ZERO PAY

Submitted By

Student Name	Student ID
Md. Abdul Alim Sarkar	0242310005101872
Ariful Islam	0242310005101610
Sharif Md Husain	0242310005101300
Rezaul Islam	0242310005101334
S M Masum	0242310005101570

LAB PROJECT REPORT
This Report Presented in Partial Fulfillment of the course
CSE312: Database Management System Lab

DAFFODIL INTERNATIONAL UNIVERSITY
Dhaka, Bangladesh
April 12, 2025

DECLARATION

We hereby declare that this lab project has been done by us under the supervision of **Taslima Ferdaus Shuva**, **Assistant Professor**, Department of Computer Science and Engineering, Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere as lab projects.

Submitted To:

Taslima Ferdaus Shuva

Assistant Professor
Department of Computer Science and Engineering
Daffodil International University

Submitted by

<table>
<tr><td colspan="2">Md. Abdul Alim Sarkar
Student Name Student
ID: 0242310005101872
Dept. of CSE, DIU</td></tr>
<tr><td>Ariful Islam
Student Name
ID: 0242310005101610
Dept. of CSE, DIU</td><td>Sharif Md Husain
Student Name
ID: 0242310005101300
Dept. of CSE, DIU</td></tr>
<tr><td>Rezaul Islam
Student Name
ID: 0242310005101334
Dept. of CSE, DIU</td><td>S M Masum
Student Name
ID: 0242310005101570
Dept. of CSE, DIU</td></tr>
</table>

Table of Contents

Declaration **1**

1 Introduction **3**
1.1 Introduction....3
1.2 Motivation....3
1.3 Objectives....3
1.4 Feasibility Study....3
1.5 Gap Analysis....3
1.6 Project Outcome....3

2 Proposed Methodology/Architecture **4–16**
2.1 Requirement Analysis & Design Specification....4
2.1.1 Overview....4
2.1.2 Tool....4
2.1.3 Schema....4
2.1.4 ER Diagram…….....5
2.1.5 UI Design ....(6 –16)
2.2 Overall Project Plan 16

3 Implementation and Results **17**
3.1 Implementation....17
3.2 Performance Analysis....17
3.3 Results and Discussion…….....17

4 Engineering Standards and Mapping **18–21**
4.1 Impact on Society, Environment and Sustainability....18
4.1.1 Impact on Life....18
4.1.2 Impact on Society & Environment…….....18
4.1.3 Ethical Aspects…….....19
4.1.4 Sustainability Plan…….....19
4.2 Project Management and Teamwork....19
4.3 Complex Engineering Problem…….....20
4.3.1 Mapping of Program Outcome…….....20
4.3.2 Complex Problem Solving…….....20
4.3.3 Engineering Activities…….....21

5 Conclusion **22**
5.1 Summary…….....22
5.2 Limitation…….....22
5.3 Future Work…….....22

Chapter 1: Introduction

1.1 Introduction

ZERO PAY is a banking system designed to manage and store critical banking information including employees, branches, customers, and transactions. It allows secure login for admins, management of customers, and real-time updates on account balances.

1.2 Motivation

Banking data management demands accuracy and reliability. Our motivation was to create a robust, secure, and scalable solution that provides full CRUD operations, security features, and optimized data storage.

1.3 Objectives

- To design a relational database for a banking system.
- To implement CRUD operations using SQL.
- To use triggers and stored procedures for automation.
- To build a user interface for admins and users.

1.4 Feasibility Study

Existing banking systems often rely on outdated or overly complex platforms. This system is tailored for mid-sized banking institutions seeking a customizable and efficient solution.

1.5 Gap Analysis

Current banking systems lack adaptability and are often cost-prohibitive. ZERO PAY addresses these gaps with a lightweight, scalable solution.

1.6 Project Outcome

- Efficient schema for customer and transaction management.
- Secure admin access.
- Real-time balance tracking with transaction logging.

Chapter 2: Proposed Methodology/Architecture

2.1 Requirement Analysis & Design Specification

2.1.1 Overview

The system has five core entities: Admin_info, Branch, Employee, Customer_info, and Accounts_balance.

2.1.2 Tools

- **XAMPP**
- **MySQL**
- **IntelliJ IDEA**

2.1.3 Schema

Admin_Info(Username(PK), Password)

