA TECHNOLOGICAL UNIVERSITY
BELAGAVI-590018**



**Technical Seminar
Report On**

“ DIGITAL BANKING ”

*submitted in partial fulfilment for the award of the degree
of*
Master of Computer Applications

Submitted by

RAKSHITH A [1DT24MC071]

SHREYAS GOWDA HV [1DT24MC084]

Under the Guidance of

Guide Name

**Mr. Bharath Kumara J ,
Assistant Professor , Dept of MCA**



**Department of Master of Computer Applications
DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND
MANAGEMENT**

Accredited by NBA, New Delhi, Accredited by NAAC with Grade A+
Udayapura, Kanakapura Road, Bengaluru-560082

2024-2025

**DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND
MANAGEMENT**

Accredited by NBA, New Delhi, Accredited by NAAC with Grade A+
Udayapura, Kanakapura Road, Bengaluru-560082

Department of Master of Computer Applications



CERTIFICATE

This is to certify that the Technical seminar entitled “**DIGITAL BANKING**” has been successfully carried by **RAKSHITH A** and **SHREYAS GOWDA H V**, bearing the usn **1DT24MC071** and **1DT24MC084** bonafide student of Dayananda Sagar Academy of Technology and Management in partial fulfillment for the award of Master of Computer Applications from Visvesvaraya Technological University, Belagavi, during the year 2024-2025. It is certified that all corrections / suggestions indicated for internal assessment have been incorporated in the report.

.....
Guide sign

.....
HOD-MCA

**DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND
MANAGEMENT**

Accredited by NBA, New Delhi, Accredited by NAAC with Grade A+
Udayapura, Kanakapura Road, Bengaluru-560082

Department of Master of Computer Applications

.....
MCA



**DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND
MANAGEMENT**

Accredited by NBA, New Delhi, Accredited by NAAC with Grade A+
Udayapura, Kanakapura Road, Bengaluru-560082

Department of Master of Computer Applications



DECLARATION

I, **RAKSHITH A**, bearing the USN: **1DT24MC071**, pursuing Master of Computer Applications, Dayananda Sagar Academy of Technology and Management, Bangalore. I hereby declare that the Technical Seminar titled, “**DIGITAL BANKING**” submitted as a partial fulfillment for the award of Master of Computer Applications from Visvesvaraya Technological University, Belagavi, during the academic year 2024-25.

**DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND
MANAGEMENT**

Accredited by NBA, New Delhi, Accredited by NAAC with Grade A+
Udayapura, Kanakapura Road, Bengaluru-560082

Department of Master of Computer Applications



DECLARATION

I, **SHREYAS GOWDA H V**, bearing the USN: **1DT24MC084**, pursuing Master of Computer Applications, Dayananda Sagar Academy of Technology and Management, Bangalore. I hereby declare that the Technical Seminar titled, “**DIGITAL BANKING**” submitted as a partial fulfilment for the award of Master of Computer Applications from Visvesvaraya Technological University, Belagavi, during the academic year 2024-25.

ACKNOWLEDGEMENT

It gives me immense pleasure in bringing out this report of the Technical Seminar entitled. **“DIGITAL BANKING”** Dreams never turn to reality unless a lot of effort and hard work is put into it and no effort bears fruit in the absence of support and guidance. It takes a lot of effort to work my way through this goal and having someone to guide me and help me is always a blessing.

We would like to take this opportunity to thank a few who were closely involved in completing and executing this Report. At the outset, we thank God almighty for making my endeavor success. We would like to express my sincere thanks to the Management of Dayananda Sagar Academy of Technology and Management for providing excellent infrastructure and other facilities, which enabled us to sharpen our skills.

We would like to express a deep sense of gratitude to **Dr. Ravishankar M, Principal of DSATM, Bengaluru**, for providing us with adequate facilities, ways and means by which we were able to complete this project.

We express my sincere gratitude to **Dr. Manjula Sanjay Koti, Professor and Head of Department of Master of Computer Applications** for encouraging us to aim higher and constant support and valuable suggestions without which the successful completion of this project would not have been possible.

We would like to express our gratitude to our project guide

in Department of Master of Computer Applications for constant motivation, support, and guidance.

We express our immense indebtedness to all the teachers and staff of Dept. of Master of Computer Application, **Dayananda Sagar Academy of Technology and Management** for their cooperation and support. Finally, we thank all others, and especially our classmates and our family members who in one way or another helped us in the successful completion of this work.

CONTENTS

S.No	Contents	Page No
1	Introduction	1
2	Objectives	2
3	Scope of the project	3
4	Existing Vs Proposed System	4
5	Software And Hardware Requirements	5
6	Software Design	6-9
7	Implementation	10-11
8	Testing	12-13
9	Code Design	14-25
10	Conclusion	26
11	Bibliography	27

INTRODUCTION

Digital banking refers to the digitization of all traditional banking activities and programs that were historically only available to customers when physically inside a bank branch. Through digital platforms—websites, mobile applications, and automated services—customers can access accounts, transfer funds, apply for loans, pay bills, and invest with minimal physical interaction.

