DESIGN DOCUMENTATION

Ticketing System with Multiple Stops

Table of Contents

1. Introduction

Ticketing system is a computerised system for easy reservation of seats in advance. Ticketing System with Multiple Stops is a system to provide route information of the bus with all stops along with the fares. It allows us to view the booked and vacant seats. It facilitates booking tickets in an optimal way by checking if the seat requested by the user is available or not, thereby generating the ticket if the seat is available. Tickets records are stored in a file which helps to analyse the bus/train occupancy.

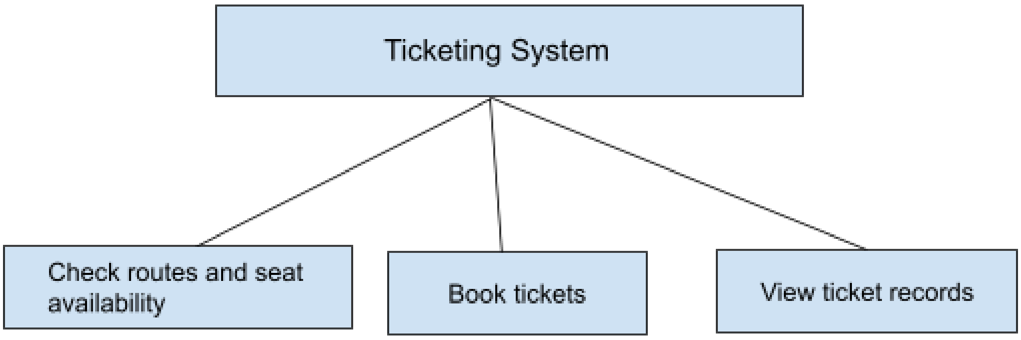

Fig 1.1 Design Overview

1.1 Design Purpose

The objective of the project is to provide a ticketing system that ensures seat allocation and ticket booking in an optimal manner with ease. To summarise, it covers the following aspects -

- Efficient booking management
- Seat allocation in an optimal way
- Ticket generation and storing in a file as backup

1.2 Scope

The scope of the document is to analyse and design each of the functionalities based on the requirements. Functionalities include displaying the route details and fare details to the user, efficient display of booked and vacant seats, ticket booking in an optimal way and storing the ticket in a file. The system has a menu driven interface and is divided into different modules.

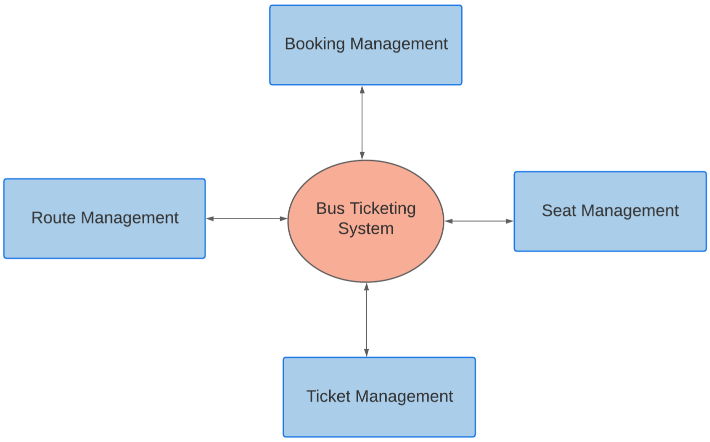

Fig 1.2 Design Scope

1.3 Functional Overview

1. Displaying the route information with regards to Boarding point, Drop off point and the distance from start point.
2. Displaying the details of Route with all the stops.
3. Displaying the price chart as a matrix.
4. Read Trip Details: Boarding point, Drop off point, Fetch the fare from matrix and issue the ticket and store details in File.
5. Display the number of booked tickets and vacant seats.
6. Ticket booking should be done in an optimal way.

2. Design Overview

2.1 Design Objectives

The ticketing system will comprise of modules depending on the various functionalities namely-

1. Menu page
2. Display Routes module
3. Display Fare Matrix module
4. Display Seat Matrix module
5. Book tickets module
6. Display tickets module

User enters the system and chooses appropriate options from the menu to view routes and fare information, view seat availability and book tickets if needed, view ticket records and finally quit.

