**XYZ Bank**



**Session 2024 - 2028**

**Submitted by:**

Hussain Shahbaz 2024-CS-04

**Supervised by:** Sir Awais Hassan

**Course:**

CSC-102 Programming Fundamentals

Department of Computer Science

**University of Engineering and Technology**

**Lahore Pakistan**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Hussain shahbaz 2024-CS-04 CSC-102 Programming Fundamentals

Table of Contents

1. About XYZ Bank .............................................................................................. 4
   * Contribution: ........................................................................................................................ 4
   * Key Features: ....................................................................................................................... 4
2. Users of Application ............................................................................................................... 2
   * User: .............................................................................................................................. 4
   * Admin: ............................................................................................................................ 5
3. Functional Requirements ........................................................................................................ 5
4. Wireframes .............................................................................................................................. 7
   * Startup Interface: .................................................................................................................. 7
   * Admin Menu: ....................................................................................................................... 8
   * User Menu:................................................................................................................... 9

..................................................................................................................................................... 9

1. Data Structures (Parallel Arrays) .......................................................................................... 11
2. Function Prototypes .............................................................................................................. 13
3. Weakness in the Application ................................................................................................ 18
4. Future Directions .................................................................................................................. 18

Table of Figures

Figure 1-Startup Menu .................................................................................................................... 7

Figure 2-Owner Menu..................................................................................................................... 8

Figure 3-Manager Menu ................................................................................................................. 8

Figure 4-Employee Menu ............................................................................................................... 9

Figure 5-Customer Menu ................................................................................................................ 9

Figure 6-View Managers Menu .................................................................................................... 10

Figure 7-View Vehicles Menu ...................................................................................................... 10

1. **About Bank Management System**

The Bank Management System is a comprehensive management application designed to streamline banking operations, enhance customer experiences, and provide a secure, efficient environment for both administrators and users.

**Contribution:**The Bank Management System empowers administrators to strategically manage banking operations, optimize financial services, and improve overall customer satisfaction. Comprehensive reporting and analytics tools provide actionable insights into performance, enabling informed decision-making for better profitability and customer retention.

**Key Features:**This simplified application includes detailed specifications tailored to banking functionalities and services. Its intuitive interface ensures seamless navigation and effortless management for all user types, fostering an efficient and user-friendly environment.

**2. Users of the Application**

The Bank Management System supports different access levels for various user types. Users are divided into two primary categories: Admin and Customers. Each user type has access to specific commands and functionalities based on their roles and requirements. User authentication is implemented through username-password verification to ensure secure access.

**Admin:**

The admin manages the bank operations, including user accounts, loan processing, and financial records.

**Users:**

Customers interact with the system to manage their accounts, apply for loans, and access various banking services. The system supports up to 20 customers.

# Functional Requirements

Expected functionalities according to types of users:

|  |  |  |
| --- | --- | --- |
| ***User Type*** | ***Functionalities*** | ***Results*** |
| **User** | View transaction History | View the details of the transaction |
| Transact money | Add and remove the moneyand modify their details. |
| Transfer money | Transfer the money to other user through account number |
| Apply for loan | Can apply for a loan |
| Pay Bills | You can pay bills through the application |
| Pay taxes | Pay tax |