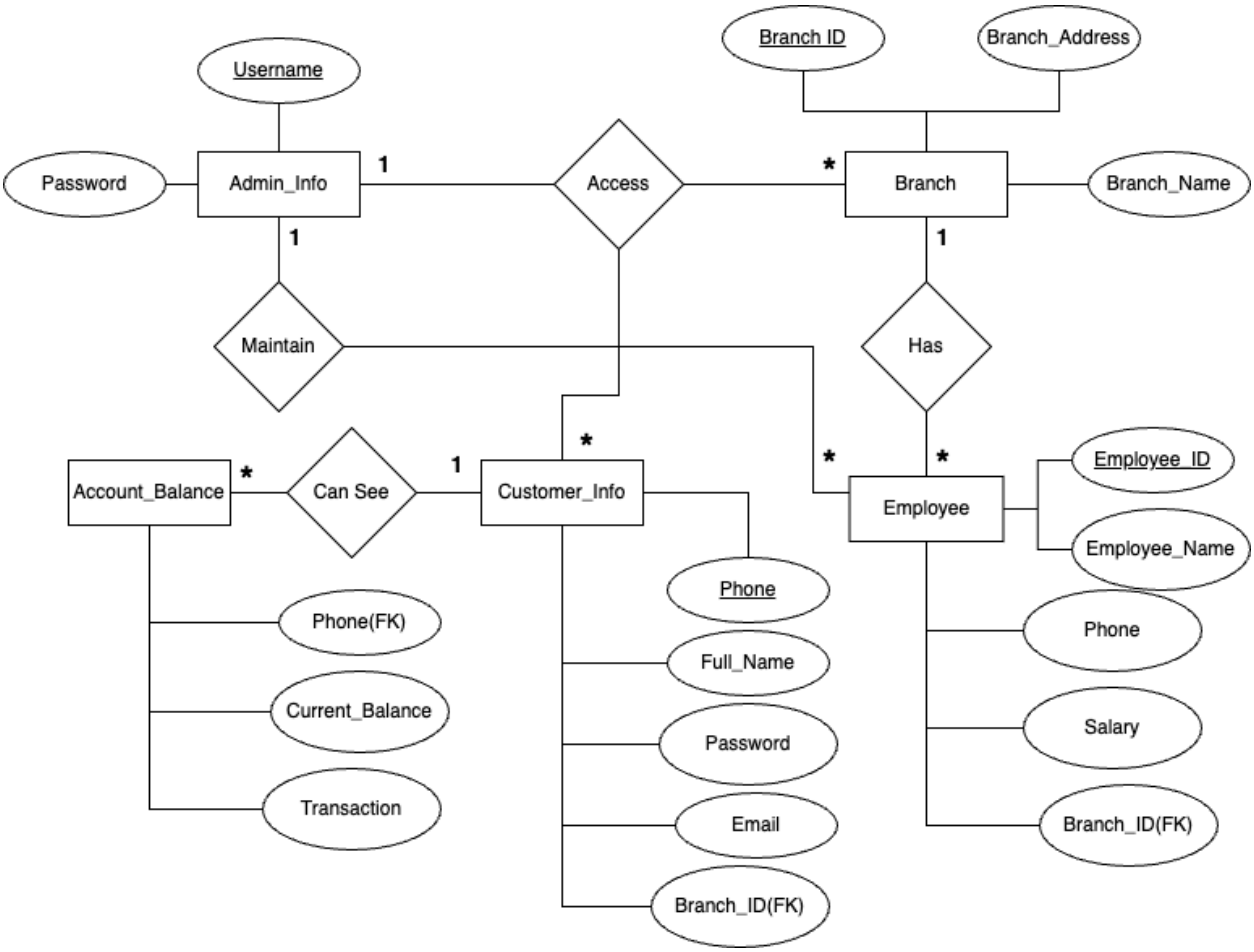
Branch(Branch_ID(PK), Branch_Name, Branch_Address)

Employee(Employee_ID(PK), Employee_Name, Phone, Salary, Branch_ID(FK))

Customer_Info(Phone(PK), Full_Name, Email, Password, Branch_ID(FK))

Accounts_Balance(Phone(FK), Customer_Balance, Transaction)

