PROJECT

PROPOSAL OF BUS RESERVATION SYSTEM

TEAM MEMBERS

1. Shahriar Kabir Omi
2. Mariya Mirza Mim
3. Esrat Esha

Submitted to-

Professor Dr. Md. Mahfuzur Rahman

OUR TEAM

MARIYA MIRJA MIM

Project Planner: I planned the whole project. Ensuring quality control and delivering a successful project within scope and budget was also my job.

SHAHRIAR KABIR OMI

Team Leader: As a team leader, I handled the Presentation and coding. I also allocated tasks for my team members and monitored them.

ESHRAT ESHA

Script Writer: I was tasked with developing the script or storyline for this project, including dialogue, narration, and scene descriptions.

PROJECT OF BUS RESERVATION SYSTEM

Welcome to our Bus Reservation System! With the power of C language, we have developed an advanced platform to revolutionize how bus ticket bookings are managed. Our system offers a seamless user experience, allowing passengers to easily search for available buses, reserve seats, and make secure payments from the comfort of their own devices.

CONCEPT

The concept of the "Bus Reservation System" project revolves around creating a software application that automates and simplifies the process of booking bus tickets. The primary objective is to develop a user-friendly system that allows passengers to search for available buses, select seats, and make secure online payments. By digitizing and streamlining the reservation process, the project aims to improve bus ticket bookings' efficiency, accuracy, and convenience for passengers and bus operators.

CODE

```c
#include <stdio.h>
#include <stdlib.h>

#define MAX_SEATS 40

typedef struct {
    char name[50];
    int seatNumber;
} Passenger;

void reserveSeat(Passenger passengers[], int seatNumber) {
    if (seatNumber < 1 || seatNumber > MAX_SEATS) {
        printf("Invalid seat number. Please try again.\n");
        return;
    }

    if (passengers[seatNumber - 1].seatNumber != 0) {
        printf("Seat already reserved. Please choose another seat.\n");
        return;
    }

    printf("Enter passenger name: ");
    scanf("%s", passengers[seatNumber - 1].name);

    passengers[seatNumber - 1].seatNumber = seatNumber;
    printf("Seat %d reserved for %s.\n", seatNumber, passengers[seatNumber - 1].name);
}

void cancelReservation(Passenger passengers[], int seatNumber) {
    if (seatNumber < 1 || seatNumber > MAX_SEATS) {
        printf("Invalid seat number. Please try again.\n");
        return;
    }

    if (passengers[seatNumber - 1].seatNumber == 0) {
        printf("No reservation found for seat %d.\n", seatNumber);
        return;
    }
```

```c
 printf("Cancelled reservation for seat %d (Passenger:
%s).\n", seatNumber, passengers[seatNumber - 1].name);
    passengers[seatNumber - 1].seatNumber = 0;
    passengers[seatNumber - 1].name[0] = '\0';
}
void displaySeatStatus(Passenger passengers[]) {
    printf("==============================================
==\n");
    printf("                    Seat
Status                  \n");
    printf("==============================================
==\n");
    printf("  Seat No.    Passenger Name\n");
    printf("----------------------------------------------
--\n");
    for (int i = 0; i < MAX_SEATS; i++) {
        if (passengers[i].seatNumber != 0) {
            printf("    %2d         %s\n",
passengers[i].seatNumber, passengers[i].name);
        }
    }
    printf("==============================================
==\n");
}
int main() {
    Passenger passengers[MAX_SEATS] = {0};
    int choice, seatNumber;
while (1) {
        printf("\nBus Reservation System\n");
        printf("1. Reserve a seat\n");
        printf("2. Cancel seat reservation\n");
        printf("3. Display seat status\n");
        printf("4. Exit\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);

        switch (choice) {
            case 1:
                printf("Enter seat number to reserve: ");
                scanf("%d", &seatNumber);
                reserveSeat(passengers, seatNumber);
                break;
            case 2:
                printf("Enter seat number to cancel reservation:
");
                scanf("%d", &seatNumber);
                cancelReservation(passengers, seatNumber);
                break;
            case 3:
                displaySeatStatus(passengers);
                break;
            case 4:
                exit(0);
            default:
                printf("Invalid choice. Please try again.\n");
        }
    }

    return 0;
}
```

OUTPUT

```
Bus Reservation System
1. Reserve a seat
2. Cancel seat reservation
3. Display seat status
4. Exit
Enter your choice:
```

```
Bus Reservation System
1. Reserve a seat
2. Cancel seat reservation
3. Display seat status
4. Exit
Enter your choice: 1
Enter seat number to reserve: 1
Enter passenger name: Shahriar
Seat 1 reserved for Shahriar.
```

```
                    Seat Status
=============================================
  Seat No.     Passenger Name
---------------------------------------------
      1           Shahriar
=============================================
```

PROCESS

STEP 1

User Registration and Login:
Implement a user registration system where new users can create an account

STEP 2

Bus and Seat Management:Create a database or data structure to store information about buses, including their routes, schedules, and seat availability.

STEP 3

Reservation and Payment:Develop the booking process that allows users to reserve seats on selected buses.

STEP 4

Ticket Management and Reporting:
Implement a ticket management system to store and retrieve booking information for future reference.