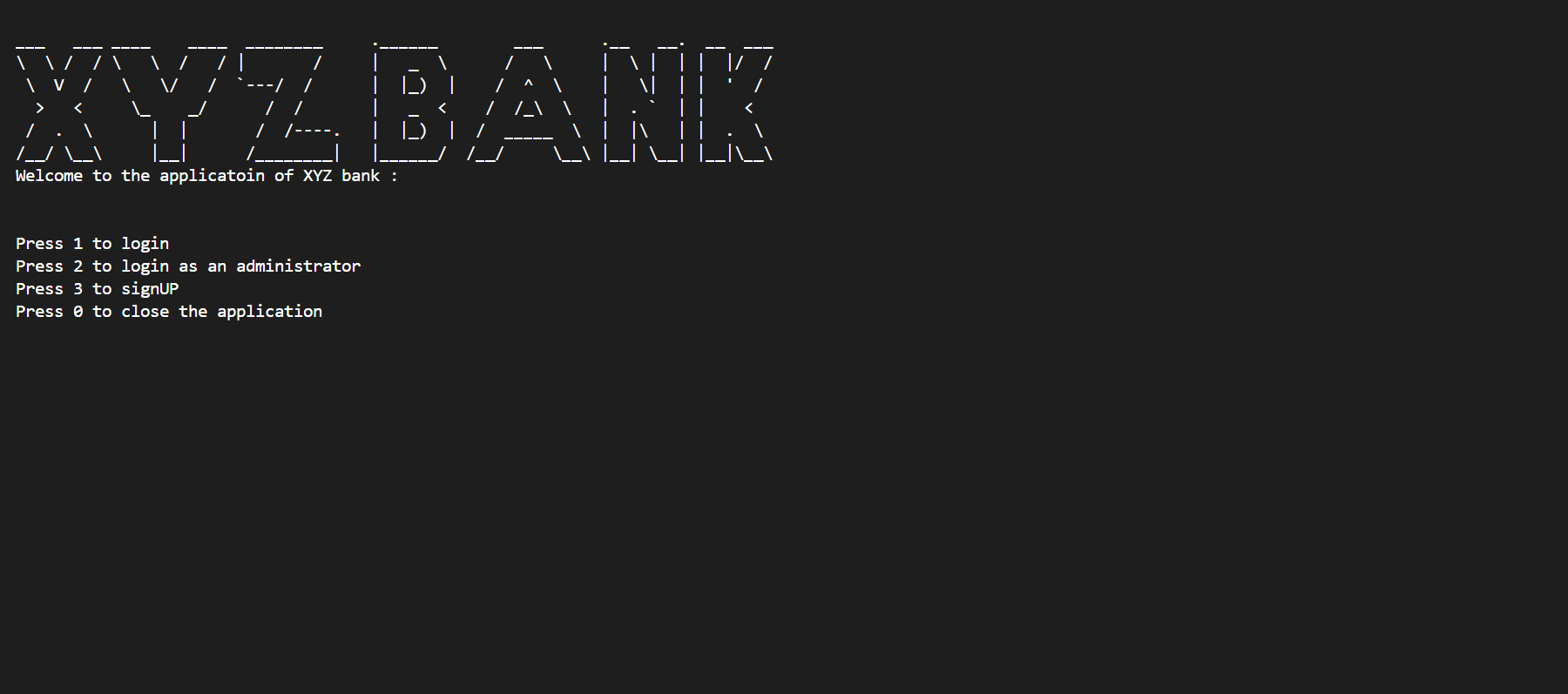
|  |  |  |
| --- | --- | --- |
| **Admin** | Retrieve data | Check user data |
| Approve loans | Approve loans applied by the user with soe interest |
| Delete user | Delete user through the database |
| Update user passwords | Add remove update password |
| Give bonuses | Give all the bonuses to users |
| Obtain banking history | Check banking history |

# Wireframes

A few wireframes of this application displayed in Command Line Interface are as follows**:**

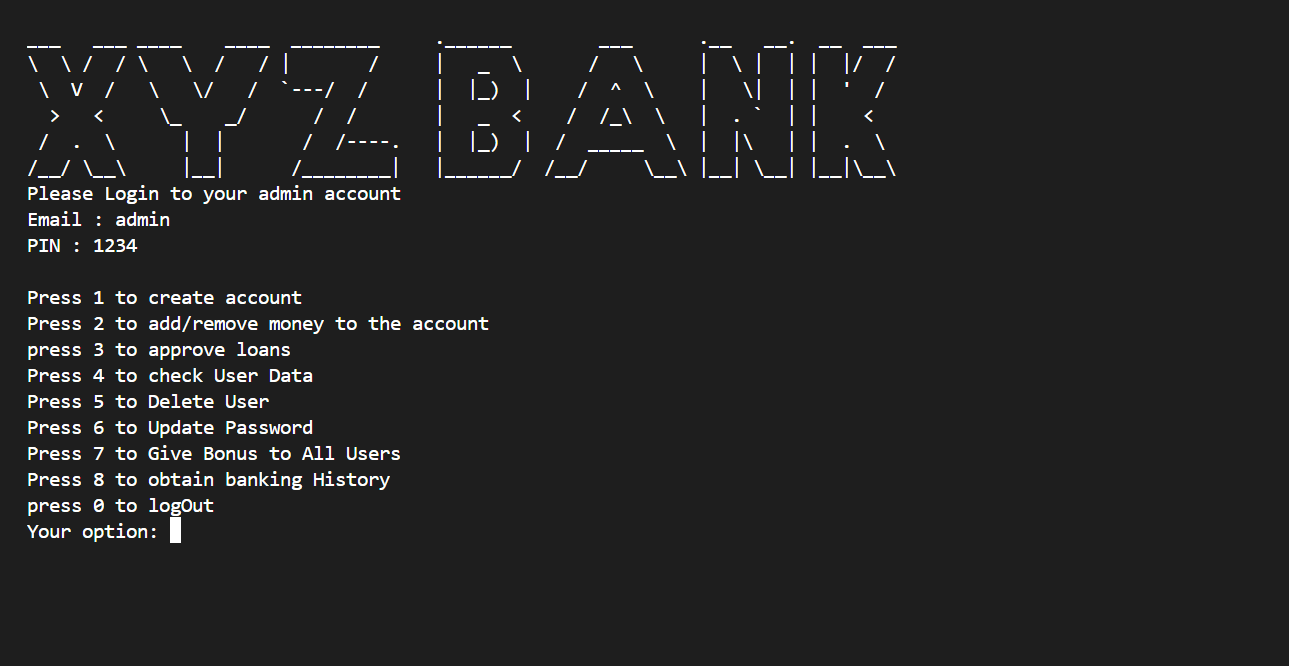
* **Startup Interface:**

Through this menu user type will be selected….



