

.NET CORE

Flight Booking System

Document Revision History

Date	Revision No.	Author	Summary of Changes

Table of Contents

1. Introduction.....	4
2. Objective(s).....	5
3. Use-Case Diagram.....	6
4. Design Pattern.....	7
5. Solution Diagram.....	8
6. ER Diagram.....	9
7. Database-Schema.....	10
8. Class Diagram.....	12
9. Flowchart.....	13
10. Project Structure.....	14

1.0 INTRODUCTION

This document outlines a project for the .NET Core . The project is to develop Web API for Flight Booking System. This document contains the requirements, workflow of the system and gives guidelines on how to build the functionality gradually in each of the course modules of the .NET core.

1.1 DOCUMENT PURPOSE

Purpose of the Flight booking sys is to eliminate the many difficulties/problems that occur to the people in the Flight booking sys and as well as make it easy for the customers that save the flights and they will be able to utilize the project to make their reservations modify reserve seat and cancel that reserved seat

1.2 Scope

Airline Reservation System is one of the modifications that were carried out in the Passenger Service System so that the working and availability of Service area can be broadened. On one hand, it helps the customers and on the other, it also makes the life of the airline service companies easier by keeping all the records of the passengers and if there is any change in the flight due to some reason, the passengers are promptly informed. This system is also used by companies to keep track of user preferences of regular travelers so that they can provide better service and give offers to customers

1.3 SETUP CHECKLIST

Minimum System Requirements

- Intel core vpro i5 4 and Windows 2010
- Internet Explorer 11.0 or higher / Chrome
- SQL Server 2019
- Visual Studio 2019
- Angular 5.

2.0 PROBLEM STATEMENT

2.1 OBJECTIVE

In this application we will create a Flight Booking System web Application with the dotnet core web API backend and Angular frontend. The entities that we will be working with are users, flights, Booking, notification. These entities will have their own access limits in the web app.

2.2. Assumptions

It is assumed that the details of the flight are already known to the customer. Future changes like providing different types of flights with different classes like business class, economic class will allow the customers to benefit from one facility.

2.3 Requirement

Entities-Users, Flights, Booking, Payment, Notification serve

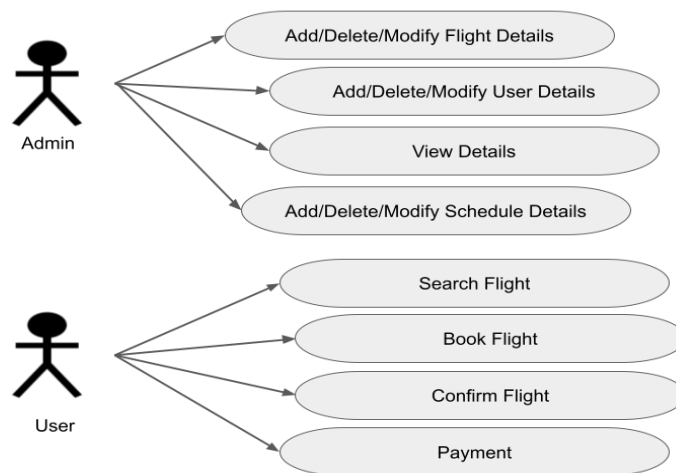
1. Functional requirements
2. Non-Functional requirements

2.3 Process of the project:

1. User should be able to search flights based on arrival + destination
+date of travel
2. Users should be able to login to the application.
3. Select the flight based on time/preference
4. Enter personal details
5. Make payment in your application.
6. Confirmation mail /notification.