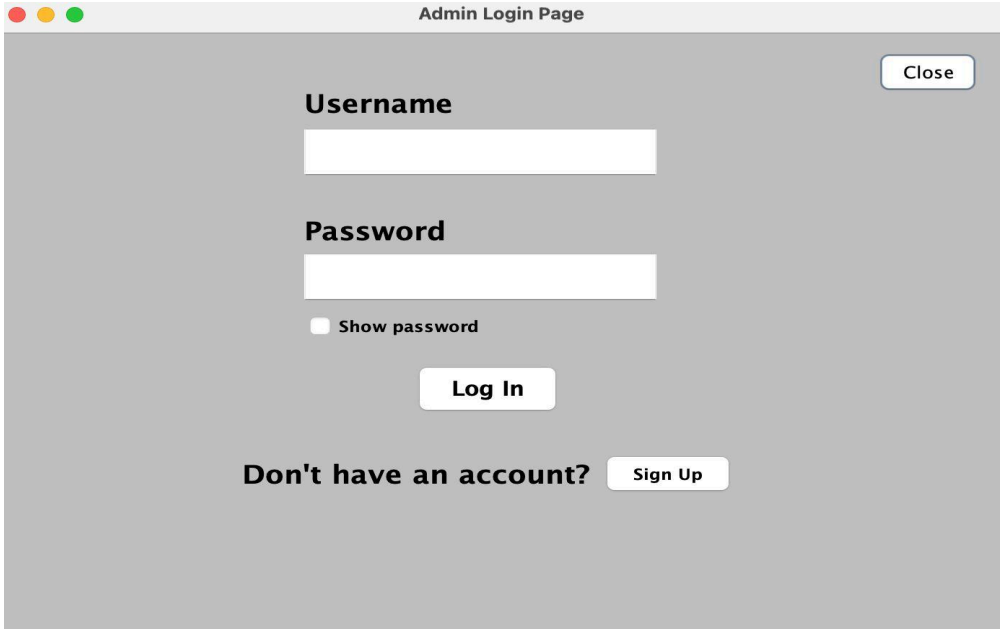
2.1.4 ER Diagram



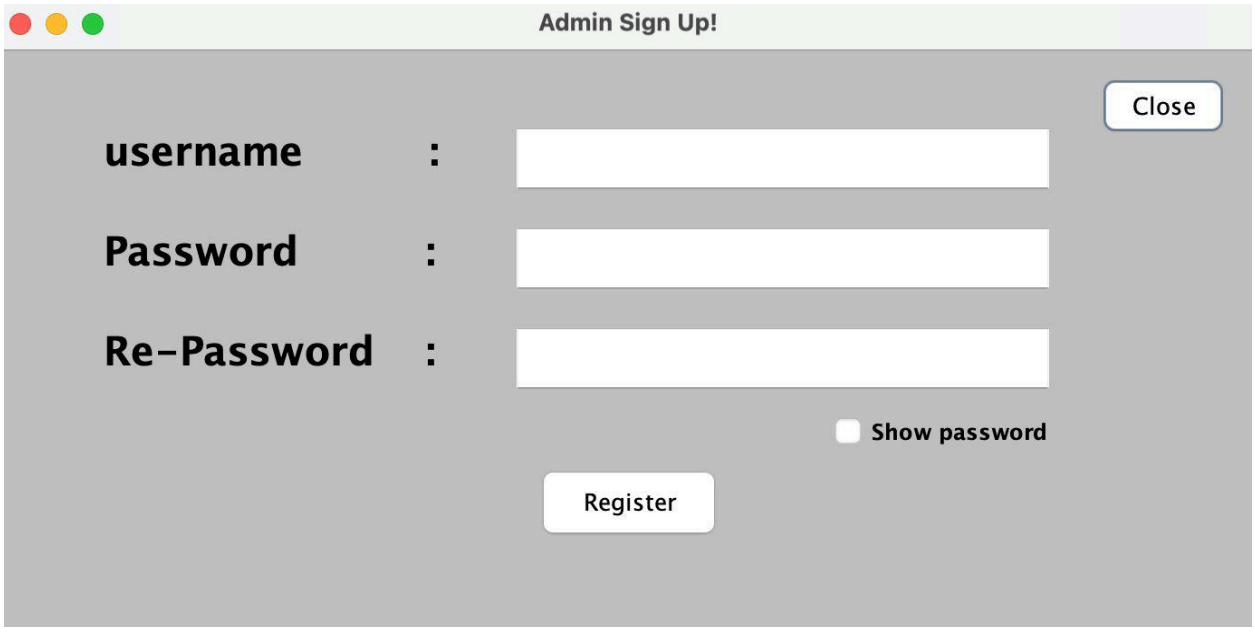
2.1.5 UI Design

- Login admin/user

Admin login page



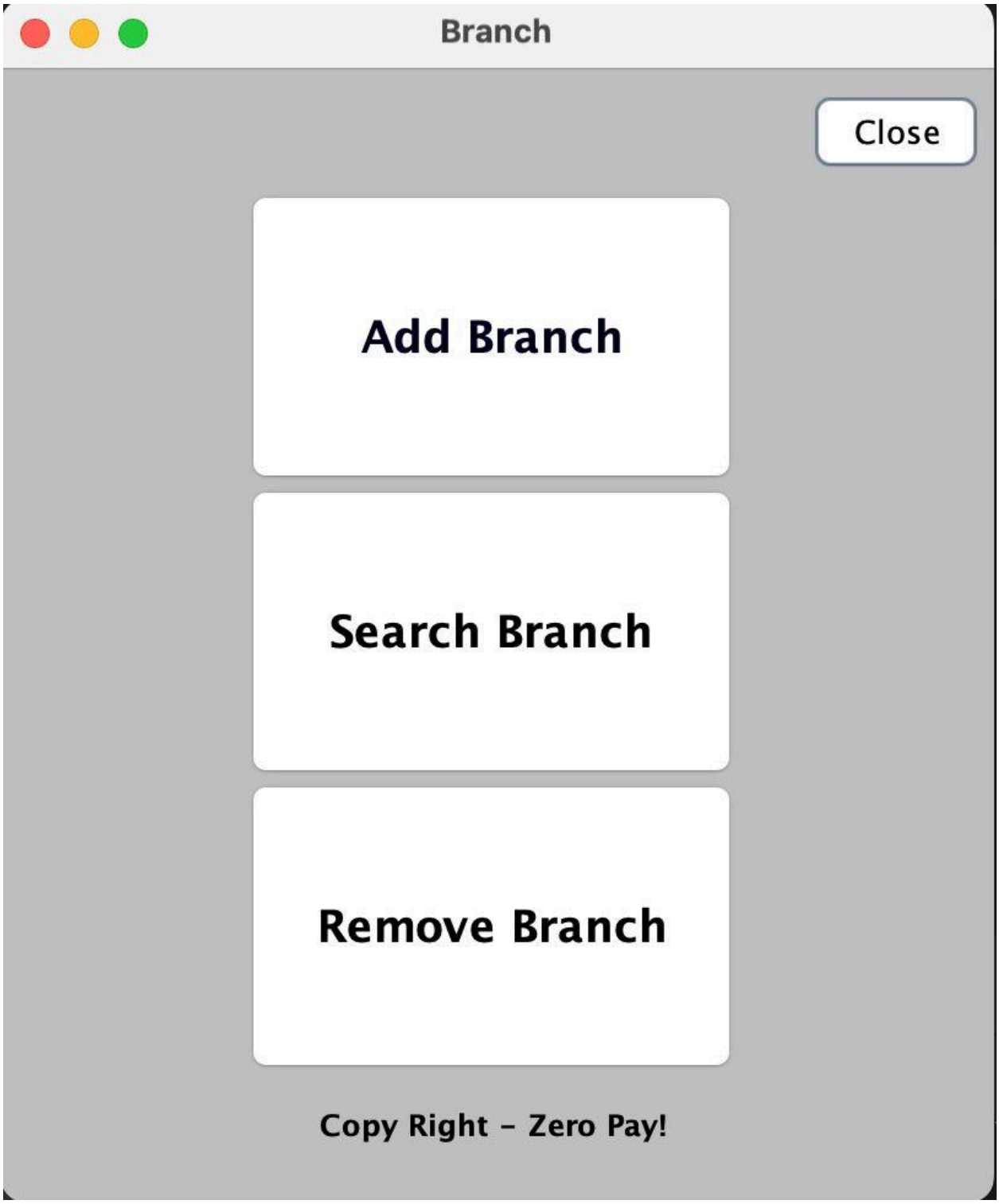
