**Mobile Payment System**

A Project Report

Submitted in partial fulfilment of the

Requirements for the award of the Degree of

**BACHELOR OF ENGINEERING**

IN

**COMPUTER SCIENCE & ENGINEERING**

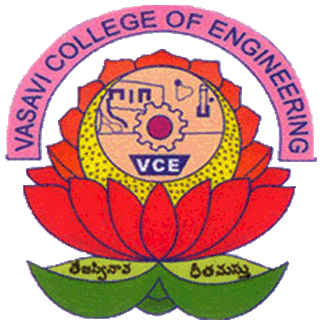
By

**B.HARIKA**

1602-22-733-304

**T.DIVYA**

1602-22-733-305



**Department of Computer Science&Engineering**

**Vasavi College of Engineering(Autonomous)**

**DECLARATION BY THE CANDIDATE**

We, **BASABOINA HARIKA** and **TALLAPALLY DIVYA** bearing hall ticket numbers, **1602-22-733-304** and **1602-22-733-305** hereby declare that the project report entitled “MOBILE PAYMENT SYSTEM” under the guidance of **Dr.D.BASWARAJ,** professor Department of Computer Science & Engineering, VCE, Hyderabad, is submitted in partial fulfilment of the requirement for the award of the degree of **Bachelor of Engineering in ComputerScience & Engineering.**

This is a record of bonafide work carried out by us and the results embodied in this project report have not been submitted to any other university or institute for the award of any other degree or diploma.

**BASABOINA HARIKA(1602-22-733-304)**

**TALLAPALLY DIVYA (1602-22-733-305)**

Vasavi College of Engineering (Autonomous)

(Affiliated to Osmania University)

Hyderabad-500 031

Department of Computer Science & Engineering



**BONAFIDE CERTIFICATE**

This is to certify that the project entitled **“MOBILE PAYMENT SYSTEM”** being submitted by **BASABOINA HARIKA** and **TALLAPALLY DIVYA** bearing **1602-22-733-304** and **1602-22-733-305** in partial fulfilment of the requirements for the award of the degree of Bachelor of Engineering in Computer Science & Engineering is a record of bonafide work carried out by him/her under my guidance.

**Dr.D.BASWARAJ, Dr. T. ADILAKSHMI, Professor , Professor &HOD, Internal**

**Guide, Dept. of CSE.**

ACKNOWLEDGEMENT

We take this opportunity with pride and enormous gratitude, to express the deeply embedded feeling and gratefulness to our respectable guide **Dr.D.BASWARAJ sir, professor** Department of Computer Science and Engineering, whose guidance was unforgettable and filled with innovative ideas as well as his constructive suggestions has made the presentation of my mini project a grand success.

We are thankful to **Dr. T. ADILAKSHMI**, Head of Department (CSE), **Vasavi College of Engineering** for the help during our course work.

Finally we express our gratitude to the management of our college, **Vasavi College of Engineering** for providing the necessary arrangements and support to complete our project work successfully.

**ABSTRACT**:

The Mobile Payment System Project aims to simulate a basic payment system where users can register, check their account balance, and perform fund transfers. We’ll use a text file to store user data (for simplicity). And another text file to maintain the transaction history .this is a simplified version, and in a real-world scenario, security and robustness would be critical.

**Project Outline:**

User Registration:

Users can register by providing their details (name, phone number, initial balance).

Store user data in a text file (e.g., users.txt).

Balance Inquiry:

Users can check their account balance.

Retrieve the balance from the user’s record in the text file.

Fund Transfer:

Users can transfer funds to other registered users.

Deduct the transferred amount from the sender’s balance and add it to the recipient’s balance.

Transaction History:

User registration timing details and payment history can stores in a text file.

Removing User:

User and all his/her details can be removed based on given phone number. So that the user cannot have access anymore.

**TABLE OF CONTENTS**

1. INTRODUCTION……………

1.1 OVERVIEW

1.2 PROJECT SCOPE

2. SYSTEM REQUIREMENTS……………………

2.1 SOFTWARE REQUIREMENTS

2.2 HARDWARE REQUIREMENTS

3. IMPLEMENTATION…………………………….

3.1 PROJECT CODE

4. OUTPUTS OF PROJECT CODE………………

5. CONCLUSION

6.REFERENCES

**1.INTRODUCTION:**

The **Mobile Payment System** is a robust and user-friendly application designed to facilitate financial transactions, manage user accounts. In an increasingly digital world, mobile payment systems play a crucial role in simplifying financial interactions for users. Our system aims to enhance convenience, security, and accessibility for individuals.