3.0 Use Case Diagram

Actors of the System will be: Admin, User



3.1 Role of Actor:

For Admin

1. Admin will verify the entered credential.
2. Delete, Modify, Add details of user, flight and schedule.

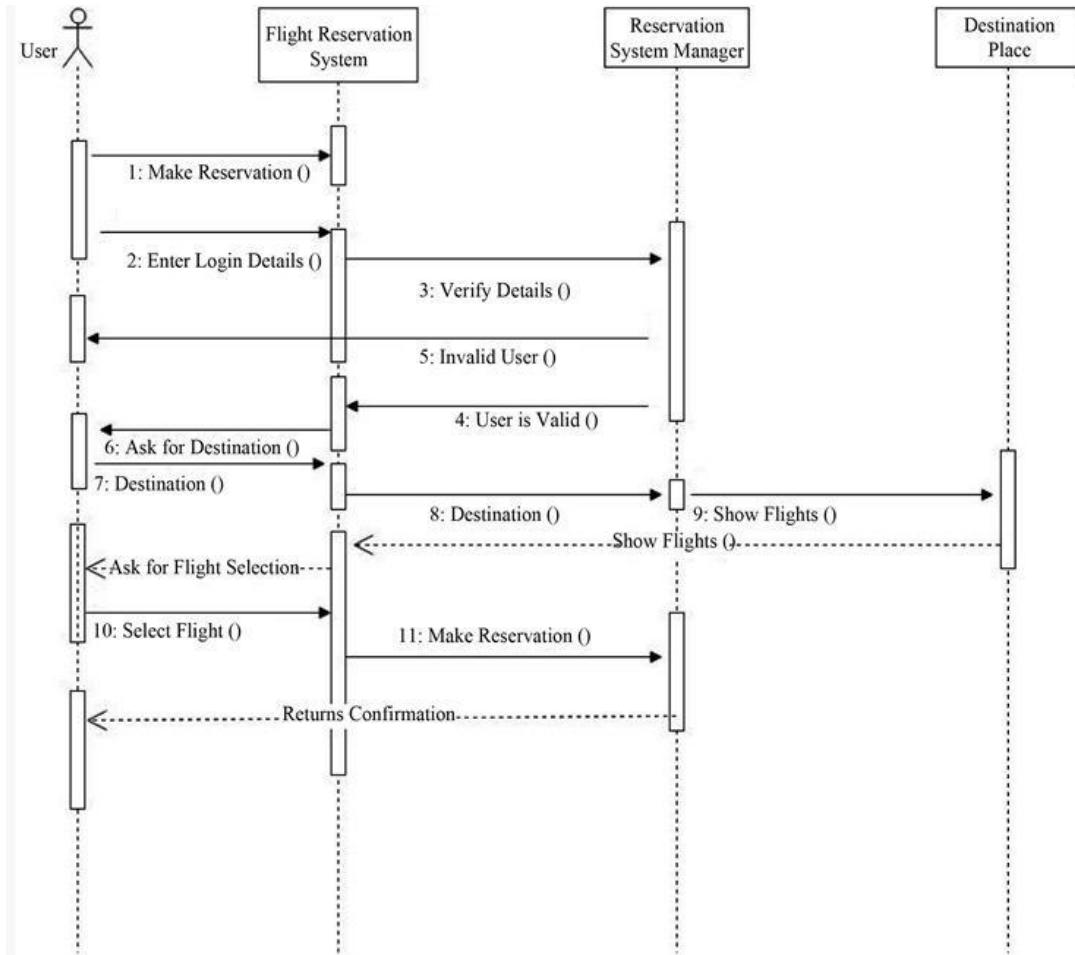
For user:

