SOFTWARE REQUIREMENTS SPECIFICATION

FOR

AIRLINE RESERVATION SYSTEM

Prepared by: -

1.Charan.R

2.Balaji.P

3.Arrthi.M

1.Introduction

1.1 Purpose

The main objective of this document is to illustrate the requirements of the project Airline Reservation System. The document gives a detailed description of both functional and non-functional requirements proposed by the client. The purpose of this project is to provide a friendly environment to maintain an easy way of airline reservation for customers.

1.2 Document Conventions

- Convention for Main title
  - Font face: Times New Roman
  - Font style: Bold
  - Font Size: 36
- Convention for Subtitle
  - Font face: Times New Roman
  - Font style: Bold
  - Font Size: 20
- Convention for Body
  - Font face: Times New Roman
  - Font Size: 18

1.3 Scope of Development Project

The scope of an airline reservation system typically includes features like flight search, booking, payment processing, and ticket management. It may also include functionalities like seat selection, flight schedule management, check-in procedures, boarding pass generation, and passenger information management. It can also integrate with other systems such as airport operations, customer service, and flight status updates.

1.4 Definitions, Acronyms and Abbreviations

- JAVA -> platform independence
- SQL-> Structured query Language
- ER-> Entity Relationship
- UML -> Unified Modeling Language
- IDE-> Integrated Development Environment
- SRS-> Software Requirement Specification

1.5 References:

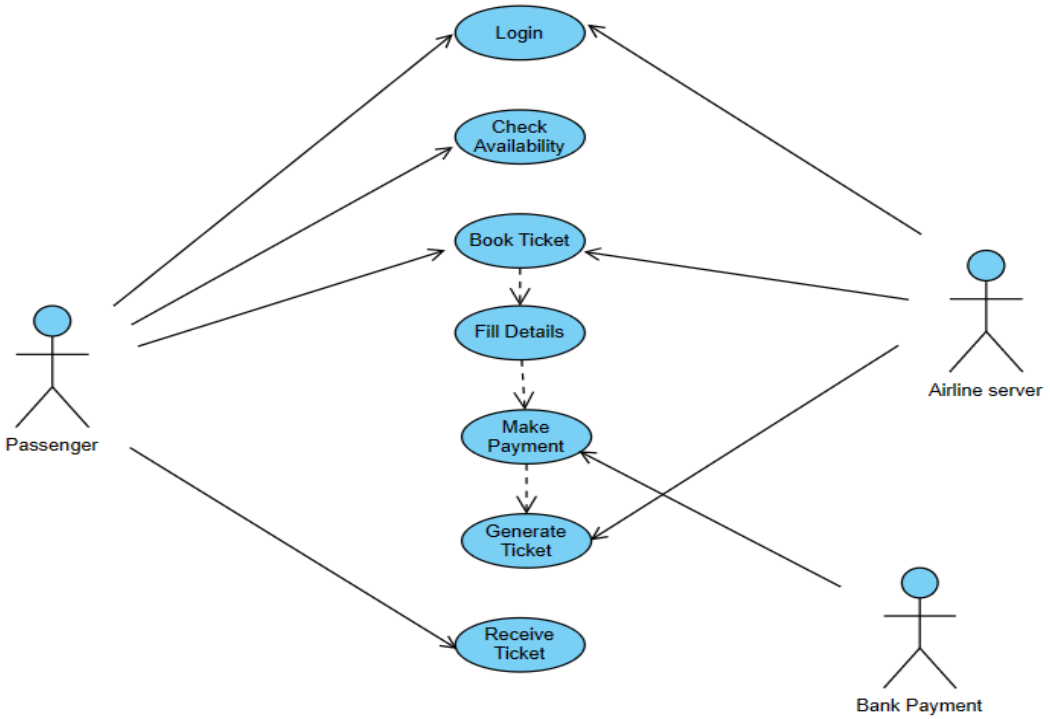
Websites:

- ER Diagram for Airline Reservation System - SourceCodeHero.com
- Airline Booking System ER Diagram | FreeProjectz