Admin sign up



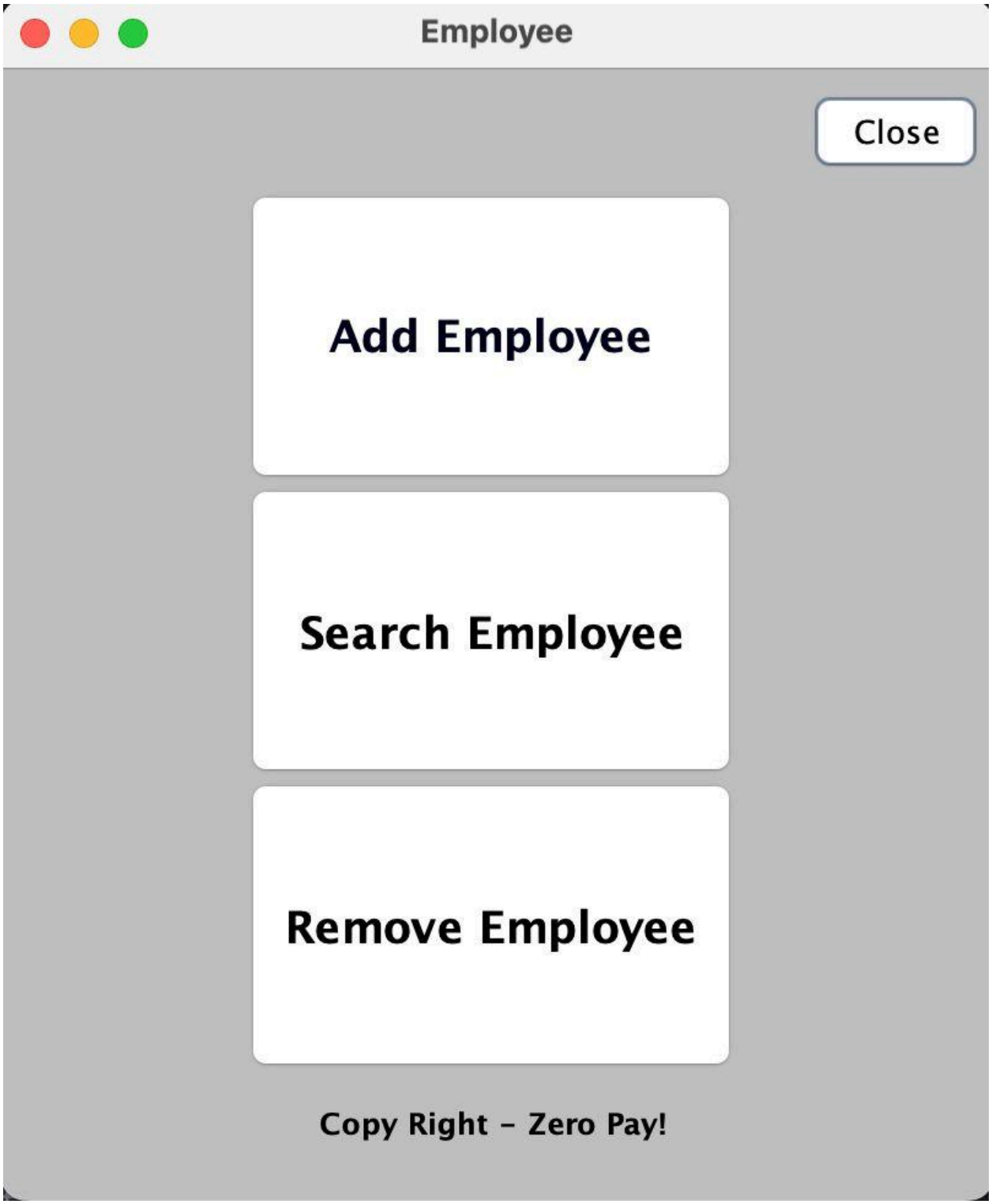
Admin panel


Admin Panel!
Welcome, alim
Log Out
Branch
Employee
Customer
Change Password
Copy Right – Zero Pay!