**1.1OVERVIEW:**

Our system consists of four primary components:

1. User Registration **(**register.sh**):**
   * Allows users to create accounts by providing essential details such as name, pin,initial balance and phone number.
   * Validates user input and ensures uniqueness of account information.
   * Stores user data securely in the users.txt file.
2. Balance Inquiry **(**balance.sh**):**
   * Enables users to check their account balances.
   * Retrieves balance information from the users.txt file.
   * Displays the balance to the user.
3. Fund Transfer **(**transfer.sh**):**
   * Facilitates secure money transfers between registered users.
   * Validates sender and recipient information.
   * Updates account balances and records transactions in the transactions.txt file.
4. User-Friendly Menu System:

* Create an intuitive menu system for the shell scripts.
* Users should be able to navigate through options (e.g., check balance, transfer funds) easily.
* Enhance the menu’s visual appear.

**1.2 PROJECT SCOPE:**

The **Mobile Payment System** aims to provide a comprehensive solution for managing financial transactions, user accounts.

**2. SYSTEM REQUIREMENTS**

**2.1 Software Requirements:**

The major software requirements of our project include:

1. Operating system: Linux(ubuntu)

2. Tools: bash-shell(terminal)

3.Language: shell scripting

**2.2 Hardware Requirements:**

The hardware requirements that preferably map towards the software are as follows:

1. RAM: 4 GB

2. Processor: AMD RYZEN7

**3.IMPLEMENTATION:**

**3.1 Project Code:**

**1.register.sh**

#! /usr/bin/bash

is\_valid\_phone\_number() {

    local phone\_number="$1"

    if [[ "$phone\_number" =~ ^[0-9]{10}$ ]]; then

        return 0

    else

        return 1

    fi

}

is\_valid\_pin\_number() {

    local pin\_number="$1"

    if [[ "$pin" =~ ^[0-9]{4}$ ]]; then

        return 0

    else

        return 1

    fi

}

# Prompt user for registration details

read -p "Enter your name: " name

read -p "Enter your phone number: " phone\_number

read -p "Enter initial balance: " balance

# In register.sh (after user registration):

read -sp "Enter  your 4-digit PIN: " pin

# Check if the phone number already exists

if grep -q "$phone\_number" users.txt; then

    echo "User with phone number $phone\_number is already registered."

else

    if ! is\_valid\_phone\_number "$phone\_number"; then

      echo "The phone number must be a 10-digit number."

    else

        if ! is\_valid\_pin\_number "$pin"; then

          echo "The pin must be a 4-digit number."

        else

          echo "$name|$phone\_number|$balance|$pin" >> users.txt

          echo "Registration successful!"

          echo "$(date '+%Y-%m-%d %H:%M:%S') | $name registered" >> transactions.txt

        fi

    fi

fi

**2.balance.sh**

#! /usr/bin/bash

# Prompt user for phone number

read -p "Enter your phone number: " phone

read -sp "Enter your PIN: " entered\_pin

#Search for user in users.txt

balance=$(grep "$phone|" users.txt | cut -d '|' -f 3)

if [ -z "$balance" ]; then

    echo "User not found. check the entered phone number"

else

    stored\_pin=$(grep "$phone|" users.txt | cut -d '|' -f 4)

    if [ "$entered\_pin" != "$stored\_pin" ]; then

        echo "Incorrect PIN. Access denied."

    else

        echo "Your balance: $balance"

    fi

fi

**3.delete\_user.sh**

#! /usr/bin/bash

# Read the phone number from user input

read -p "Enter the phone number to delete: " phone\_number

# Check if the user exists in the users.txt file

if grep -q "$phone\_number" users.txt; then

    # Remove the user entry from users.txt

    sed -i "/$phone\_number/d" users.txt

    echo "User with phone number $phone\_number has been deleted."

else

    echo "User with phone number $phone\_number does not exist."

fi

**4.transfer.sh**

#! /usr/bin/bash

# Prompt user for sender and recipient phone numbers

read -p "Enter sender's phone number: " sender

read -p "Enter recipient's phone number: " recipient

read -p "Enter transfer amount: " amount

read -sp "Enter your PIN: " entered\_pin

# Update balances in users.txt

sender\_balance=$(grep "$sender|" users.txt | cut -d '|' -f 3)

recipient\_balance=$(grep "$recipient|" users.txt | cut -d '|' -f 3)

if [ -z "$sender\_balance" ] || [ -z "$recipient\_balance" ]; then

    echo "Invalid sender or recipient. check the entered phone numbers."