1. Search flight
2. Book Flight
3. confirm flight

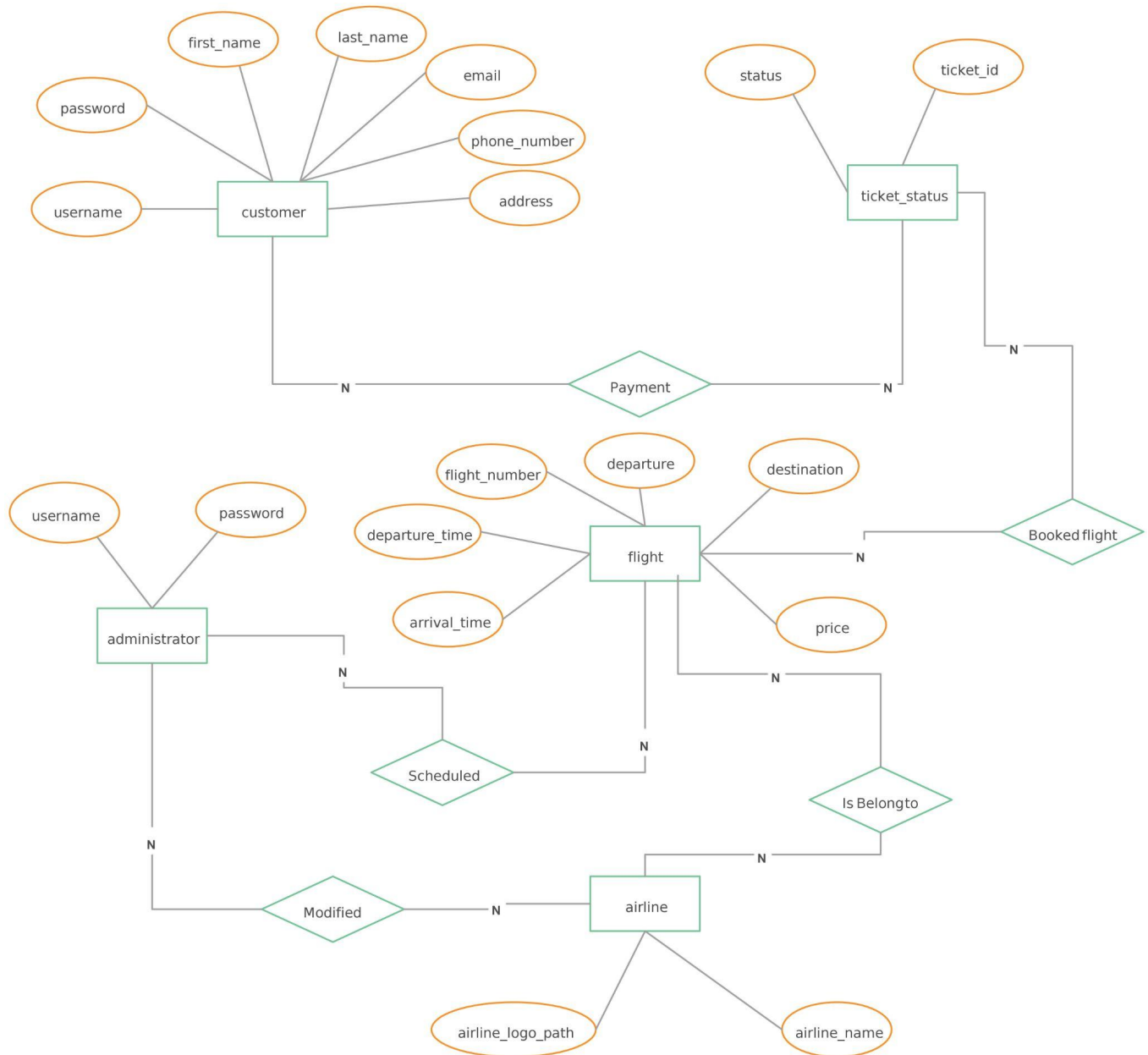
4.0 Design Pattern:

#	Name	Description
1	API	Using HTTP Request, We Will use the respective action to trigger various operations

5.0 Solution Diagram :



6.0 ER DIAGRAM



7.0 Database Schema:

Flight Details Table:

FBS_Flight_Master

Field	Type	Length
FBS_Flight_Id(pk)	Int	20
FBS_Flight_Origin(fk)	Varchar	50
FBS_Flight_Destination(fk)	Varchar	50
FBS_Flight_Date	date	50
FBS_Flight_Fare	Int	20

