IACSD

INSTITUTE FOR ADVANCED

COMPUTING AND

SOFTWARE DEVELOPEMENT

AKURDI, PUNE

Software Requirements Specification for

"FLIGHT MANAGEMENT SYSTEM"

e-DAC

NAME	ROLL NO
• Rahul Tambe	2196
• Sayyam Telang	2199

Center Coordinator

Mr. Prashant Karhale

Project Guide

Mr. Akshay Parab

INTRODUCTION

1.1 Purpose

This document lists the requirement specifications for an Airline Ticket Reservation System (ATRS). The document is subject the change as the project progresses. The given version of the document is the initial one. Further changes of the project will be recorded to the document.

Product Scope

User Interface, searching multiple destinations, Flight reservations, Reservation cancellation, Online payment, Displaying warning messages. Usage facet: Searching, Sorting of flights, Reservation of tickets, Managing flight details. IT facet: Database, Web-based software system, Performance maintenance. Development facet: Internal policy and culture of the airlines company should be taken under consideration.

Overall Description

Product Perspective

Flight Management System is the digitized version of the traditional manual reservation system at the airline office. Existing manual system requires every customer to come to the airline office in order to make a booking. Apart from the fact that not all customers have time to come to the office, existing system also causes long queues at the office. Some customers get bored from waiting in the queue and airline loses its potential customers.

In addition to that, a hard copy of the passport is required during the reservation at office. Customers who are not able to present their passports at the office for whatever reason cannot make reservations.

The new system aims to overcome the above-mentioned drawbacks of the existing system. It will allow users to make reservations according their needs from different parts of the world without leaving their places. Furthermore, it will reduce the workload of the employees who are responsible to make reservations at the office.

The system allows customers to check the availability of flights for specific dates and routes, get information about durations of available flights. It also allows customers to check the prices and the things that are included in the ticket and can book the ticket. Administrator can modify, also add new flights to the system.

Product Functions

Search for flights

Description: Using this function a customer is able to search for destination flights by choosing specific dates and destinations.

Specify passengers

Description: With the help of this function customer selects the number of passengers.

Book flights

Description: This function allows customer to book flights by choosing departure and the arrival airports.

Add new flights

Description: The function grants administrator the privilege of adding new flights to the system

All Passenger Details

By clicking on All passenger link (button) only the admin will be having the authority to view all the passenger details who has booked the Flights with them(website).

View Flights

By clicking on the view flights link(button) admin can view all the available flights which are ready to give service to the client/customer/End User.

Use-case Diagram

Description: In the use case diagram given below, we have displayed how our users interact with the system to accomplish their goals and responsibilities. Here in our diagram, we have 2 actors (Customer, Administrator), main users (Customer, Administrator). In every use case mentioned in the diagram, the actions of the users are described, and how these use cases are related to one another is represented by the help of arrows. In our Use case diagram, all functionalities of the system are displayed. Customer can search for flights based on dates and destination, specify the passengers, book flight, choose the ticket and proceed. Another actor, Administrator, is responsible for adding new flights, view all passenger details and view flights.

