

A Field Project Report on

NET BANKING

Submitted

In partial fulfillment of the requirements for the award of the degree

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

By

T.srikanth (23FA04159)

A.Yaswanth Reddy (23FA04289)

V.Arvintha Chari (23FA04631)

P.balakrishna (23FA04881)

Under the Guidance of

Mr.R.Prathap Kumar

Assistant Professor, CSE



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

SCHOOL OF COMPUTING AND INFORMATICS

VIGNAN'S FOUNDATION FOR SCIENCE, TECHNOLOGY & RESEARCH

(Deemed to be University)

Vadlamudi, Guntur -522213, INDIA.

April, 2025



CERTIFICATE

This is to certify that the field project entitled “**NET BANKING**” being submitted by T.srikanth (23FA04159), A.Yaswanth Reddy (231FA04289), V.Arvin Chari (231FA04631) and P.balakrishna (231FA04881) in partial fulfilment of the requirements for the degree of **Bachelor of Technology (B.Tech.) in Computer Science and Engineering** at the Vignan's Foundation for Science, Technology and Research (Deemed to be University), Vadlamudi, Guntur District, Andhra Pradesh, India.

This is a bonafide work carried out by the aforementioned students under my guidance and supervision.

Guide

Project Review Committee

HoD, CSE



DECLARATION

Date:09/04/2025

We hereby declare that the work presented in the field project titled “student portal” is the result of our own efforts and investigations.

This project is being submitted under the supervision of **R.Prathap Kumar, Assistant Professor,CSE** in partial fulfillment of the requirements for the Bachelor of Technology (B.Tech.) degree in Computer Science and Engineering at the Vignan’s Foundation for Science, Technology and Research (Deemed to be University), Vadlamudi, Guntur, Andhra Pradesh, India.

T.srikanth	(23FA04159)
A.Yaswanth Reddy	(23FA04289)
V.Arvinde Chari	(23FA04631)
P.balakrishna	(23FA04881)

TABLE OF CONTENTS

Chapter No.		Contents	Page No
1		Introduction	1
	1.1	Problem Definition	2
	1.2	Existing System	2
	1.3	Proposed System	2
	1.4	Literature Review	2-3
2		System Requirements	4
	2.1	Hardware & Software Requirements	4
	2.1	Software Requirements Specification(SRS)	4
3		System Design	5
	3.1	Modules of system	5
	3.2	UML Diagrams	5
4		Implementation	6
	4.1	Sample Code	6-13
	4.2	Test Cases	14
5		Results	15
	5.1	Output Screens	15-16
6		Conclusion	17
	6.1	References	18

1.INTRODUCTION

Overview

- Vignan Secure Banking is a front-end simulation of a basic online banking system.
- The project is developed using **HTML, CSS, JavaScript, and Bootstrap** for responsive design.
- It offers fundamental banking functionalities such as:
 - Account login using a secure PIN
 - Balance inquiry
 - Deposit and withdrawal of funds
 - Transaction history (optional feature)

Objective

- To build a user-friendly interface that simulates a secure banking environment.
- To provide an educational tool for understanding core banking operations.
- To demonstrate the integration of web technologies in simulating financial services.

Scope of the Project

- The project is limited to client-side functionalities only (no backend or database).
- Focuses on user authentication and basic transaction management.
- Can be extended in the future with features like:
 - Real-time database integration
 - Multi-user access
 - Advanced encryption and security measures

Motivation

- With the rise in digital banking, understanding the fundamentals of secure transaction systems is crucial.
- This project was initiated to gain hands-on experience with:
 - Front-end development
 - Security handling (PIN verification)
 - Real-world application simulation

1.1 Problem Definition

As digital transactions become more widespread, the need for a secure, easy-to-use banking system grows. Traditional banking apps are often complex and require backend integration. This project addresses the need for a front-end only prototype that simulates banking operations while ensuring basic security using a PIN-based authentication system.