The rise of smartphones, secure internet connections, and fintech innovations has significantly accelerated the adoption of digital banking. This system not only offers convenience but also reduces operational costs for banks, provides real-time services, and enhances customer satisfaction.

Gives an overview of digital banking, explaining what it is, its evolution from traditional banking, and its significance in today's financial ecosystem. May touch on features like mobile banking, internet banking, and online transactions

OBJECTIVES

The main objectives of this project are:

- To design and implement a secure and user-friendly digital banking system.
- To provide customers with 24/7 access to banking services via online and mobile platforms.
- To ensure secure transactions through encryption and multi-factor authentication.
- To integrate value-added services such as bill payment, investment tracking, and loan management.
- To reduce the reliance on physical bank branches and manual paperwork.
- Enable quick transactions and payments.
- Improve customer experience.

Scope of the Project

Covers account management, fund transfers, bill payments, loan applications, and customer support. Excludes high-frequency trading and complex corporate banking systems. This project focuses on the development and deployment of a digital banking platform that includes core banking functionalities:

- Account management (balance inquiry, statements).
- Fund transfers (within bank and interbank).
- Loan application and tracking.
- Bill payments and recharge services.
- Target individual retail customers and small business account holders.
- Allow easy scalability to include additional services like investment tracking or insurance in future.
- Customer support via chatbots and messaging.

The scope excludes in-depth core banking infrastructure replacement and high-frequency trading systems. It is designed for retail banking customers and small business account holders.

Existing Vs Proposed System

Existing System (Traditional Banking)

- Banking services are mostly branch-based and require physical visits.
- Transactions processed manually, leading to delays.
- Limited working hours.
- Paper-based record keeping.
- Higher operational costs due to manpower and infrastructure.
- Security based mainly on physical measures (branch security, passbooks).

Proposed System (Digital Banking)

- Accessible 24/7 via internet and mobile applications.
- Instant processing of transactions.
- Fully automated and paperless operations.
- Reduced operational costs through digitization.
- Advanced security measures such as encryption, OTP, and biometric authentication.
- Cloud-based storage for easy retrieval of record.

Software and Hardware Requirements

Software Requirements:

- Operating System: Windows / Linux / macOS
- Server: Apache / Nginx
- Database: MySQL / PostgreSQL
- Programming Languages: Java / Python / PHP
- Frameworks: Spring Boot / Django / Laravel
- Frontend: HTML5, CSS3, JavaScript, React or Angular
- Security: SSL Certificates, OAuth 2.0, JWT

Hardware Requirements:

- Processor: Intel i5 or higher
- RAM: 8 GB minimum
- Storage: 500 GB HDD or SSD
- Network: High-speed internet connection
- Backup Storage: External/Cloud backup solution

Software Design

The software design defines the architecture, components, interfaces, and data flow of the digital banking system. It ensures that the system is scalable, secure, and easy to maintain.

1. Architecture

The system follows a Three-Tier Architecture:

- Presentation Layer (Client Side)
 - Web and mobile user interfaces for customers.
 - Built using HTML5, CSS3, JavaScript, and frameworks like React or Angular.
 - Responsible for displaying account details, processing user input, and initiating transactions.
- Application Layer (Server Side)
 - Contains business logic for authentication, transaction processing, and account management.
 - Developed using Java (Spring Boot) or Python (Django/Flask).
 - Implements RESTful APIs for communication between client and server.
- Database Layer (Data Storage)
 - Stores user information, account details, and transaction history.
 - Uses relational databases like MySQL or PostgreSQL.
 - Ensures data security through encryption and role-based access control.

2. Design Principles

- Modularity – Dividing the system into independent modules like login, fund transfer, and loan management.
- Scalability – Allows easy addition of new features without major redesign.
- Security – Implements encryption (SSL/TLS), secure coding, and authentication mechanisms.
- Maintainability – Uses clean code structure (MVC pattern) for easier updates and bug fixes.

3. Key Components

- User Authentication Module – Manages login, OTP verification, and password reset.
- Account Management Module – Displays balance, transaction history, and statements.
- Transaction Module – Handles fund transfers, bill payments, and scheduled payments.