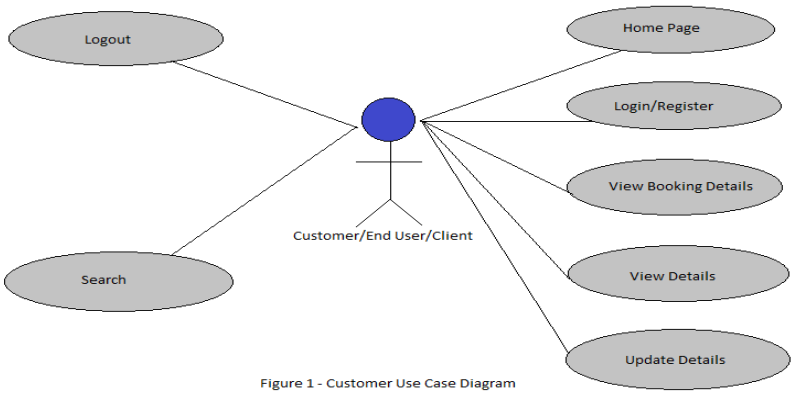


Figure 1 - Customer Use Case Diagram

Customer-use-case-diagram

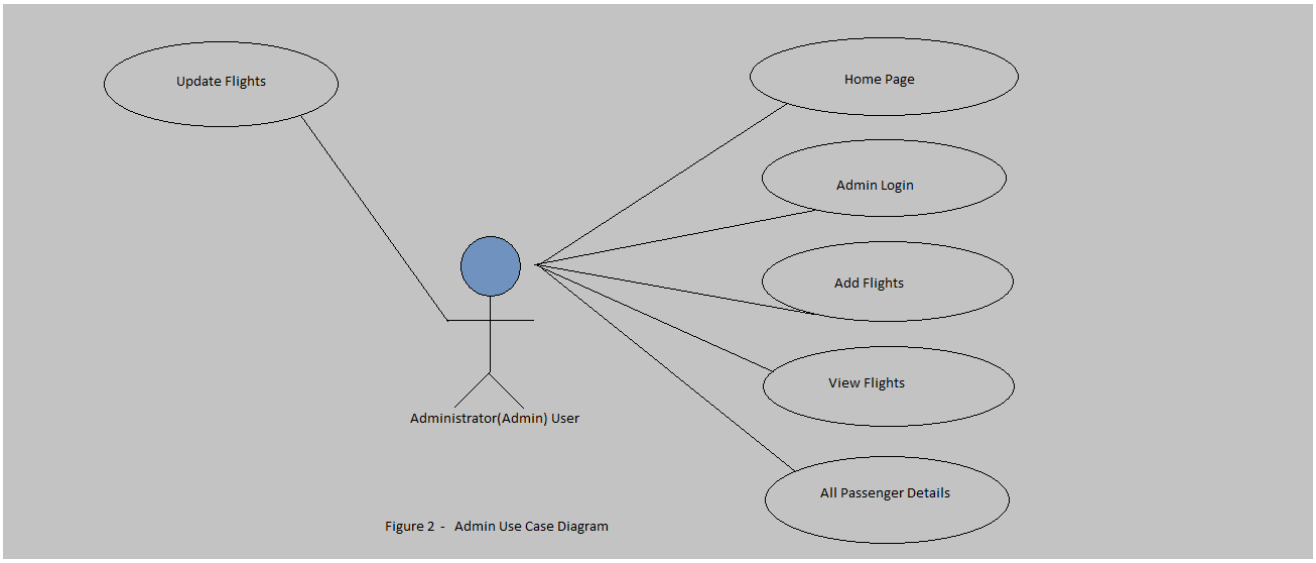


Figure 2 - Admin Use Case Diagram

Administrator(admin) use case diagram.

User Classes and Characteristics

The system users are divided into two categories: administrators, customers. Administrators should be trained and have a knowledge about using this application. On the other hand, customers do not need a training or a background knowledge.

User Interests

Customers' interests are getting information about available flights of specific dates and routes, checking the prices of the tickets. Moreover, customers concern about booking flights without leaving the places where they are. Administrator's interests include adding flight details, view flight details, view all passenger details.

Operating Environment

The designed system is thought to be a website and will be available via any web-browser application. It will not be dependent on the technical capabilities or operating system of user's device.

Design and Implementation Constraints

Flight Number and Id should be displayed according to the booking of the flight and along with user will also get the user details based on the user id and can also get the booking details (if any). Additionally, information about any changes that are made in the database should be displayed with no delay.

Assumptions and Dependencies

It is assumed that the user has an internet access and has a vague knowledge of web-application usage. The performance of Flight Management System depends on the quality and speed of the internet connection.