Admin Branch access



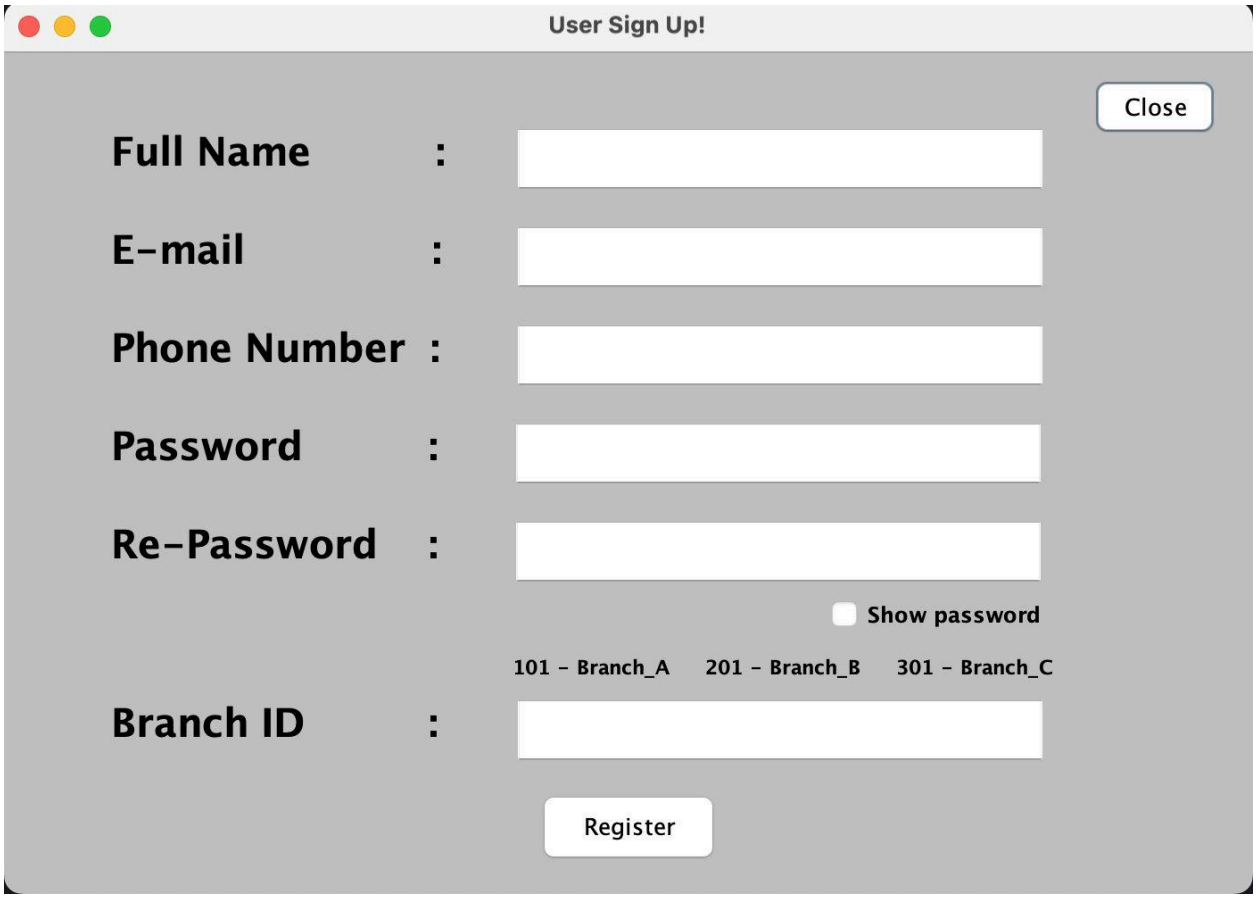
Admin employee access



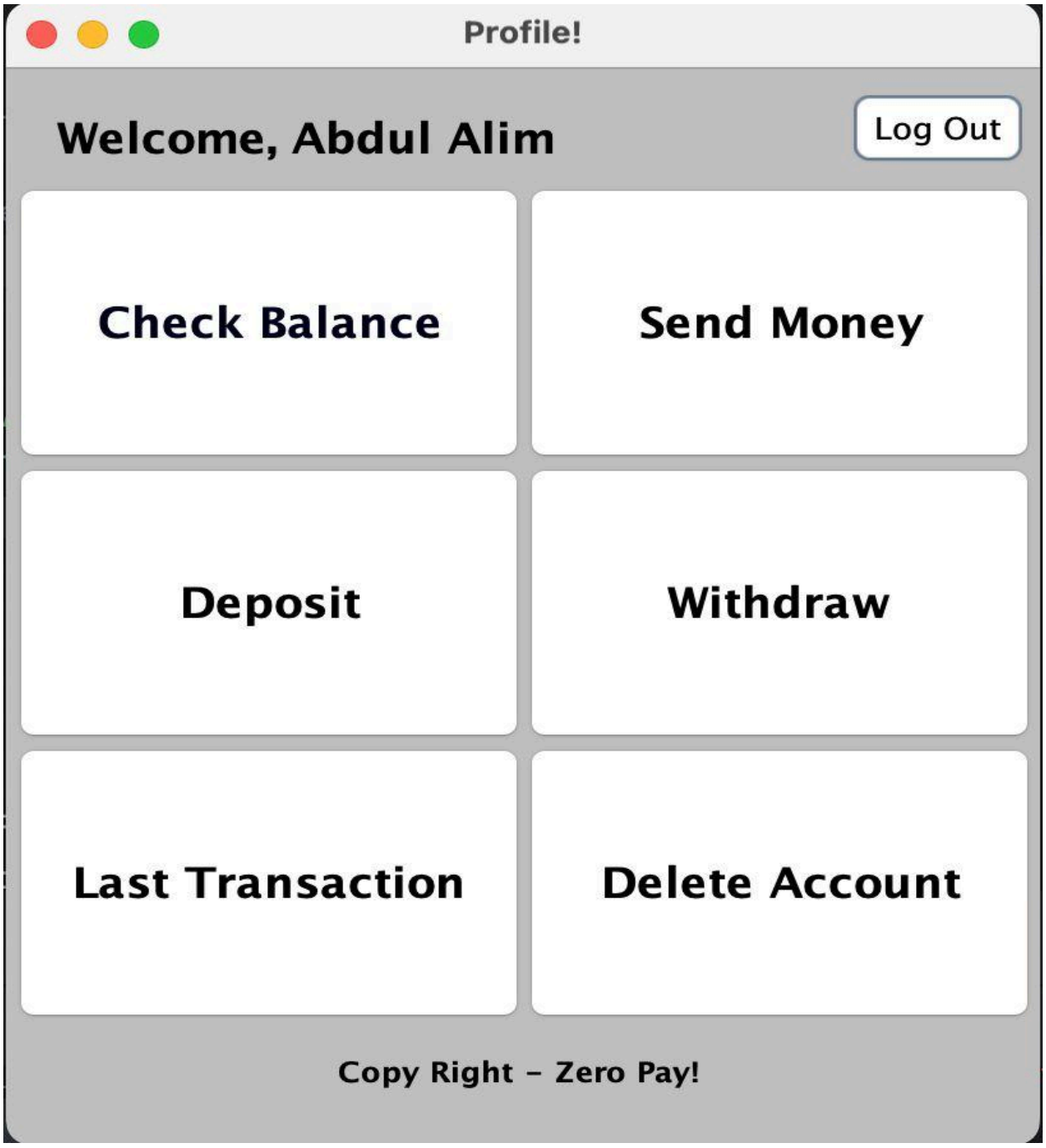
Admin customer access

User login page

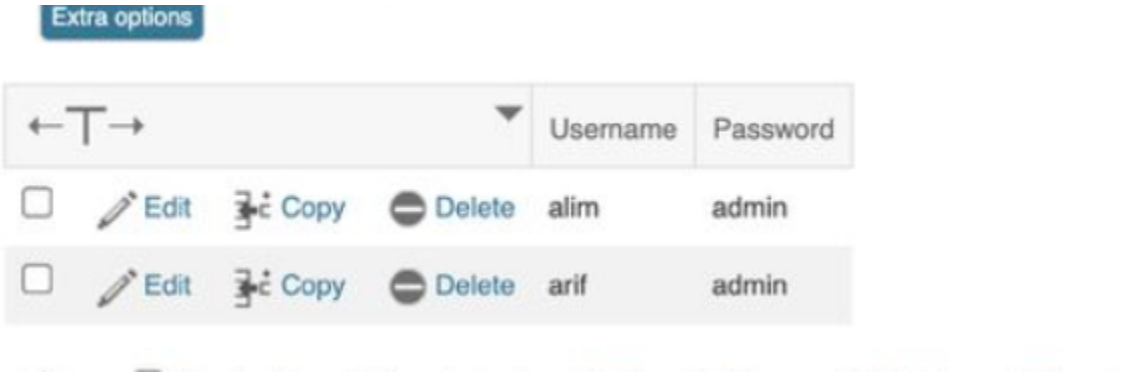
User signup page



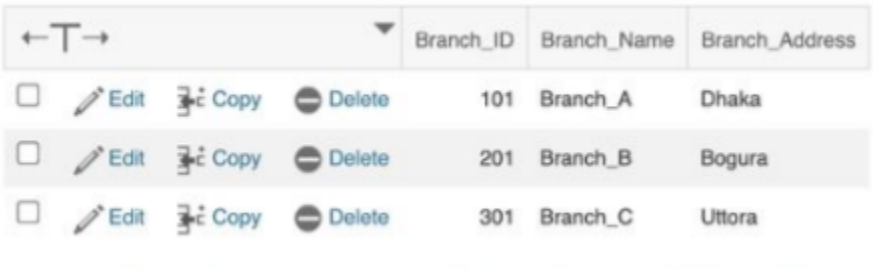
User profile



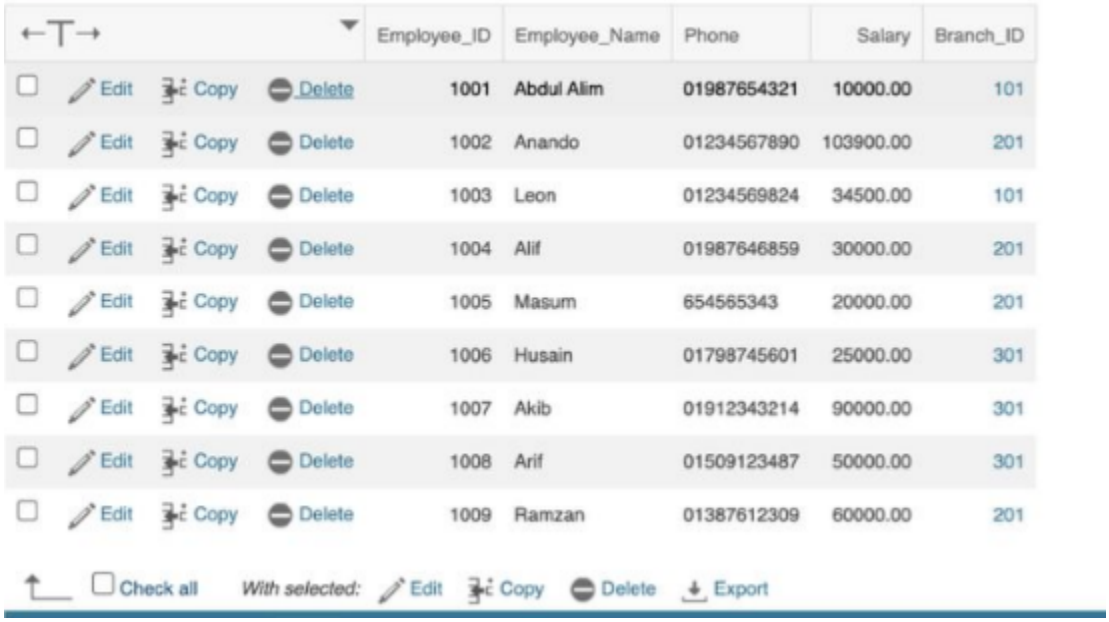
Displaying Admin information



Displaying Branch Information



Displaying Employee information



Displaying Customer_info information

	Phone	Full_Name	Email	Password	Branch_ID
Edit Copy Delete	01712345678	Ariful Islam	arif@gmail.com	1122	201
Edit Copy Delete	01787654321	Husain	husain@gmail.com	1111	101
Edit Copy Delete	01912345678	Abdul Alim	alim@gmail.com	1234	101

Check all *With selected:* Edit Copy Delete Export

Displaying Account_Balance information

Phone	Current_Balance	Transaction ▲ 1
01787654321	850	100 TK Received From 01912345678
01912345678	850	50 TK Received From 01712345678
01712345678	50	50 TK Send to 01912345678

2.2 Overall Project Plan

The project followed these phases: Planning → Design → Implementation → Testing → Finalization. Estimated time: 6 weeks. Total cost: Approx. $500 (hypothetical).

Chapter 3: Implementation and Results

This chapter describes the implementation process of the ZERO PAY banking system, covering database design, development, optimization techniques, and the testing process. It also evaluates the system's performance and discusses the outcomes.

3.1 Implementation

The database schema was created using MySQL. PHP was used to create a secure admin login system. The system uses CRUD operations and implements triggers for balance updates.