DIGITAL BANKING

- Support Module – Provides chatbot or live chat functionality.
 - Admin Panel – Allows bank staff to manage accounts, monitor transactions, and generate reports.
-

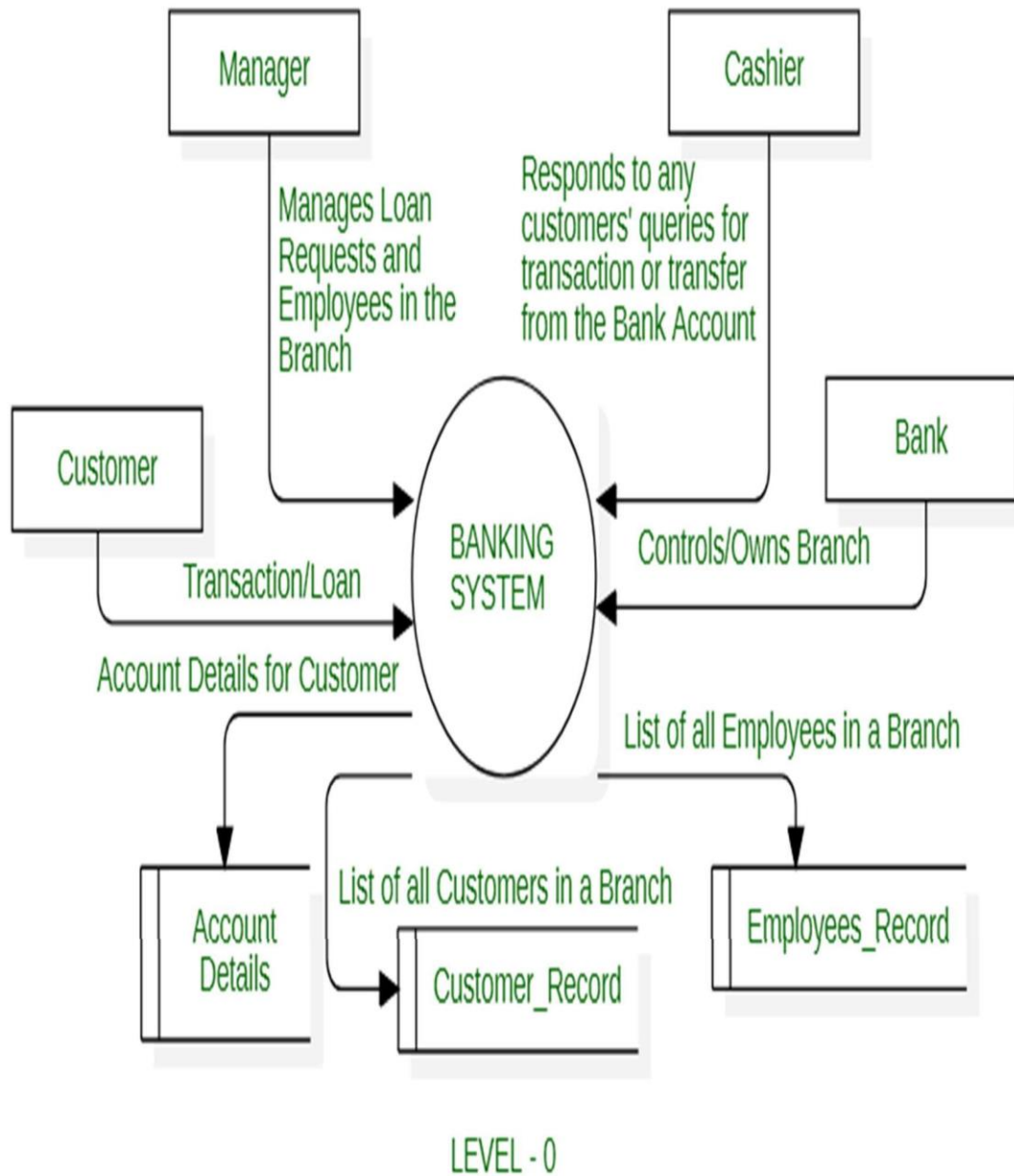
4. Data Flow

- The user logs in via web/mobile application.
 - Request is sent to the server for authentication.
 - Upon success, the server retrieves data from the database.
 - The system processes user actions like fund transfer and updates records.
 - Response is sent back to the client for display.
-