2. Overall Descriptions

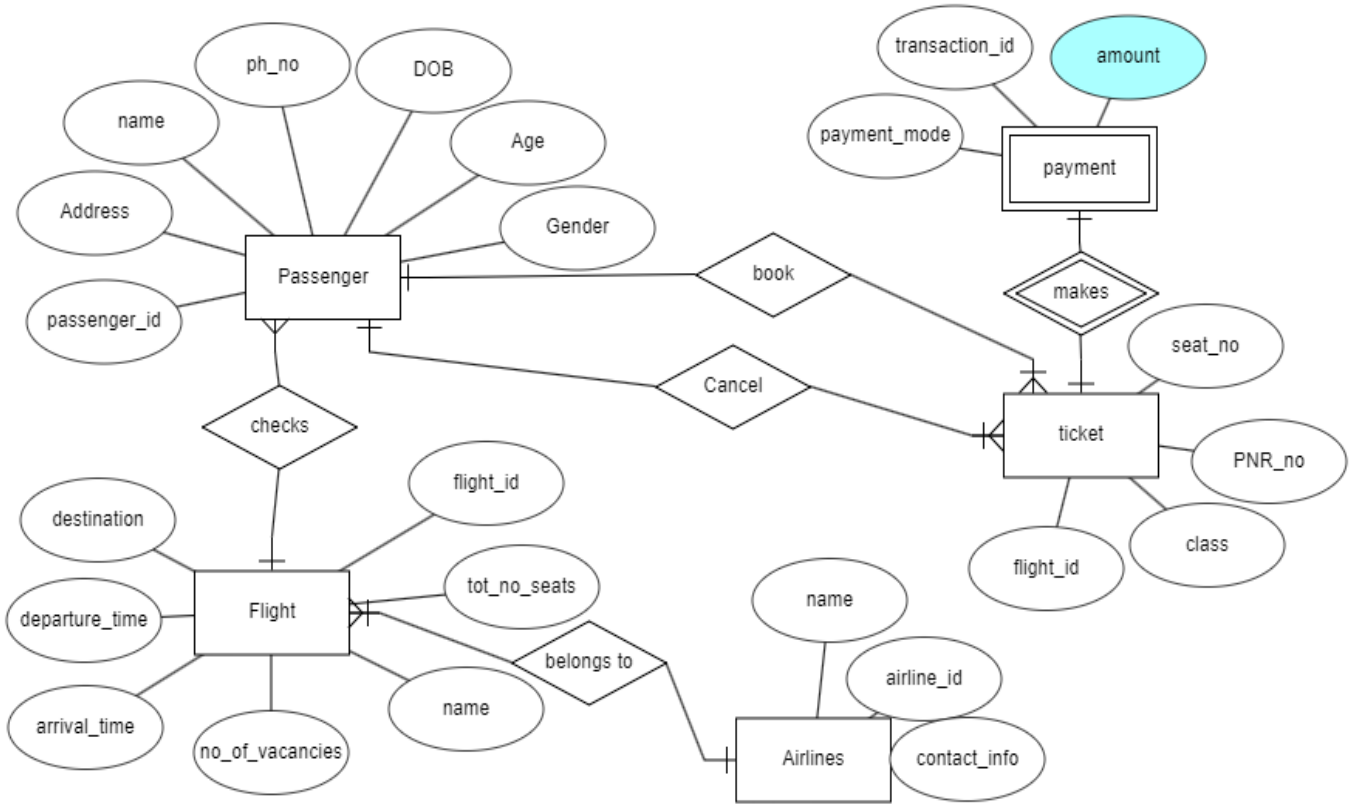
2.1 Product Perspective

Use case diagram for Airline Reservation System



2.2 Product Function

Entity Relationship Diagram of Airline Reservation System



2.3 User Classes and Characteristics

- The passenger can book one or many tickets
- The passenger can also cancel the tickets
- Many passengers can check for same flight
- Many flights belong to a Airline
- A single ticket takes a single payment

2.4 Operating Environment:

The product will be operating in a windows environment. The Airline Reservation System is a website and shall operate in all famous browsers, for a model we are taking Microsoft Internet Explorer, Google Chrome, and Mozilla Firefox. Also, it will be compatible with IE 6.0. Most of the features will be compatible with the Mozilla Firefox & Opera 7.0 or higher version. The only

requirement to use this online product would be the internet connection. The hardware configuration includes Hard Disk: 40 GB, Monitor: 15” Color monitor, Keyboard: 122 keys. The basic input devices required are keyboard, mouse and output devices are monitor, printer etc.

2.5 Assumptions and Dependencies:

The assumptions are: -

 The coding should be error free

 The system should be user-friendly so that it is easy to use for the users

 The information of all customers must be stored in a database that is accessible by the website

 The system should have more storage capacity and provide fast access to the database

 The system should provide search facility and support quick transactions

 The System should run for 24/7.

 Users may access from any computer that has Internet browsing capabilities and an Internet connection

 Users must have their correct usernames and passwords to enter their online accounts and do actions.

 The specific hardware and software due to which the product will be run

 On the basis of listing requirements and specification the project will be developed and run

 The end users (admin) should have proper understanding of the product

 The information of all the passengers must be stored in a database that is accessible by the Reservation System

2.6 Requirement Software Configuration:

This software package is developed using java as front end which is supported by sun micro system. Microsoft SQL Server as the back end to store the database. Operating System: Windows NT, windows 98, Windows XP Language: Java Runtime Environment, Net beans 7.0.1 (front end) Database: MS SQL Server (back end) Hardware Configuration: - Processor: Pentium(R)Dual-core CPU Hard Disk: 40GB RAM: 256 MB or more

2.7 Data Requirement:

The input consists of the query to the database and the output consists of the solutions for the query. The output also includes the user receiving the details of their accounts.