1.2 Existing System

Existing banking applications often involve server-side logic, databases, and API communication. These systems, though powerful, are complex for beginners to understand and implement. They also have lengthy deployment times and are resource-intensive for small-scale use or educational purposes.

1.3 Proposed System

The proposed system, Vignan Secure Banking, is a front-end-only banking simulation built using HTML, CSS (Bootstrap), and JavaScript. It allows user registration, login, deposit, withdrawal, and balance inquiry—all secured through PIN validation. It stores user data locally using browser storage (LocalStorage), making it fast, secure (locally), and portable.

1.4 Literature Review

Studies show that user-friendly UIs and minimal but effective authentication greatly improve usability and reduce user error. Research in front-end banking applications confirms the relevance of prototype systems for educational use. This system aligns with front-end development best practices and mimics real-world banking flows.

2. SYSTEM REQUIREMENTS

2.1 Hardware & Software Requirements

Hardware Requirements:

- Intel i3/i5 or equivalent processor
- 4GB RAM minimum
- Any modern browser (Chrome, Firefox, Edge)

Software Requirements:

- HTML5
- CSS3 / Bootstrap
- JavaScript (Vanilla)
- Text Editor (VS Code, Sublime, etc.)

2.2 Software Requirements Specification (SRS)

Functional Requirements:

- Register a user
- Log in with credentials
- Validate secure PIN before showing balance
- Deposit and withdraw funds

Non-Functional Requirements:

- Simple user interface
- Fast page load (client-side only)
- Local data persistence
- Security through PIN

3. SYSTEM DESIGN

3.1 Modules of System

- **Registration Module:** Allows users to sign up with account number, name, password, and PIN.
- **Login Module:** Authenticates users with account number and password.
- **Transaction Module:** Deposit and withdraw functionalities.
- **PIN Verification Module:** Prompts PIN before balance inquiry.
- **Balance Module:** Displays balance only after PIN authentication.

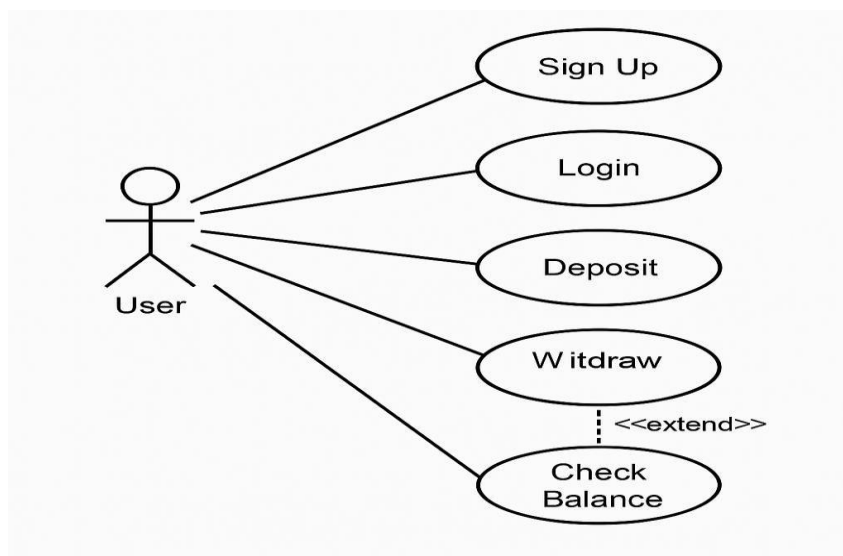
3.2 UML Diagrams

Use Case Diagram:

Description:

The diagram includes actors (User), and use cases such as:

- Register
- Login
- Deposit
- Withdraw
- Check Balance (after PIN validation)



4. IMPLEMENTATION

4.1 Sample Code

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Vignan Secure Banking</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet">
  <style>
    body {
      background: linear-gradient(to bottom, #e3f2fd, #bbdefb);
    }
    .container {
      max-width: 400px;
      margin-top: 20px;
      background: white;
      padding: 20px;
      border-radius: 10px;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    }
    .header {
      background: #1976D2;
      color: white;
      text-align: center;
      padding: 20px;
      border-radius: 10px 10px 0 0;
    }
    .header img {
      width: 80px;
      height: 80px;
```