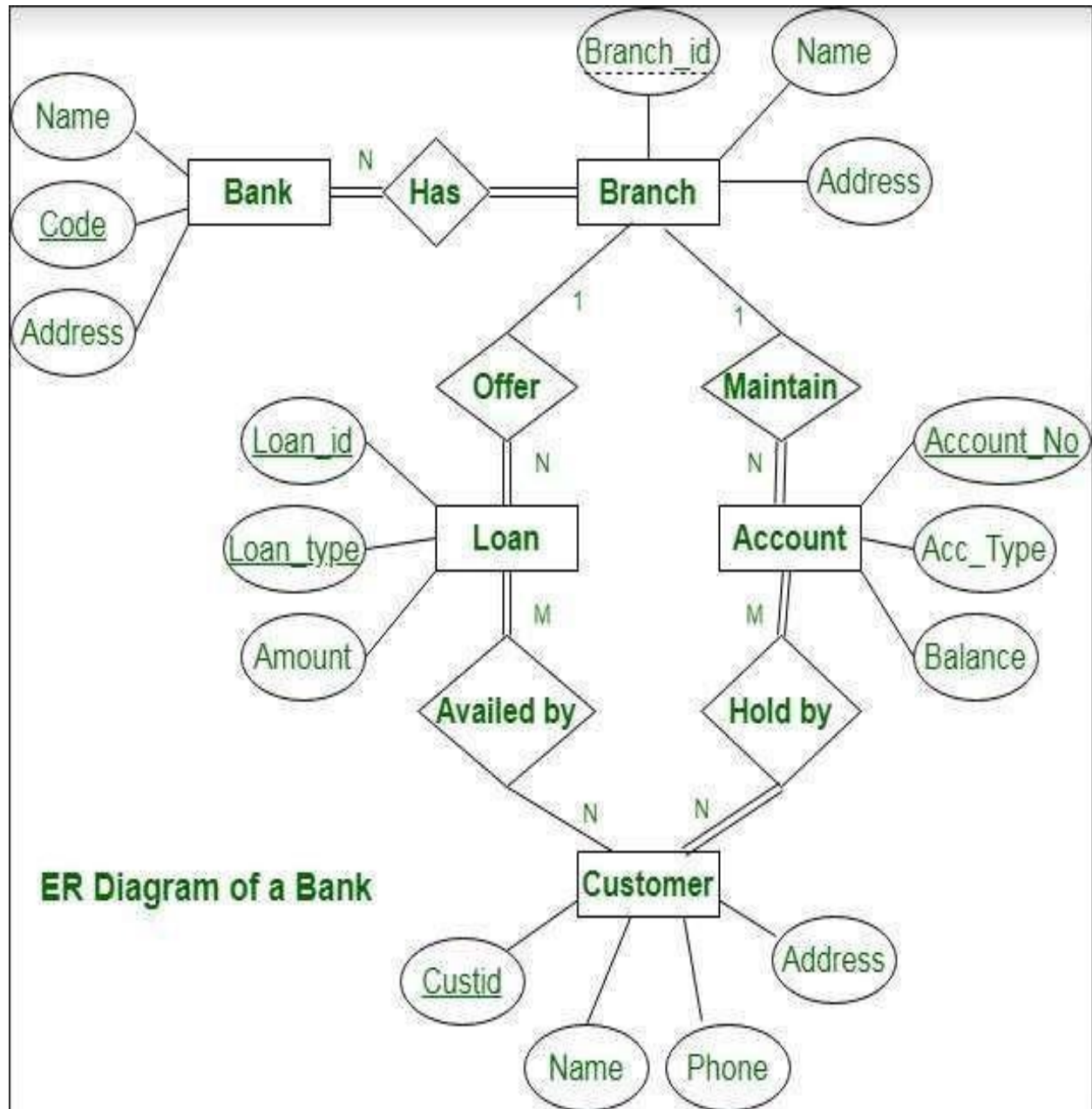
5. Security Design

- All communication is encrypted using HTTPS.
- Sensitive data like passwords is hashed before storage.
- Role-based permissions ensure only authorized access to certain features.
- Regular security audits and vulnerability testing are planned.

Data Flow Diagram



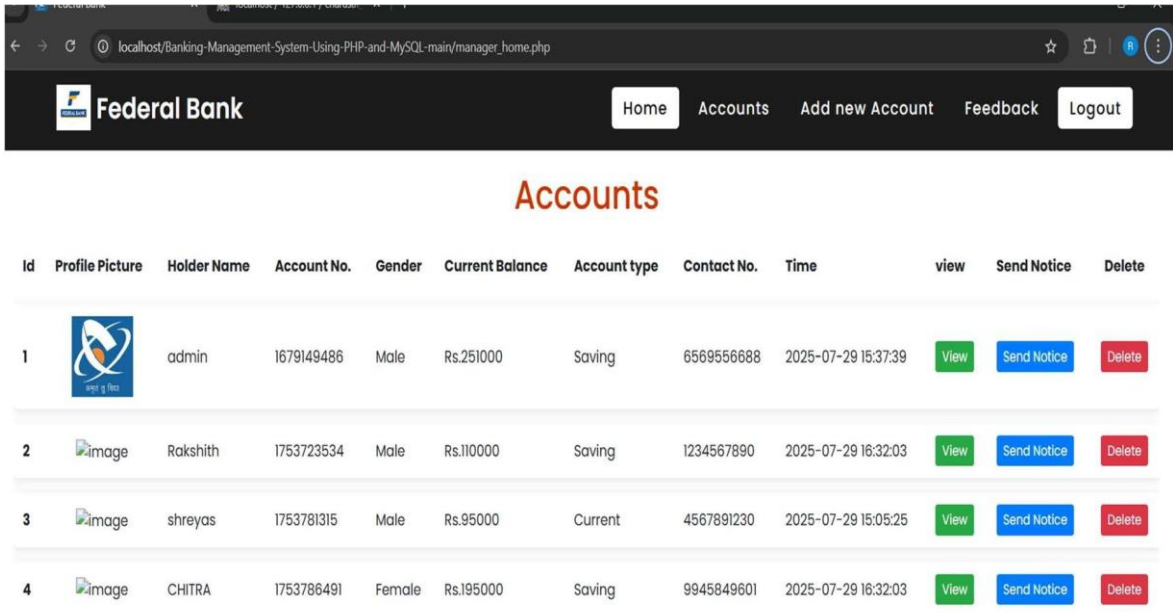
E-R Diagram







Implementation


The implementation is divided into phases:

- Requirement Analysis – Collect functional and non-functional requirements.
- Prototype Development – Build a basic version for user feedback.
- Module Development – Code different modules such as account management, fund transfer, and bill payment.
- Integration – Combine all modules into a single system.
- Deployment – Host the system on a secure server and configure mobile applications.