3. External Interface Requirement

3.1 GUI

The software provides a good graphical interface for the user and the administrator can operate on the system, performing the required tasks such as booking, canceling, viewing the details of the ticket.

 The user interface must be customizable by the administrator
 All the modules provided with the software must fit into this graphical user interface and accomplish to the standard defined
 The design should be simple, and all the different interfaces should follow a standard template

 The user interface should be able to interact with the user management module and a part of the interface must be dedicated to the login/logout module

Login Interface: -

In case the user is not yet registered, he can enter the details and register to create his account. Once his account is created, he can 'Login' which asks the user to type his username and password. If the user entered either his username or password incorrectly then an error message appears.

Search: -

The number of vacant seats available should be shown when a passenger searches for a ticket.

4. System Features

The users of the system should be provided with the assurance that their account is secure. This is possible by providing: -

 User authentication and validation of members using their unique passenger ID

 Proper accountability which includes not allowing a member to see other member's account. Only administrator will see and manage all member accounts

5. Other Non-functional Requirements

5.1 Performance Requirement

The proposed system that we are going to develop will be used as the Chief performance system for different airlines which interacts with the airline staff and passengers. Therefore, it is expected that the database would perform functionally all the requirements that are specified by the airlines.

 The performance of the system should be fast and accurate

The Airline Reservation System shall handle expected and non-expected errors in ways that prevent loss of information and long downtime periods. Thus, it should have inbuilt error testing to identify invalid username/password

 The system should be able to handle large amounts of data. Thus, it should accommodate high number of books and users without any fault

5.2 Safety Requirement

The database may crash at any certain time due to virus or operating system failure. Therefore, it is required to take the database backup so that the database is not lost. A proper UPS/inverter facility should be there in case of power supply failure.

5.3 Security Requirement

 System will use secured database

 Normal users can just read information, but they cannot edit or modify anything except their personal and some other information.

 System will have different types of users and every user has access constraints

 Proper user authentication should be provided

 No one should be able to hack users' password

 There should be separate accounts for admin and members such that no member can access the database and only admin has the right to update the database.

5.4 Requirement attributes

 There may be multiple admins creating the project, all of them will have the right to create changes to the system. But the members or other users cannot do changes

 The project should be open source

 The Quality of the database is maintained in such a way so that it can be very user friendly to all the users of the database

 The user be able to easily download and install the system

5.5 Business Rules

A business rule is anything that captures and implements business policies and practices. A rule can enforce business policy, decide, or infer new data from existing data. This includes the rules and regulations that the System users should abide by. This includes the cost of the project, and the discount

offers provided. The users should avoid illegal rules and protocols. Neither admin nor member should cross the rules and regulations.

5.6 User Requirement

The members are assumed to have basic knowledge of computers and internet browsing. The administrators of the system should have more knowledge of the internals of the system and is able to rectify the small problems that may arise due to disk crashes, power failures and other catastrophes to maintain the system. The proper user interface, user manual, online help and the guide to install and maintain the system must be sufficient to educate the users on how to use the system without any problems.

The admin provides certain facilities to the users in the form of:

 Backup and Recovery

 Forgot Password

 Data migration i.e. whenever user registers for the first time then the data is stored in the server

 Data replication i.e. if the data is lost in one branch, it is still stored with the server

 Auto Recovery i.e. frequently auto saving the information

 Maintaining files i.e. File Organization

 The server must be maintained regularly, and it must be updated from time to time

6. Other Requirements

6.1 Data and Category Requirements

There are different data and categories available for an airline reservation system. The data are information about the passenger, airline and flight's information, payment information etc. The categories include flight categories (Domestic or International), user categories (Passenger, Admins), seat categories (standard, premium), payment categories (cards, net banking, wallet). According to the category the relevant data should be displayed.

6.2 Appendix :

A: Admin, Abbreviation, Acronym, Assumptions; B: Books, Business rules; C: Class, Client, Conventions; D: Data requirement, Dependencies; G: GUI; K: Key; L: Library, Librarian; M: Member; N: Non-functional Requirement; O: Operating environment; P: Performance,Perspective,Purpose; R: Requirement, Requirement attributes; S: Safety, Scope, Security, System features; U: User, User class and characteristics, User requirement.