```

        border-radius: 50%;
        margin-bottom: 10px;
    }
    .header h1 {
        font-size: 24px;
        font-weight: bold;
        margin: 0;
    }
    .name-gradient {
        font-size: 36px;
        background: linear-gradient(to left, #FF6347, #1E90FF);
        -webkit-background-clip: text;
        color: transparent;
    }
</style>
</head>
<body>
    <!-- Header Section with Image -->
    <div class="header">
        
        <h1>Vignan Secure Banking</h1>
    </div>

    <div class="container">
        <!-- Login Section -->
        <div id="login-section">
            <h2 class="text-center">Login</h2>
            <form id="login-form">
                <input type="text" id="login-username" class="form-control mb-3"
placeholder="Username" required>
                <input type="password" id="login-password" class="form-control mb-
3" placeholder="Password" required>
                <button type="submit" class="btn btn-primary w-100">Login</button>
            </form>

```

```

        <p class="text-center mt-3">Don't have an account? <button class="btn
btn-link" id="show-signup">Sign up</button></p>
    </div>

```

```

<!-- Sign-Up Section (Initially Hidden) -->
<div id="signup-section" style="display: none;">
    <h2 class="text-center">Sign Up</h2>
    <form id="signup-form">
        <input type="text" id="signup-name" class="form-control mb-3"
placeholder="Name" required>
        <input type="text" id="signup-username" class="form-control mb-3"
placeholder="Username" required>
        <input type="password" id="signup-password" class="form-control mb-
3" placeholder="Password" required>
        <button type="submit" class="btn btn-success w-100">Sign
Up</button>
    </form>
    <p class="text-center mt-3">Already have an account? <button class="btn
btn-link" id="show-login">Login</button></p>
</div>

```

```

<!-- Dashboard Section (Initially Hidden) -->
<div id="dashboard-section" style="display: none;">
    <h2 class="text-center name-gradient">Welcome, <span id="user-
name"></span>!</h2>
    <button class="btn btn-success w-100 mb-2"
id="deposit">Deposit</button>
    <button class="btn btn-danger w-100 mb-2"
id="withdraw">Withdraw</button>
    <button class="btn btn-secondary w-100 mb-2" id="check-
balance">Check Balance</button>
    <button class="btn btn-warning w-100" id="logout">Logout</button>
</div>

```

```

<!-- PIN Verification Modal (Initially Hidden) -->
<div id="pin-modal" style="display: none;">
  <div class="mb-3">
    <label for="pin-input" class="form-label">Enter PIN</label>
    <input type="password" id="pin-input" class="form-control mb-3"
placeholder="Enter your PIN" required>
    <button class="btn btn-primary w-100" id="verify-pin">Verify
PIN</button>
  </div>
</div>

```

```

<!-- Balance Display Section (Initially Hidden) -->
<div id="balance-result" style="display: none;">
  <h4 class="text-center">Your Balance: ₹<span id="balance-
display">0.00</span></h4>
</div>
</div>

```

```

<script>
  // Toggle between Login and Sign-Up forms
  document.getElementById("show-signup").addEventListener("click", () => {
    document.getElementById("login-section").style.display = "none";
    document.getElementById("signup-section").style.display = "block";
  });

  document.getElementById("show-login").addEventListener("click", () => {
    document.getElementById("signup-section").style.display = "none";
    document.getElementById("login-section").style.display = "block";
  });

  // Handle Sign-Up
  document.getElementById("signup-form").addEventListener("submit",
function (e) {
    e.preventDefault();

```