Id	Profile Picture	Holder Name	Account No.	Gender	Current Balance	Account type	Contact No.	Time	view	Send Notice	Delete
1		admin	1679149486	Male	Rs.251000	Saving	6569556688	2025-07-29 15:37:39	View	Send Notice	Delete
2		Rakshith	1753723534	Male	Rs.110000	Saving	1234567890	2025-07-29 16:32:03	View	Send Notice	Delete
3		shreyas	1753781315	Male	Rs.95000	Current	4567891230	2025-07-29 15:05:25	View	Send Notice	Delete
4		CHITRA	1753786491	Female	Rs.195000	Saving	9945849601	2025-07-29 16:32:03	View	Send Notice	Delete

localhost/Banking-Management-System-Using-PHP-and-MySQL-main/addnewaccount.php

 **Federal Bank**

Home Accounts **Add new Account** Feedback Logout

NEW ACCOUNT FORM

Name	Aadhaar card
<input type="text" value="Enter Your Name"/>	<input type="text" value="Enter Your Aadhaar card"/>
Gender	Email address
<input type="text" value="Select Gender"/>	<input type="text" value="email address"/>
Mobile Number	Password
<input type="text" value="Mobile Number"/>	<input type="text" value="Enter Your Password"/>
City	Address
<input type="text" value="Enter You City"/>	<input type="text" value="Enter Your Address"/>
Profile Picture	Date Of Birthdate
<input type="text" value="Choose file..."/> <input type="button" value="Browse"/>	<input type="text" value="dd-mm-yyyy"/>
Account Number	Account Type
<input type="text" value="1754907108"/>	<input type="text" value="Saving"/>

Testing

Testing Phase

The testing phase is critical to ensure that the digital banking system is **error-free, secure, and user-friendly** before it goes live. This phase validates the system against the functional and non-functional requirements gathered during the planning stage.

1. Objectives of Testing

- Detect and fix software bugs.
 - Verify that the system meets the specified requirements.
 - Ensure data security and system stability under various conditions.
 - Confirm that the user experience is smooth and intuitive.
-

2. Types of Testing

- **Unit Testing**
 - Tests individual components or modules (e.g., login, fund transfer, statement generation).
 - Ensures that each function works independently as expected.
- **Integration Testing**
 - Checks the communication between modules, such as how the transaction module updates the account management module.
- **System Testing**
 - Tests the complete system as a whole to ensure all features work together seamlessly.
- **Security Testing**
 - Validates the protection of sensitive data.
 - Includes testing against SQL injection, cross-site scripting (XSS), brute force attacks, and unauthorized access.
- **Performance Testing**
 - Measures system speed, responsiveness, and stability under normal and high loads.

➤ User Acceptance Testing (UAT)

- Conducted by real users to ensure the system is easy to use and meets their needs.
-

3. Tools and Methods Used

- **Automated Testing Tools:** Selenium, JUnit, Postman (for API testing).
 - **Security Tools:** OWASP ZAP, Burp Suite.
 - **Manual Testing:** Performed for usability and exploratory testing.
-

4. Outcomes

- All critical bugs fixed before release.
- Verified high-level security compliance for sensitive banking data.
- Positive UAT feedback confirming readiness for deployment.

Code Design

The codebase follows:

- MVC Structure for maintainability.
- Reusable Components for UI elements and backend functions.
- Secure Coding Practices including input validation, parameterized queries, and encryption.
- API Documentation using tools like Swagger.
- Version Control via Git for collaboration and rollback.

Source Code

```
-- phpMyAdmin SQL Dump
-- version 5.2.1
-- https://www.phpmyadmin.net/
--
-- Host: 127.0.0.1
-- Generation Time: Mar 18, 2023 at 04:07 PM
-- Server version: 10.4.24-MariaDB
-- PHP Version: 8.1.6

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
```

DIGITAL BANKING

```
-- Database: `charusat_bank`

--

-----

--

-- Table structure for table `feedback`

--

CREATE TABLE `feedback` (
  `feedbackid` int(11) NOT NULL,
  `message` varchar(111) NOT NULL,
  `userid` int(11) NOT NULL,
  `date` timestamp NOT NULL DEFAULT current_timestamp() ON UPDATE
current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

--

-- Dumping data for table `feedback`

--

INSERT INTO `feedback` (`feedbackid`, `message`, `userid`, `date`) VALUES
(10, 'hii \r\n', 57, '2023-01-02 05:31:43'),
(11, 'hii \r\n', 57, '2023-01-02 05:31:58'),
(12, 'hiiii', 58, '2023-02-06 21:50:46');

-----

--

-- Table structure for table `login`
```