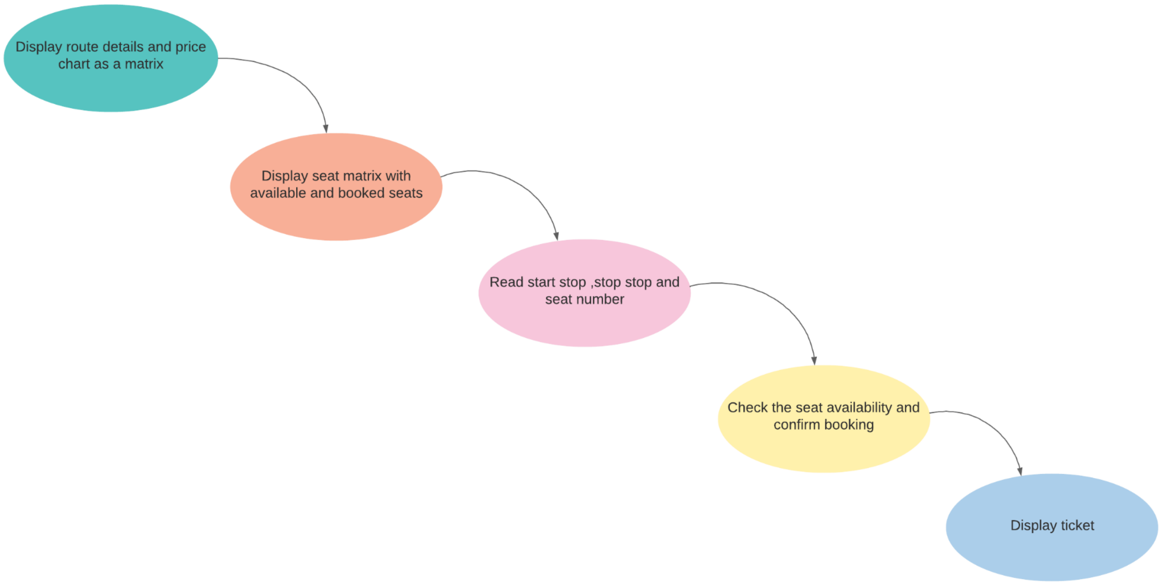
Display route details and price chart as a matrix
Display seat matrix with available and booked seats
Read start stop ,stop stop and seat number
Check the seat availability and confirm booking
Display ticket

Fig 2.1 Design Architecture

2.2 Data Flow Diagram

A data-flow diagram is a way of representing a flow of data through a process or a system. The DFD also provides information about the outputs and inputs of each entity and the process itself.

2.2.1 Level-0 DFD

It's designed to be an abstraction view, showing the system as a single process with its relationship to external entities. It represents the entire system as a single bubble with input and output data indicated by incoming/outgoing arrows.

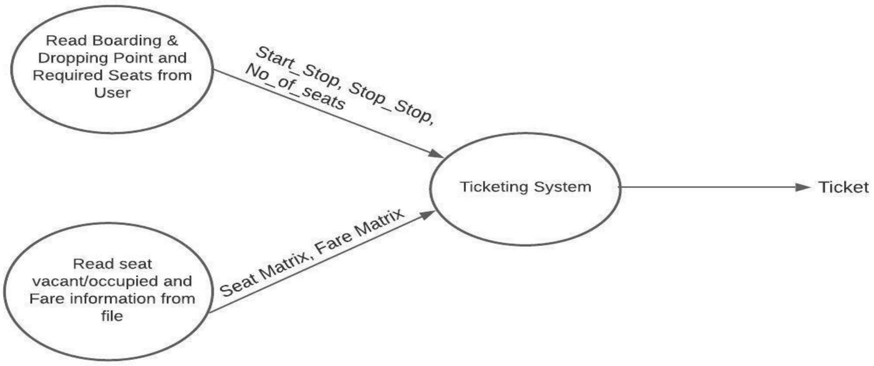
Read Boarding & Dropping Point and Required Seats from User
Start_Stop, Stop_Stop, No_of_seats
Ticketing System
Ticket
Read seat vacant/occupied and Fare information from file
Seat Matrix, Fare Matrix