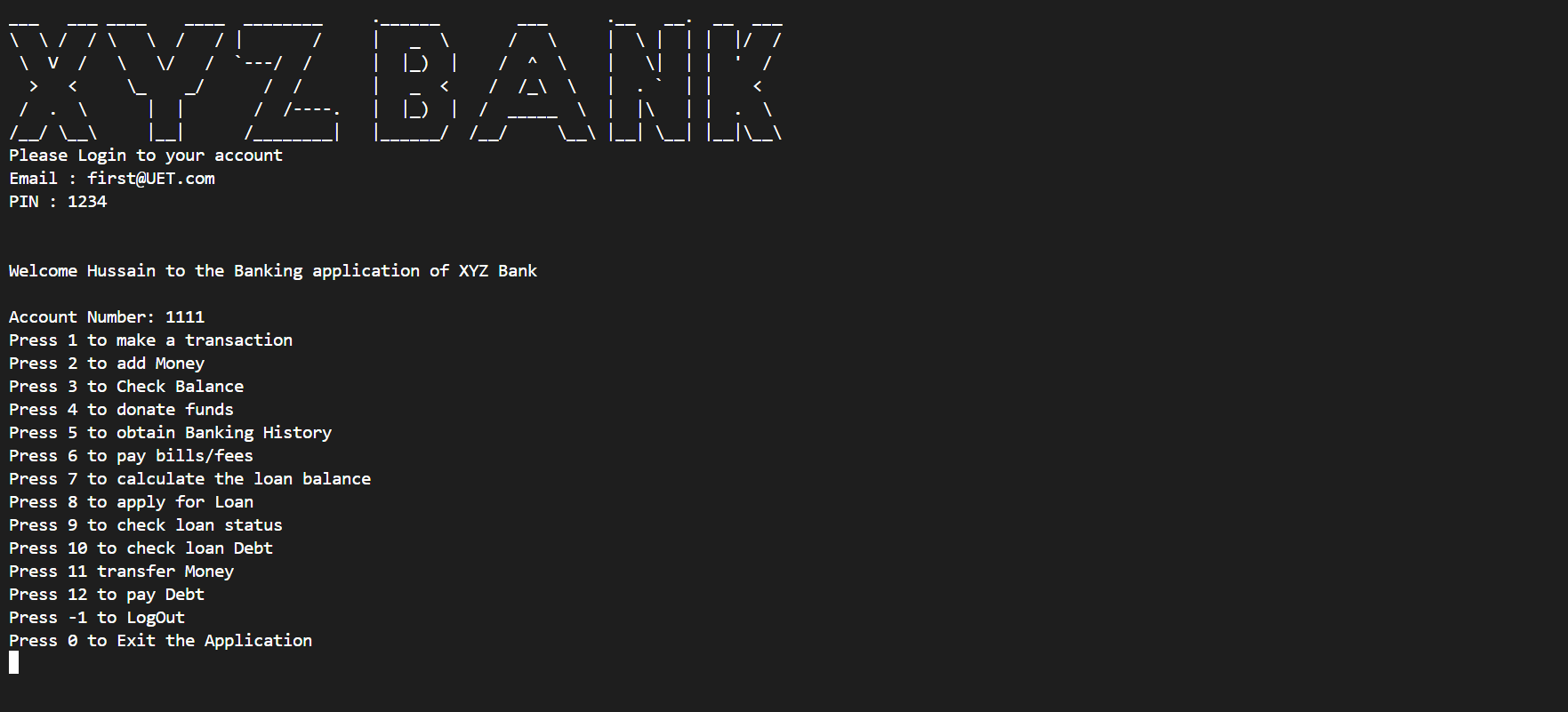
*Figure 1-Startup Menu*

* **Admin Menu:**

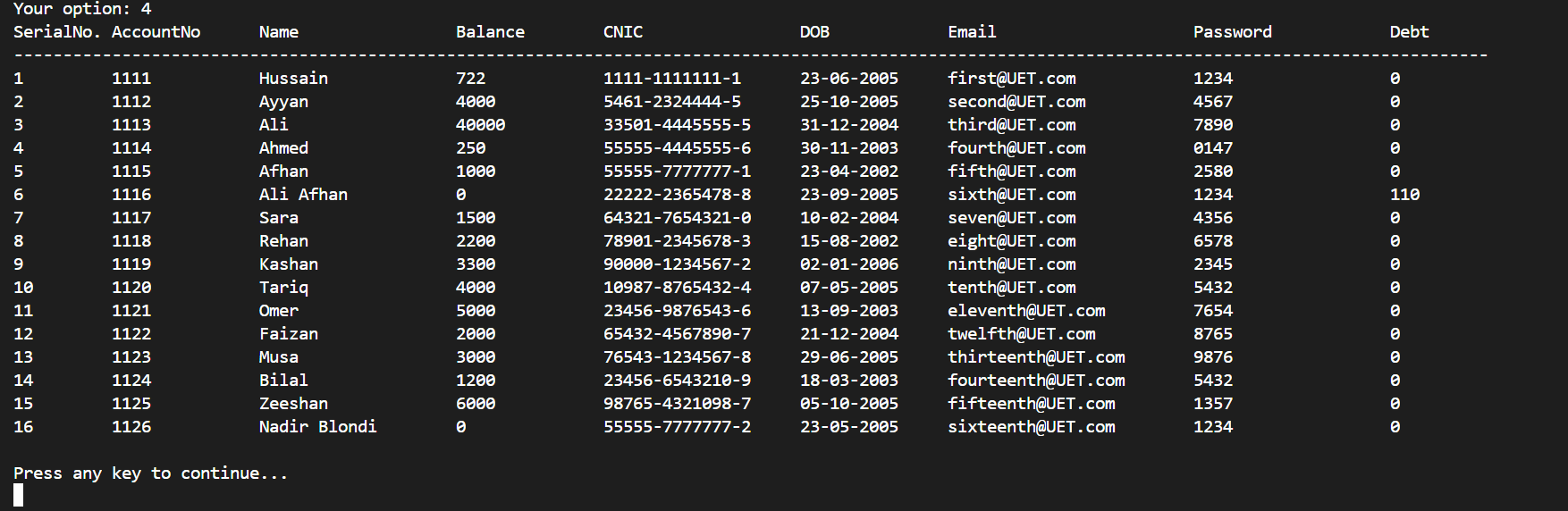


*Figure 2-Owner Menu*

* **User Menu:**



* **View Users Menu:**



*Figure 6-View Managers Menu*

# Data Structures (Parallel Arrays)

const int userNumbers = 20;

int userCount =0;

string nameData[userNumbers] =          {};

string emailData[userNumbers] =         {};

string passwordData[userNumbers] =      {};

string DOBData[userNumbers] =           {};

string CNICData[userNumbers]  =         {};

int accNumberData[userNumbers] =        {};

int balanceData[userNumbers] =          {};

float debtData[userNumbers] =           {};

int debt[userNumbers]                   {};

float loanAmountData[userNumbers] =     {};

float interest[userNumbers] =           {};

float yearsData[userNumbers];

bool loanApplicationData[userNumbers] =    {};

int loanApprovalData[userNumbers] =        {};

int userNo;

const int transactionLimit = 50;

string transactionHistory[userNumbers][transactionLimit];

int transactionCount[userNumbers] = {};

# Function Prototypes

void currentDateTime();

float myStof(const std::string& str);

void anyKey();

void loadData();

void storeData();

string getField(string record, int field);

int myStoi(string str);

void printInterface();

void Footer();

void pressAnyKey();

int getAccNo(int accNo);

bool signIN(string username, string password);