--

```
CREATE TABLE `login` (  
  `id` int(11) NOT NULL,  
  `email` varchar(100) NOT NULL,  
  `password` varchar(100) NOT NULL,  
  `type` varchar(100) NOT NULL DEFAULT 'cashier',  
  `date` timestamp NOT NULL DEFAULT current_timestamp() ON UPDATE  
  current_timestamp()  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

--

-- Dumping data for table `login`

--

```
INSERT INTO `login` (`id`, `email`, `password`, `type`, `date`) VALUES  
(3, 'cashier@gmail.com', '123', 'cashier', '2022-11-10 05:15:51');
```

--

-- Table structure for table `manager`

--

```
CREATE TABLE `manager` (  
  `id` int(10) NOT NULL,  
  `email` varchar(100) NOT NULL,  
  `password` int(100) NOT NULL,  
  `type` varchar(100) NOT NULL DEFAULT 'manager',
```

DIGITAL BANKING

```
`date` timestamp NOT NULL DEFAULT current_timestamp() ON UPDATE  
current_timestamp()
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
--
```

```
-- Dumping data for table `manager`
```

```
--
```

```
INSERT INTO `manager` (`id`, `email`, `password`, `type`, `date`) VALUES  
(1, 'manager@manager.com', 1234, 'manager', '2022-10-07 08:13:42');
```

```
-----
```

```
--
```

```
-- Table structure for table `notice`
```

```
--
```

```
CREATE TABLE `notice` (  
  `id` int(10) NOT NULL,  
  `userid` int(10) NOT NULL,  
  `notice` text CHARACTER SET latin1 NOT NULL,  
  `time` timestamp NOT NULL DEFAULT current_timestamp() ON UPDATE  
current_timestamp()  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
-----
```

```
--
```

```
-- Table structure for table `otheraccounts`
```

```
--
```



```
CREATE TABLE `otheraccounts` (  
  `id` int(11) NOT NULL,  
  `accountno` varchar(11) NOT NULL,  
  `bankname` varchar(11) NOT NULL,  
  `holdername` varchar(11) NOT NULL,  
  `balance` varchar(11) NOT NULL,  
  `date` date NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
--  
-- Table structure for table `transaction`  
--
```

```
CREATE TABLE `transaction` (  
  `transactionId` int(11) NOT NULL,  
  `action` varchar(50) NOT NULL,  
  `credit` varchar(50) CHARACTER SET latin1 NOT NULL,  
  `debit` varchar(50) NOT NULL,  
  `balance` varchar(50) NOT NULL,  
  `beneld` varchar(50) NOT NULL,  
  `other` varchar(50) NOT NULL,  
  `userid` int(11) NOT NULL,  
  `date` timestamp NOT NULL DEFAULT current_timestamp() ON UPDATE  
  current_timestamp()  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;  
  
--
```

-- Dumping data for table `transaction`

--

```
INSERT INTO `transaction` (`transactionId`, `action`, `credit`, `debit`, `balance`, `beneld`,  
`other`, `userid`, `date`) VALUES
```