Class Diagram

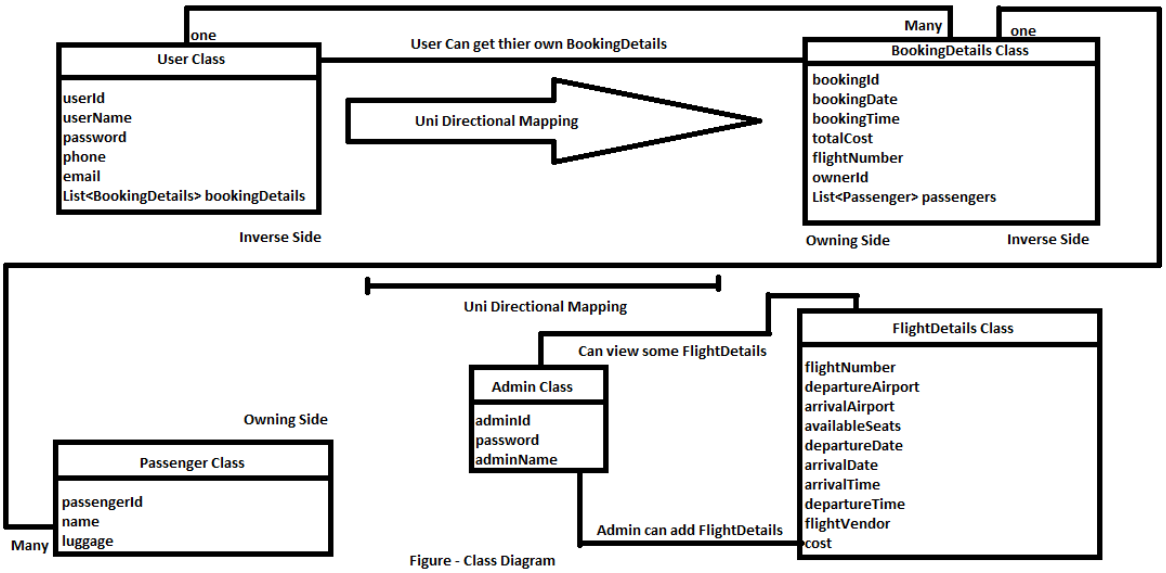


Figure - Class Diagram

User Class: User class will serve completely to the User whoever the User will be barring from admin User will be authenticated with the help of the User class. The Id and the Password which are in the homepage are verified using the User Class.

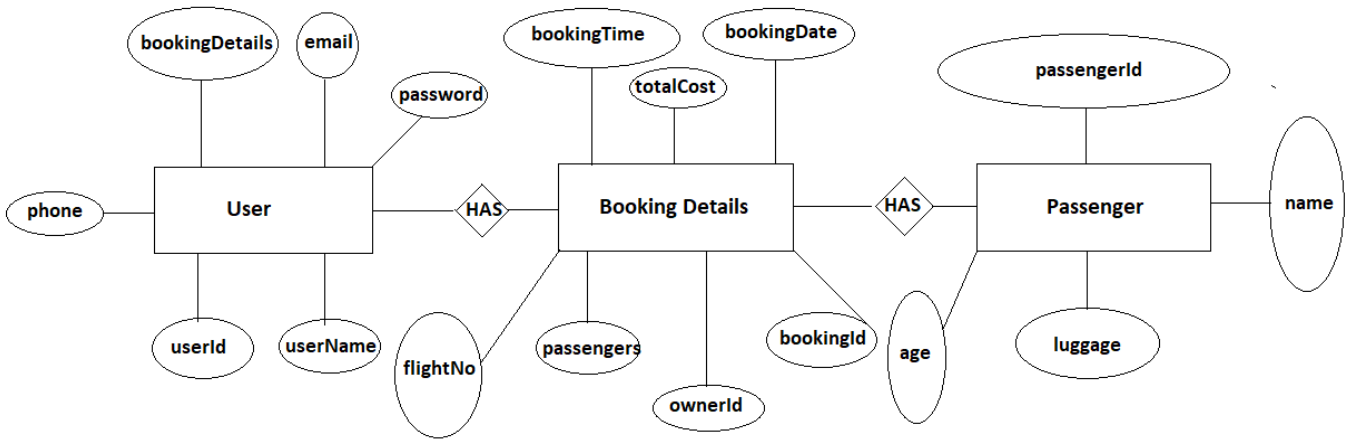
Admin Class: Same as to the User Class the admin user will be authenticated using the Admin Class and similarly as that of the User validation the Id and the Password for the Admin login will be verified using the Admin class.