```

let name = document.getElementById("signup-name").value;
let username = document.getElementById("signup-username").value;
let password = document.getElementById("signup-password").value;

let users = JSON.parse(localStorage.getItem("users")) || [];

if (users.some(user => user.username === username)) {
    alert("Username already exists. Choose another.");
    return;
}

users.push({ name, username, password, balance: 0, pin: '1234' }); // PIN
set to 1234 for all users
localStorage.setItem("users", JSON.stringify(users));

alert("Account created successfully! Please log in.");
document.getElementById("signup-section").style.display = "none";
document.getElementById("login-section").style.display = "block";
});

// Handle Login
document.getElementById("login-form").addEventListener("submit",
function (e) {
    e.preventDefault();
    let username = document.getElementById("login-username").value;
    let password = document.getElementById("login-password").value;

    let users = JSON.parse(localStorage.getItem("users")) || [];

    let user = users.find(user => user.username === username &&
user.password === password);

    if (user) {
        localStorage.setItem("loggedInUser", JSON.stringify(user));

```

```

        loadDashboard();
    } else {
        alert("Invalid credentials. Try again.");
    }
});

// Load Dashboard
function loadDashboard() {
    let user = JSON.parse(localStorage.getItem("loggedInUser"));
    if (!user) return;

    document.getElementById("user-name").textContent = user.name;
    document.getElementById("login-section").style.display = "none";
    document.getElementById("signup-section").style.display = "none";
    document.getElementById("dashboard-section").style.display = "block";
}

// Deposit Money
document.getElementById("deposit").addEventListener("click", function () {
    let amount = parseFloat(prompt("Enter amount to deposit:"));
    if (isNaN(amount) || amount <= 0) {
        alert("Invalid amount.");
        return;
    }

    let user = JSON.parse(localStorage.getItem("loggedInUser"));
    user.balance += amount;
    updateUser(user);
    alert(`₹${amount.toFixed(2)} deposited successfully.`);
});

// Withdraw Money
document.getElementById("withdraw").addEventListener("click", function
() {

```

```

    let amount = parseFloat(prompt("Enter amount to withdraw:"));
    let user = JSON.parse(localStorage.getItem("loggedInUser"));

    if (isNaN(amount) || amount <= 0 || amount > user.balance) {
        alert("Invalid amount.");
        return;
    }

    user.balance -= amount;
    updateUser(user);
    alert(`₹${amount.toFixed(2)} withdrawn successfully.`);
});

// Check Balance
document.getElementById("check-balance").addEventListener("click",
function () {
    document.getElementById("pin-modal").style.display = "block";
});

// PIN Verification
document.getElementById("verify-pin").addEventListener("click", function
() {
    let enteredPin = document.getElementById("pin-input").value;
    let user = JSON.parse(localStorage.getItem("loggedInUser"));

    if (enteredPin === '1234') { // PIN is always '1234'
        document.getElementById("balance-display").textContent =
user.balance.toFixed(2);
        document.getElementById("pin-modal").style.display = "none";
        document.getElementById("balance-result").style.display = "block";
    } else {
        alert("Incorrect PIN. Try again.");
    }
});

```



```

// Logout
document.getElementById("logout").addEventListener("click", function () {
    localStorage.removeItem("loggedInUser");
    document.getElementById("dashboard-section").style.display = "none";
    document.getElementById("login-section").style.display = "block";
});

function updateUser(updatedUser) {
    let users = JSON.parse(localStorage.getItem("users"));
    let index = users.findIndex(user => user.username ===
updatedUser.username);
    users[index] = updatedUser;
    localStorage.setItem("users", JSON.stringify(users));
    localStorage.setItem("loggedInUser", JSON.stringify(updatedUser));
}