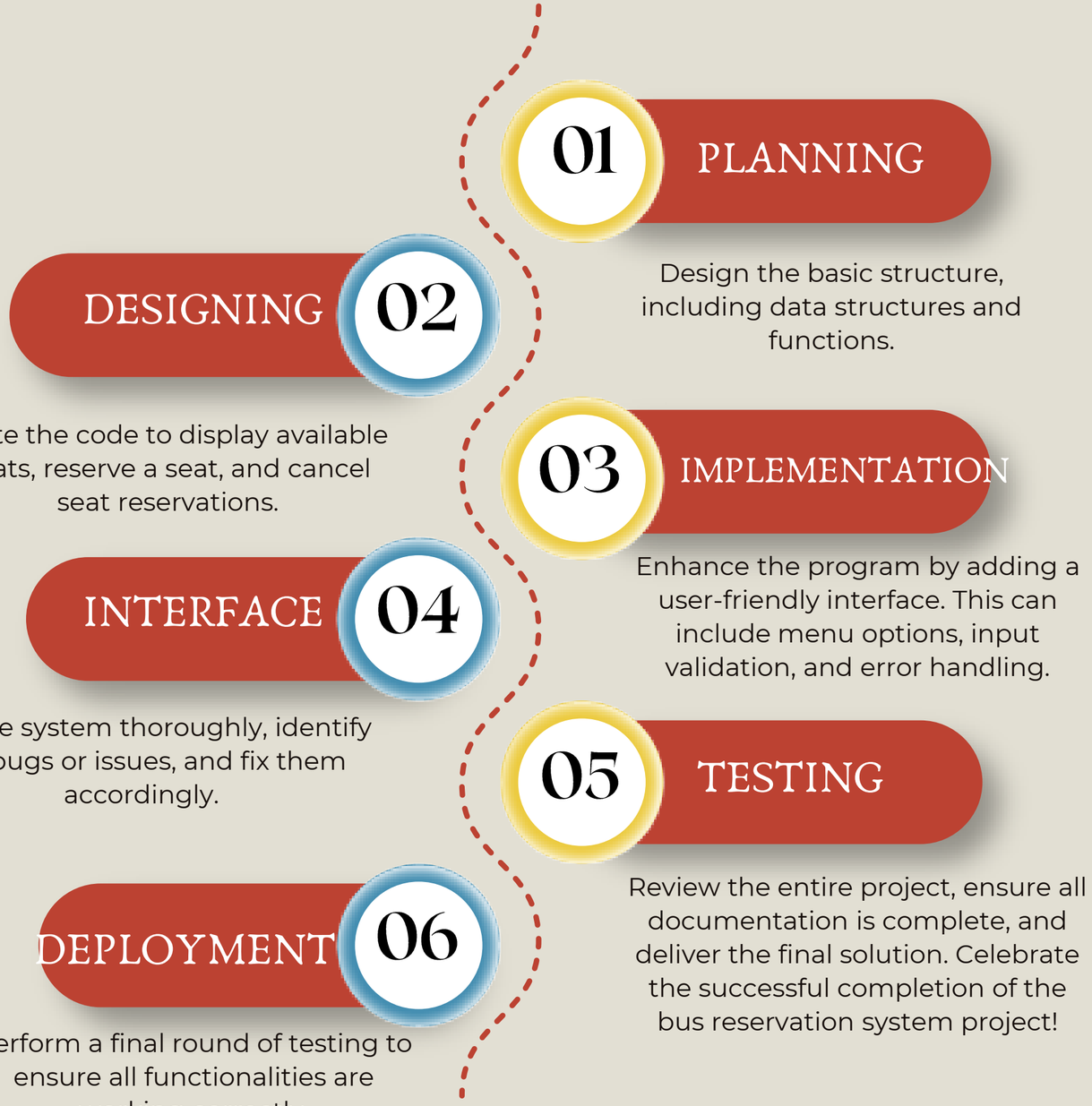
TIMELINE

Planning and Design (4 hours): Define the requirements and objectives of the system. Design the basic structure, including data structures and functions.

Implementation of Seat Reservation Logic (6 hours): Write the code to display available seats, reserve a seat, and cancel seat reservations. Test the functionality to ensure it works as expected.

User Interface Development (2 hours): Enhance the program by adding a user-friendly interface. This can include menu options, input validation, and error handling.

Testing and Bug Fixing (2 hours): Test the system thoroughly, identify any bugs or issues, and fix them accordingly. Conduct different test scenarios to ensure the reliability of the system.



EXPECTED RESULTS

- Efficient Booking Process: The system should allow users to search for available buses, view seat availability, select desired seats, and complete the booking process seamlessly.
- Accurate Seat Management: The system should accurately manage and display seat availability in real-time. It should prevent double bookings, ensure seat selection accuracy, and update seat availability promptly as bookings are made or canceled.

3.Secure Payment Processing: The system should integrate secure payment gateways or implement a reliable payment system to facilitate online transactions.
4.Ticket Generation and Management: The system should generate digital tickets or booking confirmations for successful reservations. These tickets should contain all relevant details, such as passenger information, bus details, seat numbers, and travel dates.

The expected result of the Bus Reservation System project is to develop a functional and user-friendly software application that successfully automates and improves the bus ticket booking process.

Thank You!