3.2 Database Design

- **Query Optimization:**
    1. **Indexing** was applied to primary and foreign keys such as phone and Branch_ID.
    2. **Joins** were optimized using subqueries and aliases.
    3. Frequently used queries were precompiled as **views** to enhance speed.

Example of optimized query:

```
SELECT Full_name, Current_Balance
FROM Customer_info
JOIN Accounts_Balance  ON Customer_info.phone = Accounts_Balance .phone
WHERE  Branch_ID = 10001;
```

- **Security:** Passwords hashed using MD5/SHA256
- **Data Integrity:** Foreign key constraints maintained

3.3 Results and Discussion

The system was successfully implemented with real-time data flow and basic banking operations. Future improvements include mobile UI and analytics integration.

3.3.1 Key Results

- **Functional admin and customer modules**: Seamless login, data viewing, and account management
- **Automated balance updates** via triggers and procedures
- **Scalable and normalized database schema** for real-world banking operations
- **Minimal manual error** due to automation and constraints

Chapter 4: Engineering Standards and Mapping

This chapter explores how the ZERO PAY project aligns with real-world engineering practices and professional standards. It highlights the system's impact on society, ethical considerations, sustainability, project management, and how it addresses complex engineering problems through technical and collaborative efforts

4.1 Impact on Society, Environment and Sustainability

4.1.1 Impact on Life

ZERO PAY simplifies daily banking activities, reducing the need for manual processes. It enhances the quality of life for both bank employees and customers by:

- Reducing processing time for transactions
- Minimizing human error through automation
- Allowing customers to manage accounts remotely, promoting financial inclusion

4.1.2 Impact on Society & Environment

The system helps digitize banking processes, contributing to the reduction of paper use and physical documentation. This reduces carbon footprint and supports eco-friendly initiatives. Additionally:

- Promotes data centralization and remote access, which supports digital transformation in the banking sector
- Enhances financial transparency, accountability, and accessibility

4.1.3 Ethical Aspects

The design of ZERO PAY prioritizes data privacy, security, and fair access. Ethical considerations include:

- Ensuring strong password protection and role-based access control
- Avoiding data redundancy and misuse
- Preventing unauthorized data manipulation using SQL injection prevention techniques and encryption

4.1.4 Sustainability Plan

To ensure the sustainability of ZERO PAY:

- The system is built using open-source tools (XAMPP, JAVA, MySQL) to reduce cost and dependency
- Code is modular and scalable for future upgrades
- Proper normalization and indexing improve performance and reduce hardware load

4.2 Project Management and Team Work

The ZERO PAY project was developed through collaborative efforts among all five team members, with responsibilities distributed based on technical strengths. Project management was handled through regular meetings, version control (using Git), and milestone tracking.

Project Timeline Overview:

1. **Week 1-2:** Requirement gathering, tool selection, team division
2. **Week 3:** Database schema design and ER diagram
3. **Week 4:** UI prototyping and backend logic development
4. **Week 5:** Implementation of triggers, procedures, and testing
5. **Week 6:** Final testing, debugging, and report writing

4.3 Complex Engineering Problem

The project required tackling multiple real-world challenges that reflect engineering complexity, such as transaction integrity, real-time balance tracking, and multi-entity relational mapping.

4.3.1 Mapping of Program Outcome

PO	Justification
PO1	Applied principles of database design to create normalized relational tables, minimizing redundancy and ensuring consistency
PO2	Analyzed data relationships in banking systems and implemented appropriate keys and constraints to ensure referential integrity
PO3	Designed triggers and stored procedures to handle balance updates and transactions automatically

4.3.2 Complex Problem Solving

To build ZERO PAY, we solved several engineering problems:

Technologies Used:

- **MySQL**: Database creation, query execution, indexing
- **JAVA** : Backend connection, logic, and data validation
- **JAVA Swing**: Frontend design and layout
- **JavaScript** (optional): For client-side form validation and enhanced interactivity

4.3.3 Engineering Activities

Engineering Activity	Details
EA1 - Range of Resources	Utilized MySQL, Java, Java Swing, GitHub, and XAMPP for system development.
EA2 - Level of Interaction	Integrated admin, customer, employee, and branch data for real-time, secure interaction
EA3 - Innovation	Implemented trigger-based event handling for transaction updates, removing need for manual balance management
EA4 - Consequences on Society & Environment	Reduced paper use, improved banking accessibility, supported secure online transactions
EA5 - Familiarity with Issues	Tackled familiar banking system challenges like unauthorized access, record duplication, data loss, and interface usability with practical coding and relational strategies

Chapter 5: Conclusion

5.1 Summary

ZERO PAY successfully demonstrates an end-to-end database application for banking. It is user-friendly, scalable, and secure.

5.2 Limitations

- No mobile interface yet
- No support for multi-currency
- Limited user roles

5.3 Future Work

- Mobile app integration
- Role-based admin hierarchy
- Real-time analytics dashboard

References

1. W3Schools
2. GeeksforGeeks
3. Youtube Tutorials
4. MySQL Documentation
5. GitHub Projects

.