bool checkUserCapacity();

int getUserEmail(string email);

void DeleteUserData(int userNo);

bool digitValidator(string num);

bool CNICval(string cnic);

bool isLeapYear(int year);

bool isValidDOB(int day, int month, int year);

bool isValidDOBFormat(string dob);

bool checkCom(string email);

bool checkAt(string email);

bool checkSpace(string email);

bool emailVal(string email);

int getCNIC(string cnic);

void printHeader();

void body();

void ApplyforLoan();

void checkLoanStatus();

void transaction();

void addMoney();

void checkBalance();

void donations();

void History(int userNo);

void payBills();

void payTax();

void calculateLoanBalance();

void checkDebt();

void payDebt();

void transferMoney();

void logInSuccessful();

void storeHistory();

void loadHistory();

void signUP();

void adminLogin();

void adminAddSubtractMoney();

void adminViewUser();

void adminLoan();

bool adminDelete(string emailIN);

void adminUpdatePassword(string emailIN);

void adminGiveBonus(int amount);

void adminViewHistory(string email);

void adminFunction();

# Application Feedback

This **Bank Management System** ensures seamless operation for administrators and customers while prioritizing security, transparency, and user-friendliness in managing banking and financial services.

# Future Directions

* Add the details of of taxes.
* Add graphics
* Encrypt the data base so no one else can access
* Financial help bot