else

    stored\_pin=$(grep "$sender|" users.txt | cut -d '|' -f 4)

    if [ "$entered\_pin" != "$stored\_pin" ]; then

       echo "Incorrect PIN. Access denied."

    elif [ "$sender\_balance" -lt "$amount" ]; then

       echo "Insufficient balance."

    else

       new\_sender\_balance=$((sender\_balance - amount))

       new\_recipient\_balance=$((recipient\_balance + amount))

       sed -i "s/$sender|$sender\_balance/$sender|$new\_sender\_balance/" users.txt

       sed -i "s/$recipient|$recipient\_balance/$recipient|$new\_recipient\_balance/" users.txt

       echo "Transfer successful!"

       echo "$(date '+%Y-%m-%d %H:%M:%S') | $sender transferred $amount to $recipient" >> transactions.txt

    fi

fi

**5.menu.sh**

#! /usr/bin/bash

# Initialize ncurses

clear

tput civis  # Hide cursor

echo "Mobile Payment System Menu"

echo "1. Register User "

echo "2. Check Balance"

echo "3. Make a Transfer"

echo "4. Delete a User "

echo "5. Exit"

while :; do

    echo -n "Enter your choice [1-5]: "

    read option

    case $option in

        1)

            # Execute registration logic (in register.sh)

            ./register.sh  # Replace with the actual path to your register.sh script

            ;;

        2)

            # Execute balance checking logic (in balance.sh)

            ./balance.sh  # Replace with the actual path to your balance.sh script

            ;;

        3)

            # Execute transfer logic (in transfer.sh)

            ./transfer.sh  # Replace with the actual path to your transfer.sh script

            ;;

        4)

            # Execute delete logic (in delete\_user.sh)

            ./delete\_user.sh  # Replace with the actual path to your transfer.sh script

            ;;

        5)

            echo "Exiting..."

            exit

            ;;

        \*)

            echo "Invalid choice. Please select 1, 2, 3, 4 or 5."

            ;;