Fig 2.2 Level-0 DFD

2.2.2 Level-1 DFD

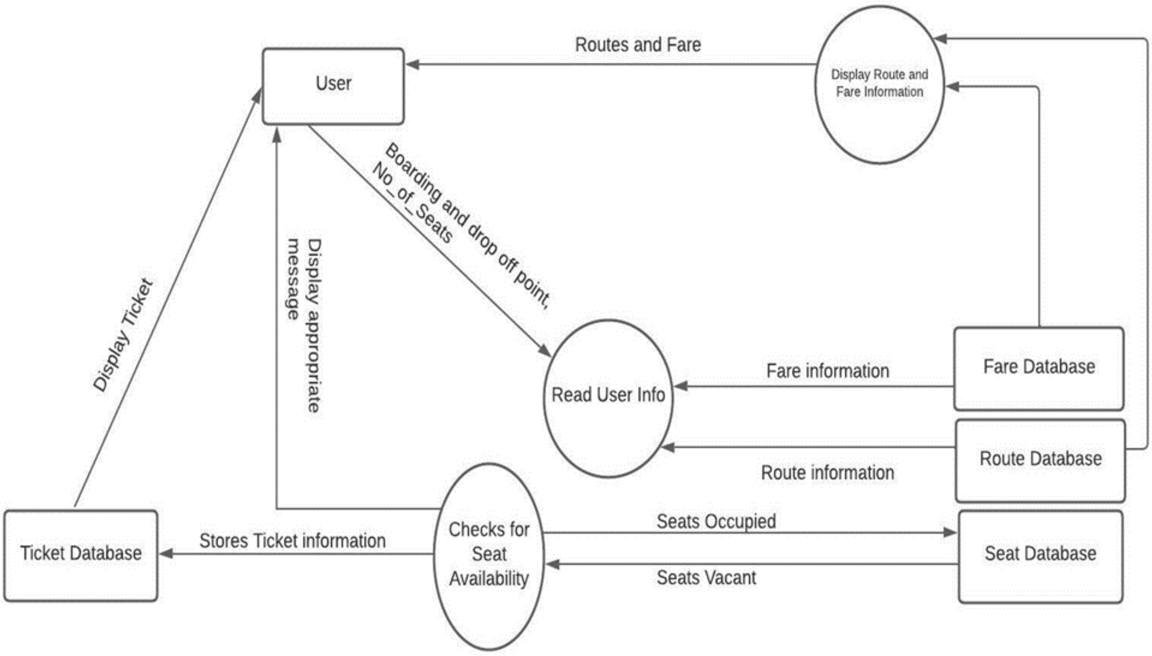

Fig 2.3 Level-1 DFD

2.3 User Interface Paradigm

- GUI: There is no GUI involved or created for the project.
- CLI: The application is wholly based on CLI, and the commands are given through it.
- The system follows a menu driven paradigm.

2.4 Design Flow

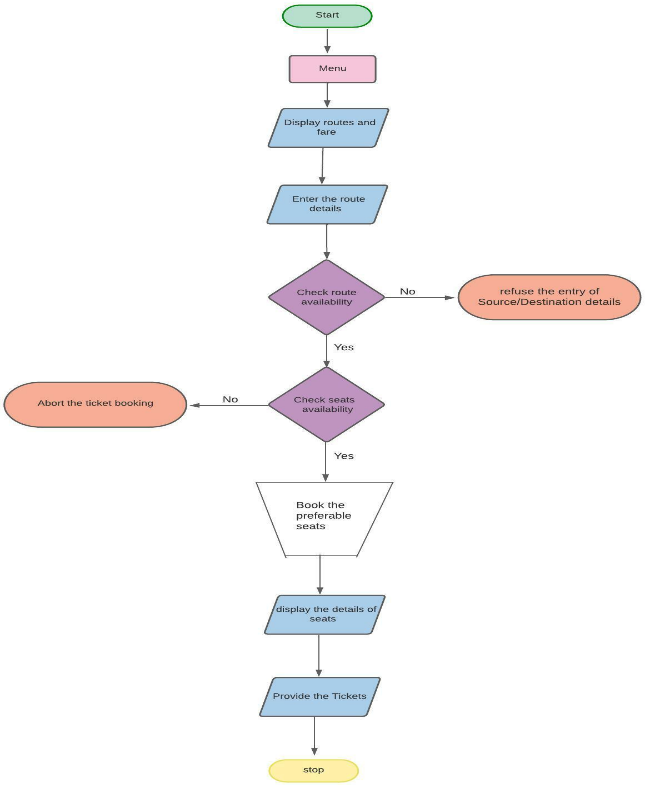

Fig 2.4 Design Flow chart

3. Detailed System Design

3.1 Functional Description

1. Storing route information: Using file handling in C, storing the route information with all the stops and distance from the start stop in a file and loading the same into the main memory.
2. Displaying the details of Route: Route information with all stops is displayed to the passenger or user using a file and reading from it.
3. Displaying price chart: The price chart is displayed to the user as a matrix by storing it in a file and reading from it.
4. Reading Start stop and Stop stop: Taking the input from the user about the user's boarding point, drop off point and then fetching the fare for the user's journey.
5. Displaying booked seats and vacant seats: A seat matrix with the available seats and booked seats will be displayed to the user.
6. Ticket Booking in an optimal way: Ticket booking is done in an improved and optimised way at intermediate stops as well the ticket will be stored in a file and can be viewed later.

3.2 HLD

High Level Design in short HLD is the general system design means it refers to the overall system design. It describes the overall description/architecture of the application. It converts the Business/client requirement into High Level Solution.