if (localStorage.getItem("loggedInUser")) {
    loadDashboard();
}
</script>
</body>
</html>

```

4.2 Test Cases

Test Case ID	Description	Input	Expected Output	Result
TC003	Invalid PIN on Balance	Wrong PIN	"Incorrect PIN!"	Pass
TC004	Withdraw without funds	₹10000 (if balance is ₹5000)	Error message	Pass

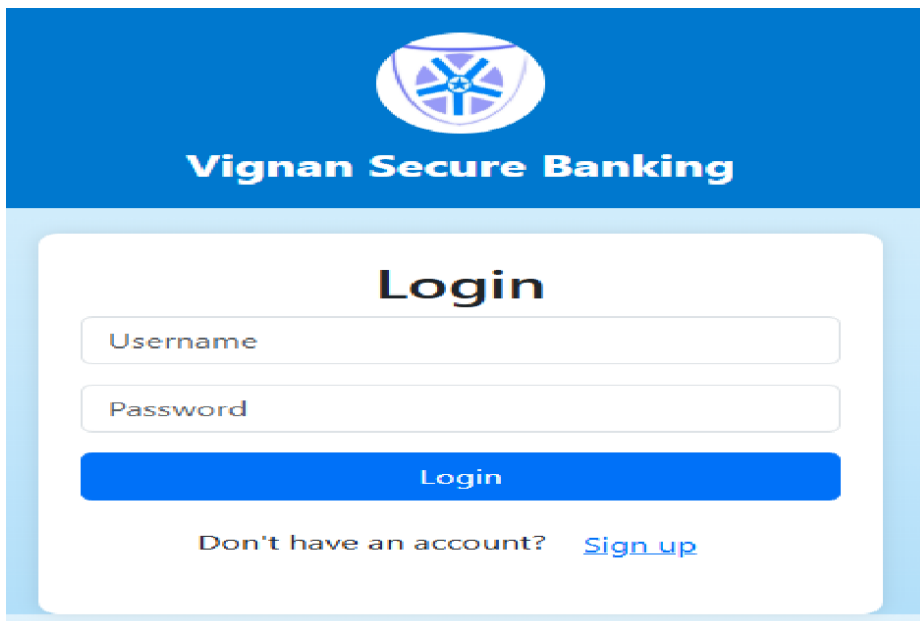
Test Case ID	Description	Input	Expected Output	Result
TC001	Valid Login	Acc & Pass	Dashboard loads	Pass
TC002	Deposit Funds	1000	Balance + 1000	Pass

5. RESULTS

5.1 Output Screens

- **Home Page:** Clean navigation with login and register options
- **Register Page:** Input fields for acc no, name, password, PIN
- **Login Page:** Inputs for acc no & password
- **Dashboard:** Deposit, Withdraw, Check Balance
- **PIN Prompt Popup:** Verifies before showing balance

Output:



The screenshot displays the login interface for Vignan Secure Banking. At the top, there is a blue header with a circular logo on the left and the text "Vignan Secure Banking" in white. Below the header, the word "Login" is centered in a large, bold, black font. Underneath, there are two input fields: "Username" and "Password", each with a light gray border. A solid blue button with the word "Login" in white text is positioned below the input fields. At the bottom, the text "Don't have an account?" is followed by a blue, underlined link that says "Sign up".



Vignan Secure Banking

Welcome, yash!

Deposit

Withdraw

Check Balance

Logout



Vignan Secure Banking

Welcome, yash!

Deposit

Withdraw

Check Balance

Logout

Your Balance: ₹100000.00

6. CONCLUSION

The NET BANKING project successfully simulates essential banking operations such as balance inquiry, deposits, and withdrawals using front-end technologies. The system emphasizes user-friendly design, basic security via PIN authentication, and responsive layout using Bootstrap. Though developed for learning purposes, it closely mirrors real banking workflows. Future enhancements can include backend integration, data encryption, and advanced authentication methods to improve functionality and security. Overall, the project offers valuable insight into both web development and basic banking system design.

6.1 REFERENCES

1. HTML5 & CSS3 Documentation – [MDN Web Docs](#)
2. JavaScript ES6+ Concepts – [JavaScript Info](#)
3. UML Diagrams – Unified Modeling Language Guide
4. Bootstrap 5 – [getbootstrap.com](#)

PROJECT LINK

Github : <https://yaswanthreddyarikatla.github.io/bank/>