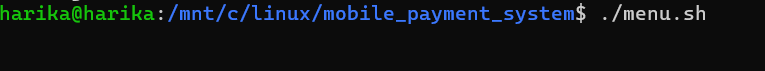
    esac

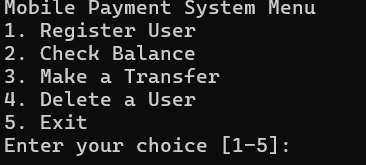
done

# Clean up ncurses

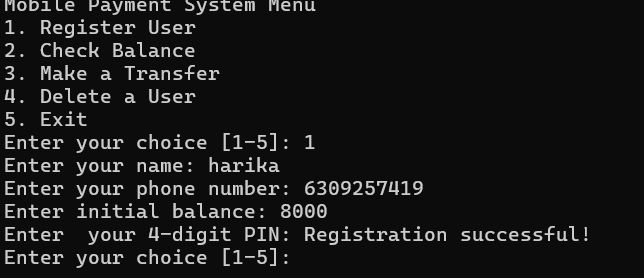
tput cnorm  # Restore cursor

**4.OUTPUTS OF PROJECT CODE:**

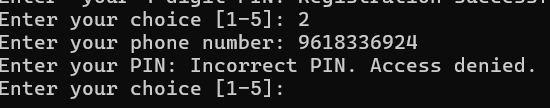




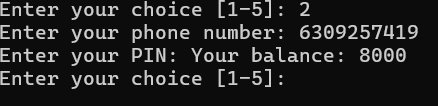
**Registering a new user:**



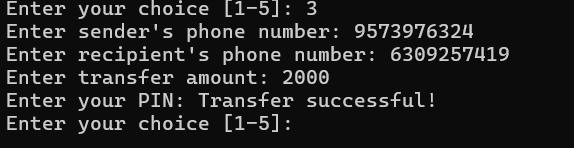
**Validating user input:**

****

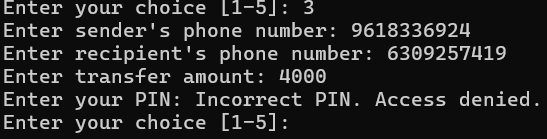
**Checking balance:**

****

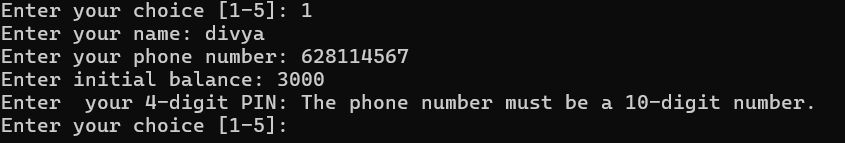
**Transferring amount:**

****

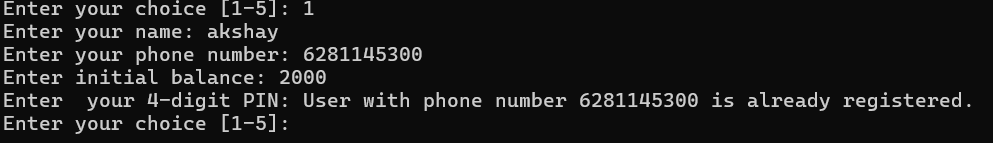
**Validating the pin while transferring money:**

****

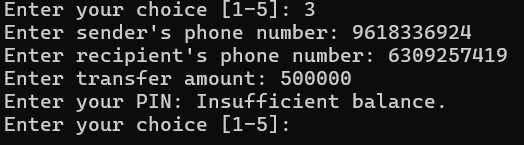
**Validating the phone number:**

****

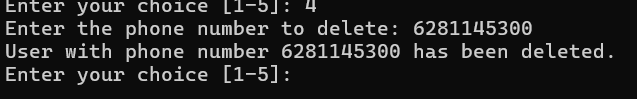
**No duplicate registrations:**

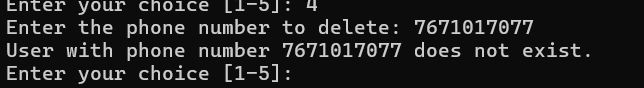
****

**Checking for sufficient balance:**

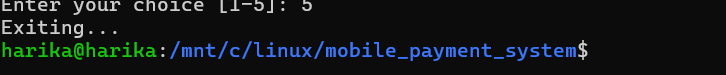
****

**Deleting a user:**

****

****

**Exiting from the menu:**

****

**users.txt file:**

ravi|9618336924|5000|4567

anitha|9573976324|6000|1234

harika|6309257419|10000|1234

**transactions.txt file:**

2024-05-24 13:39:31 | harika registered

2024-05-24 13:50:13 | ravi registered

2024-05-24 13:53:25 | anitha registered

2024-05-24 14:00:29 | 6281145300 transferred 1000 to 9618336924

2024-05-24 14:48:28 | harika registered

2024-05-24 22:24:37 | 9573976324 transferred 2000 to 6309257419

**5.CONCLUSION:**

In summary, our **Mobile Payment System** streamlines financial interactions, offering user-friendly features like registration, balance inquiries, and fund transfers. We’ve overcome challenges such as installing ncurses . Future enhancements include notifications, SMS functionality, an intuitive menu system, and personalized interfaces. By prioritizing security and code quality, we’re poised to revolutionize digital finance. Our journey through the Bash shell empowered us to build robust features.

**Challenges Overcome**

1. **Installing ncurses:**
   * Successfully installed the ncurses package for interface enhancements.
   * Using the command: sudo apt-get install libncurses-dev.

2. **Switching to Linux (Ubuntu):**

* + Transitioning to Ubuntu provided a stable and developer-friendly environment.
  + We embraced open-source tools and leveraged the power of the terminal.

**Future Enhancements**

1. **Notifications:**
   * Extend notification capabilities to include email alerts, SMS messages, and in-app notifications.
   * Enhance user engagement by providing timely updates.
2. **SMS Functionality:**
   * Implement SMS-based transactions, allowing users to perform actions via text messages.
   * Explore integration with external SMS gateways for seamless communication.
3. **User-Friendly Menu System:**
   * Refine the menu system using ASCII art for visual appeal.
   * Ensure intuitive navigation for users.
4. **Custom Interface:**
   * Personalize the interface with a custom header or footer using ASCII art.
   * Display the system name or logo prominently.

**6.REFERENCES:**

1.By taking the help of youtube videos we installed linux operating system.

2. Taking the guidance of internet we downloaded required packages into linux.

3. <https://soloway.tech/blog/how-to-build-successful-mobile-payment-system/>

4.Github