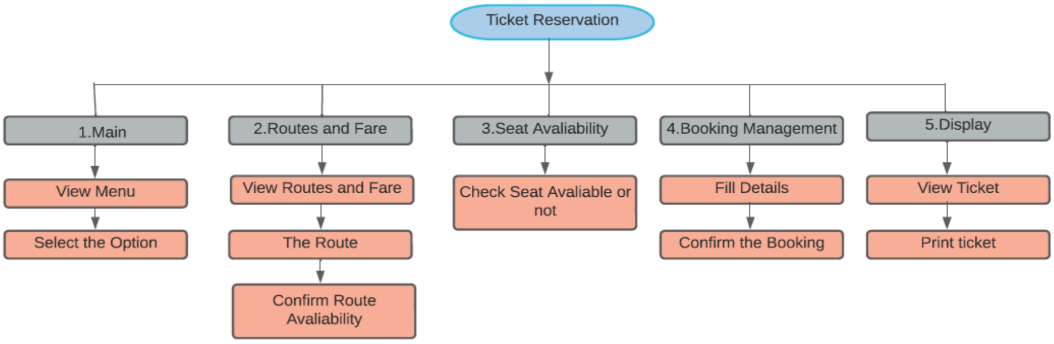

Fig 3.1 HLD

3.2 LLD

Low Level Design in short LLD is like detailing HLD means it refers to component-level design process. A detailed description of each and every module means it includes actual logic for every system component and it goes deep into each module's specification. It is also known as micro level/detailed design.

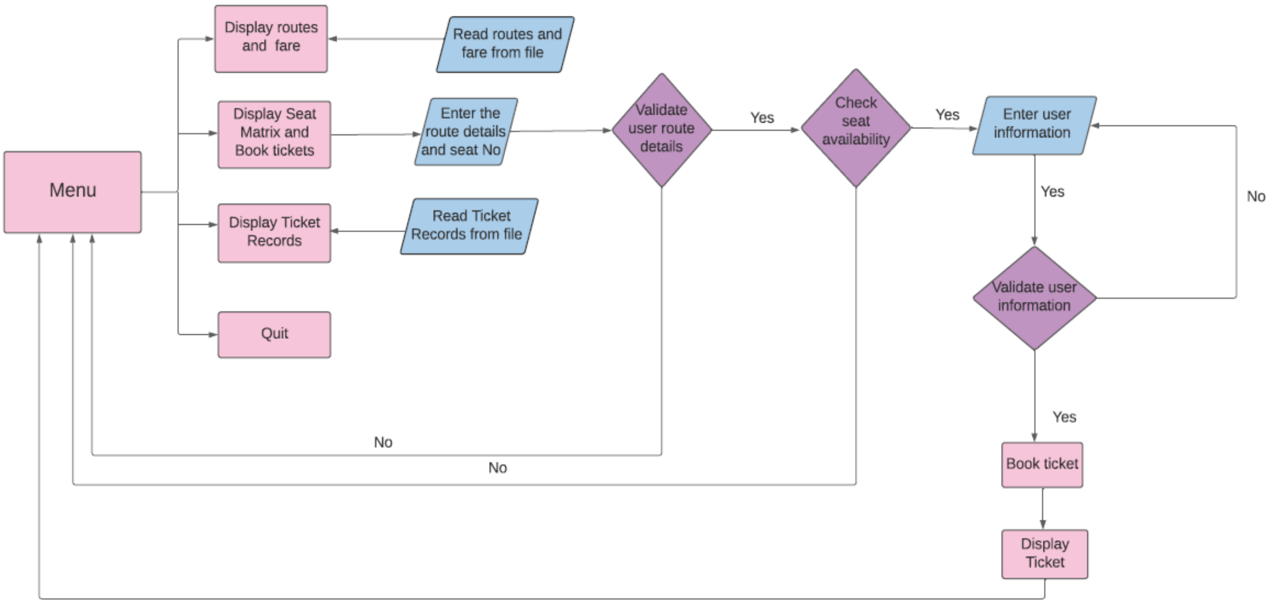
Display routes and fare
Read routes and fare from file
Display Seat Matrix and Book tickets
Enter the route details and seat No
Validate user route details
Yes
Check seat availability
Yes
Enter user information
Menu
Display Ticket Records
Read Ticket Records from file
Yes
No
Validate user information
Quit
Yes
No
No
Book ticket
Display Ticket

Fig 3.2 LLD

4. Environmental Description

4.1 Time Zone Support

The system supports all time zones.

4.2 Language Support

System supports only the English language.

4.3 User Desktop Requirements

Processor, RAM, Storage

4.3 Operating System Configuration

Windows and Linux OS

Ram	:4 GB
Storage	:256 GB SSD/HDD