```
(1, 'transfer', '', '2000', '', '', '1667924704', 52, '2022-11-08 16:26:17'),  
(2, 'transfer', '', '5500', '', '', '1667924704', 52, '2022-11-08 16:50:56'),  
(3, 'transfer', '', '5000', '', '', '1667924704', 52, '2022-11-08 16:58:58'),  
(4, 'transfer', '', '500', '', '', '1667924704', 52, '2022-11-08 16:59:35'),  
(5, 'transfer', '', '400', '', '', '1667924704', 52, '2022-11-08 17:00:06'),  
(6, 'transfer', '', '3000', '', '', '1667924704', 52, '2022-11-08 17:09:59'),  
(7, 'transfer', '', '3600', '', '', '1667924704', 52, '2022-11-08 17:10:32'),  
(8, 'transfer', '', '3000', '', '', '1667924704', 53, '2022-11-08 17:33:36'),  
(9, 'transfer', '', '3000', '', '', '1667489276', 53, '2022-11-08 17:35:14'),  
(10, 'transfer', '', '3000', '', '', '1667966401', 54, '2022-11-09 04:02:01'),  
(11, 'transfer', '', '3000', '', '', '1667966401', 55, '2022-11-09 04:05:01'),  
(12, 'transfer', '', '5000', '', '', '1667966401', 55, '2022-11-09 04:06:00'),  
(13, 'transfer', '', '5000', '', '', '1667966167', 55, '2022-11-09 04:11:17'),  
(14, 'transfer', '', '3000', '', '', '1667966401', 54, '2022-11-09 04:17:45'),  
(15, 'transfer', '', '4000', '', '', '1667966167', 55, '2022-11-09 04:19:09'),  
(16, 'transfer', '', '5000', '', '', '1667966167', 54, '2022-11-09 04:19:44'),  
(17, 'transfer', '', '3000', '', '', '1667966167', 54, '2022-11-09 04:20:09'),  
(18, 'transfer', '', '6000', '', '', '1667966401', 54, '2022-11-09 04:36:25'),  
(19, 'transfer', '', '3000', '', '', '1667966167', 55, '2022-11-09 04:50:11'),  
(20, 'transfer', '', '3000', '', '', '1667971493', 55, '2022-11-09 05:27:38'),  
(21, 'transfer', '', '5000', '', '', '1667966401', 56, '2022-11-09 14:24:07'),  
(22, 'withdraw', '', '3000', '', '', '1', 0, '2022-11-09 15:50:04'),  
(23, 'withdraw', '', '3000', '', '', '1', 0, '2022-11-09 15:50:44'),  
(24, 'withdraw', '', '3000', '', '', '1', 0, '2022-11-09 15:51:25'),  
(25, 'withdraw', '', '3000', '', '', '1', 0, '2022-11-09 15:51:36'),
```

(26, 'withdraw', ", '-4031', ", ", '1', 0, '2022-11-09 15:52:48'),
(27, 'withdraw', ", '-4031', ", ", '1', 0, '2022-11-09 15:53:14'),
(28, 'withdraw', ", '-4031', ", ", '1', 0, '2022-11-09 15:53:16'),
(29, 'withdraw', ", '500', ", ", '1', 55, '2022-11-09 15:53:25'),
(30, 'withdraw', ", '593', ", ", '15', 55, '2022-11-09 15:53:55'),
(31, 'withdraw', ", '2000', ", ", '1', 55, '2022-11-09 15:54:48'),
(32, 'deposit', '5000', ", ", ", '2', 55, '2022-11-09 15:56:33'),
(33, 'withdraw', ", '2500', ", ", '1', 55, '2022-11-09 15:58:29'),
(34, 'withdraw', ", '25', ", ", '1', 55, '2022-11-09 15:58:49'),
(35, 'withdraw', ", '7475', ", ", ", 55, '2022-11-09 16:00:46'),
(36, 'withdraw', ", '7475', ", ", ", 55, '2022-11-09 16:05:21'),
(37, 'withdraw', ", '7475', ", ", ", 55, '2022-11-09 16:06:35'),
(38, 'withdraw', ", '7475', ", ", ", 55, '2022-11-09 16:06:49'),
(39, 'deposit', '1', ", ", ", '1', 55, '2022-11-09 16:08:50'),
(40, 'withdraw', ", '3000', ", ", '1', 55, '2022-11-09 16:11:14'),
(41, 'deposit', '5000', ", ", ", '1', 55, '2022-11-09 16:29:33'),
(42, 'withdraw', ", '5000', ", ", '1', 55, '2022-11-09 16:34:15'),
(43, 'deposit', '5000', ", ", ", '1', 55, '2022-11-09 16:43:09'),
(44, 'deposit', '5000', ", ", ", '1', 55, '2022-11-09 16:43:39'),
(45, 'withdraw', ", '3000', ", ", '1', 55, '2022-11-09 16:43:52'),
(46, 'withdraw', ", '3000', ", ", '1', 55, '2022-11-09 16:44:04'),
(47, 'withdraw', ", '3000', ", ", '1', 55, '2022-11-09 16:46:49'),
(48, 'transfer', ", '5000', ", ", '1668057431', 57, '2023-01-02 05:34:29'),
(49, 'transfer', ", '5000', ", ", '1668057431', 57, '2023-02-06 21:49:02'),
(50, 'transfer', ", '4000', ", ", '1668057431', 57, '2023-02-06 21:49:51'),
(51, 'withdraw', ", '4000', ", ", '2', 59, '2023-03-18 14:41:04');

--

DIGITAL BANKING

-- Table structure for table `useraccounts`

--

```
CREATE TABLE `useraccounts` (  
  `id` int(11) NOT NULL,  
  `name` varchar(50) NOT NULL,  
  `aadhaar` varchar(15) NOT NULL,  
  `gender` varchar(8) NOT NULL,  
  `email` varchar(30) NOT NULL,  
  `phonenummer` varchar(20) NOT NULL,  
  `city` varchar(20) NOT NULL,  
  `address` varchar(50) NOT NULL,  
  `password` varchar(20) NOT NULL,  
  `profile` varchar(50) NOT NULL,  
  `dob` date DEFAULT NULL,  
  `accountno` int(30) NOT NULL,  
  `accounttype` varchar(10) NOT NULL,  
  `deposit` int(10) NOT NULL,  
  `branch` varchar(10) NOT NULL,  
  `occupation` varchar(20) NOT NULL,  
  `time` timestamp NOT NULL DEFAULT current_timestamp() ON UPDATE  
  current_timestamp()  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