6.3 Glossary

The following are the list of conventions and acronyms used in this document and the project as well:

 Administrator: A login id representing a user with user administration privileges to the software

 User: A general login id assigned to most users

 Client: Intended users for the software

 SQL: Structured Query Language; used to retrieve information from a database

 SQL Server: A server used to store data in an organized format

 Layer: Represents a section of the project

 User Interface Layer: The section of the assignment referring to what the user interacts with directly

 Application Logic Layer: The section of the assignment referring to the Web Server. This is where all computations are completed

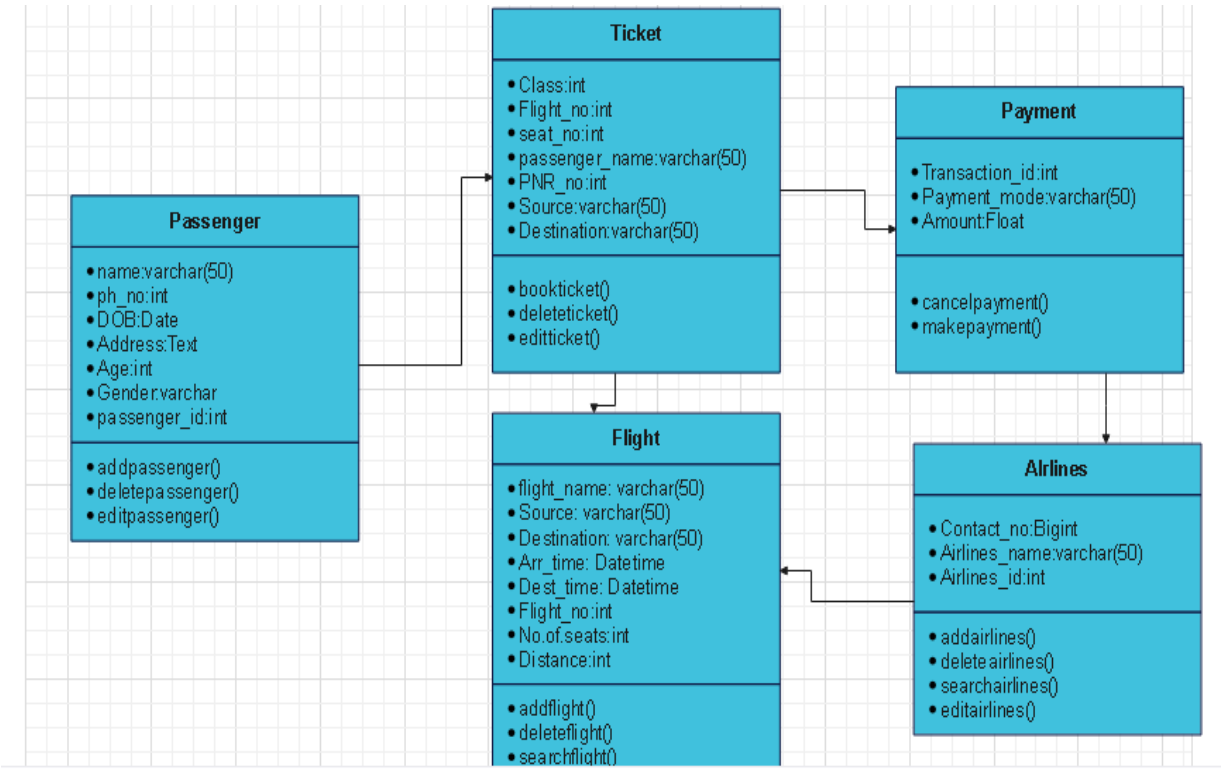
 Data Storage Layer: The section of the assignment referring to where all data is recorded

 Use Case: A broad level diagram of the project showing a basic overview

 Class diagram: It is a type of static structure diagram that describes the structure of a system by showing the system's cases, their attributes, and the relationships between the classes
 Interface: Something used to communicate across different mediums

 Unique Key: Used to differentiate entries in a database

6.4 Class Diagram


Ticket
•Class:int
•Flight_no:int
•seat_no:int
•passenger_name:varchar(50)
•PNR_no:int
•Source:varchar(50)
•Destination:varchar(50)
•bookticket()
•deleteticket()
•editticket()
Payment
•Transaction_id:int
•Payment_mode:varchar(50)
•Amount:Float
•cancelpayment()
•makepayment()
Passenger
•name:varchar(50)
•ph_no:int
•DOB:Date
•Address:Text
•Age:int
•Gender:varchar
•passenger_id:int
•addpassenger()
•deletepassenger()
•editpassenger()
Flight
•flight_name: varchar(50)
•Source: varchar(50)
•Destination: varchar(50)
•Arr_time: Datetime
•Dest_time: Datetime
•Flight_no:int
•No.of.seats:int
•Distance:int
•addflight()
•deleteflight()
•searchflight()
Airlines
•Contact_no:Bigint
•Airlines_name:varchar(50)
•Airlines_id:int
•addairlines()
•deleteairlines()
•searchairlines()
•editairlines()