Passenger Class: The passengers which will be added during the booking of the Flight ticket by the User will be visible to the admin is because of the Passenger Class.

BookingDetails Class: The authenticated user after log-in into the website will be able to get all the BookingDetails is because of the BookingDetails class.

FlightDetails Class: Only the admin user after log-in to the website will be able to fill the flight details and only admin will be able to use the fields of the FlightDetails Class. After adding the flight details the admin is also able to view some of the fields of the FlightDetails class after clicking on the view flight details button.

Entity-Relationship-Diagram (ER diagram)



Entity-Relationship-Diagram (ER)

The above fig represents the entity-relation in our project

In this we are having the User,BookingDetails and Passenger classes who are having the relationships between each other

User class is having bookingDetails field into it which specifies the uni-directional relationship between the two classes.

BookingDetails Class is also having the passengers field into which specifies the uni-directional relationship between the passengers ang the booking details class.

Table Structure

Users Table

Field	Type	Null	Key	Default	Extra
user_id	int	NO	PRI	NULL	
email	varchar(255)	YES		NULL	
password	varchar(255)	NO		NULL	
phone	bigint	NO		NULL	
user_name	varchar(255)	NO		NULL	

Admins Table

Field	Type	Null	Key	Default	Extra
admin_id	int	NO	PRI	NULL	
admin_name	varchar(255)	YES		NULL	
password	varchar(255)	YES		NULL	

Passengers Table

Field	Type	Null	Key	Default	Extra
passenger_id	int	NO	PRI	NULL	
age	int	YES		NULL	
luggage	double	YES		NULL	
name	varchar(255)	YES		NULL	

BookingDetails Table

Field	Type	Null	Key	Default	Extra
booking_id	int	NO	PRI	NULL	
booking_date	varchar(255)	YES		NULL	
booking_time	varchar(255)	YES		NULL	
flight_number	int	YES		NULL	
owner_id	int	YES		NULL	
total_cost	double	YES		NULL	
user_id	int	YES	MUL	NULL	

FlightDetails Table

Field	Type	Null	Key	Default	Extra
flight_number	int	NO	PRI	NULL	
arrival_airport	varchar(255)	NO		NULL	
arrival_date	varchar(255)	NO		NULL	
arrival_time	varchar(255)	NO		NULL	
available_seats	int	YES		NULL	
cost	double	NO		NULL	
departure_airport	varchar(255)	NO		NULL	
departure_date	varchar(255)	NO		NULL	
departure_time	varchar(255)	NO		NULL	
flight_vendor	varchar(255)	NO		NULL	

Functional Requirements

- Customer should be able to login with the specified id and password
- If the customer is not registered then he must register first with the application.
- After the Rsgistration the User will get the Id and the password which they have entered during the registration.
- Only the registered user can book the flights
- The registered user can only be able to view the booking details (if any)
- The registered user will be able to get all their details by clicking on the view details button
- If the registered user feels that they need to update any of their details then they can do that just by clicking on the update details button.
- The user can search for the flights by clicking on the search button any if any flight is available with their matchings, they can see the details of the flight

Non-Functional-Requirements

Performance Requirements

- System should grant administrative privileges only to the one who logins with predefined administrative username and password.
- Customized password should be at least 8 characters.
- System should allow administrator to login with customized password.