--

-- Dumping data for table `useraccounts`

--

```
INSERT INTO `useraccounts` (`id`, `name`, `aadhaar`, `gender`, `email`, `phonenummer`, `city`,  
  `address`, `password`, `profile`, `dob`, `accountno`, `accounttype`, `deposit`, `branch`,  
  `occupation`, `time`) VALUES
```

DIGITAL BANKING

```
(59, 'admin', '122225656525', 'Male', 'admin123@gmail.com', '6569556688', 'mumbai', '33, hihi society ', '1234', 'charusat_symbol.jpg', '2023-03-08', 1679149486, 'Saving', 1000, '', 'Bussiness', '2023-03-18 14:41:04');
```

```
--
```

```
-- Indexes for dumped tables
```

```
--
```

```
--
```

```
-- Indexes for table `feedback`
```

```
--
```

```
ALTER TABLE `feedback`
```

```
ADD PRIMARY KEY (`feedbackid`);
```

```
--
```

```
-- Indexes for table `login`
```

```
--
```

```
ALTER TABLE `login`
```

```
ADD PRIMARY KEY (`id`);
```

```
--
```

```
-- Indexes for table `manager`
```

```
--
```

```
ALTER TABLE `manager`
```

```
ADD PRIMARY KEY (`id`);
```

```
--
```

```
-- Indexes for table `notice`
```

```
--
```

```
ALTER TABLE `notice`
```

DIGITAL BANKING

```
ADD PRIMARY KEY (`id`);

--

-- Indexes for table `otheraccounts`

--

ALTER TABLE `otheraccounts`

ADD PRIMARY KEY (`id`);

--

-- Indexes for table `transaction`

--

ALTER TABLE `transaction`

ADD PRIMARY KEY (`transactionId`);

--

-- Indexes for table `useraccounts`

--

ALTER TABLE `useraccounts`

ADD PRIMARY KEY (`id`),

ADD UNIQUE KEY `email` (`email`);

--

-- AUTO_INCREMENT for dumped tables

--

--

-- AUTO_INCREMENT for table `feedback`

--

ALTER TABLE `feedback`

MODIFY `feedbackid` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=13;
```

```
--  
  
-- AUTO_INCREMENT for table `login`  
  
--  
  
ALTER TABLE `login`  
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=4;  
  
--  
  
-- AUTO_INCREMENT for table `manager`  
  
--  
  
ALTER TABLE `manager`  
  MODIFY `id` int(10) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;  
  
--  
  
-- AUTO_INCREMENT for table `notice`  
  
--  
  
ALTER TABLE `notice`  
  MODIFY `id` int(10) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=19;  
  
--  
  
-- AUTO_INCREMENT for table `otheraccounts`  
  
--  
  
ALTER TABLE `otheraccounts`  
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT;  
  
--  
  
-- AUTO_INCREMENT for table `transaction`  
  
--  
  
ALTER TABLE `transaction`  
  MODIFY `transactionId` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=52;
```

```
--  
-- AUTO_INCREMENT for table `useraccounts`  
--  
ALTER TABLE `useraccounts`  
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=60;  
COMMIT;  
  
/!*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;  
/!*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;  
/!*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```


Conclusion

The proposed digital banking system modernizes traditional banking by offering secure, efficient, and user-friendly services. It reduces operational costs, enhances customer satisfaction, and provides a platform for continuous innovation. As banking moves toward a more digital future, such solutions are essential for maintaining competitiveness and meeting customer expectations.

The implementation of a **Digital Banking System** marks a significant shift from traditional, branch-based banking to a modern, technology-driven approach. This project successfully demonstrates how core banking services can be made available to customers **anytime and anywhere** through secure web and mobile platforms. By integrating features such as account management, fund transfers, bill payments, and loan applications, the system enhances **convenience** while reducing operational costs for banks. The use of encryption, authentication, and secure coding practices ensures that customer data and transactions are **protected against potential threats**.

The proposed system also supports **scalability**, enabling banks to add future services like investment tracking, insurance, and AI-driven financial advice without overhauling the core architecture.

Overall, the project demonstrates that adopting digital banking not only improves **customer satisfaction** but also boosts **banking efficiency**. As financial technology continues to evolve, such systems will become essential for institutions aiming to remain competitive and meet growing customer expectations.

Bibliography

- Tapscott, Don. *Digital Economy: Promise and Peril in the Age of Networked Intelligence*. McGraw-Hill, 2016.
- Laudon, Kenneth C., and Jane P. Laudon. *Management Information Systems: Managing the Digital Firm*. Pearson, 2022.
- Chishti, Susanne, and Janos Barberis. *The FINTECH Book*. Wiley, 2016.
- Reserve Bank of India (RBI) – Reports on Digital Banking and Payment Systems.
- National Payments Corporation of India (NPCI) – UPI and IMPS Guidelines.