FBS_Booking_Transaction

Field	Type	Length
FBS_Booking_Id(pk)	Int	20
FBS_Booking_Date	date	20
FBS_Flight_Id(fk)	Int	20
FBS_Flight_Fare	Int	50
Cust_FirstName	Varchar	50
Cust_LastName	Varchar	50
Cust_EmailId	Varchar	50
Cust_ContactNo	Int	20
FBS_Booking_Status	date	50
Created_By	Varchar	50
Modified_By	Varchar	50
Created_Date	date	50
Modified_Date	date	50

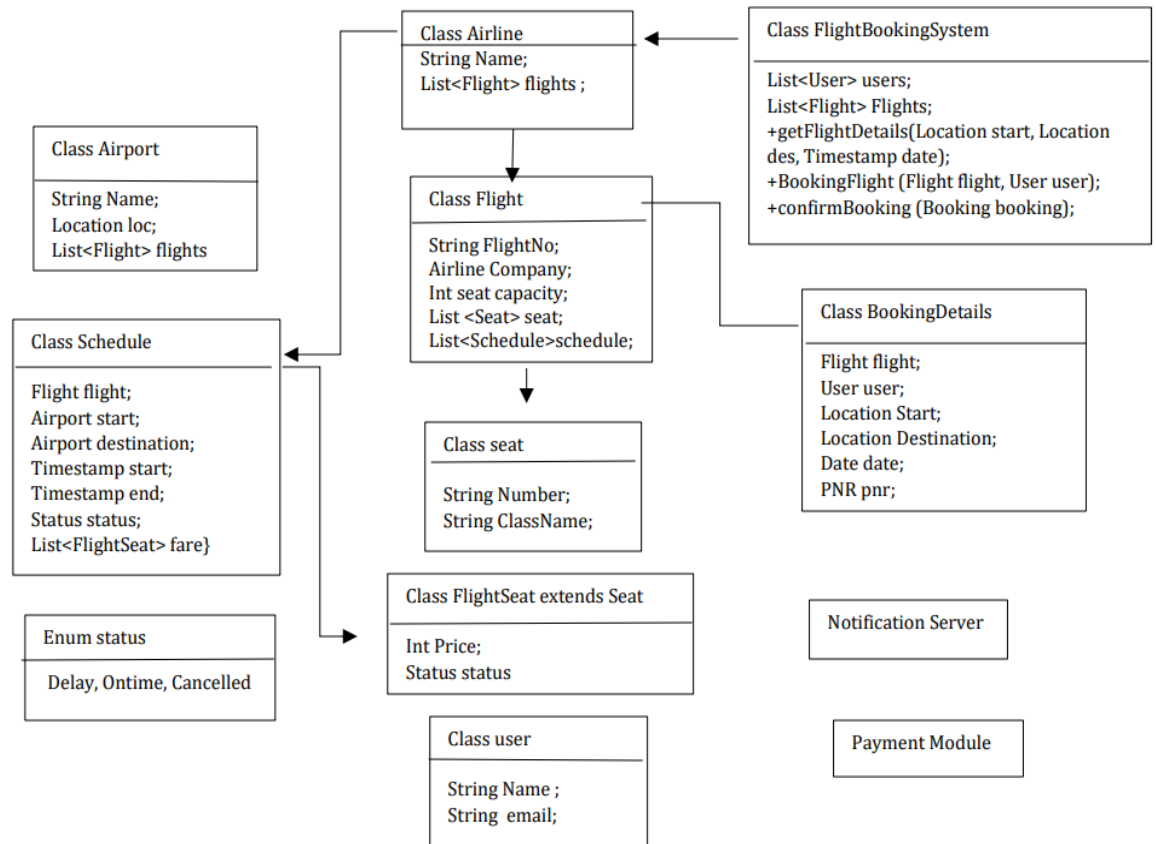
FBS_Flight_Status_DB

Field	Type	Length
FBS_Status	Varchar	20
FBS_Status_Id(pk)	Int	20
FBS_Status_Name	Varchar	50
Created_By	Varchar	50
Modified_By	Varchar	50
Created_Date	date	50
Modified_Date	date	50

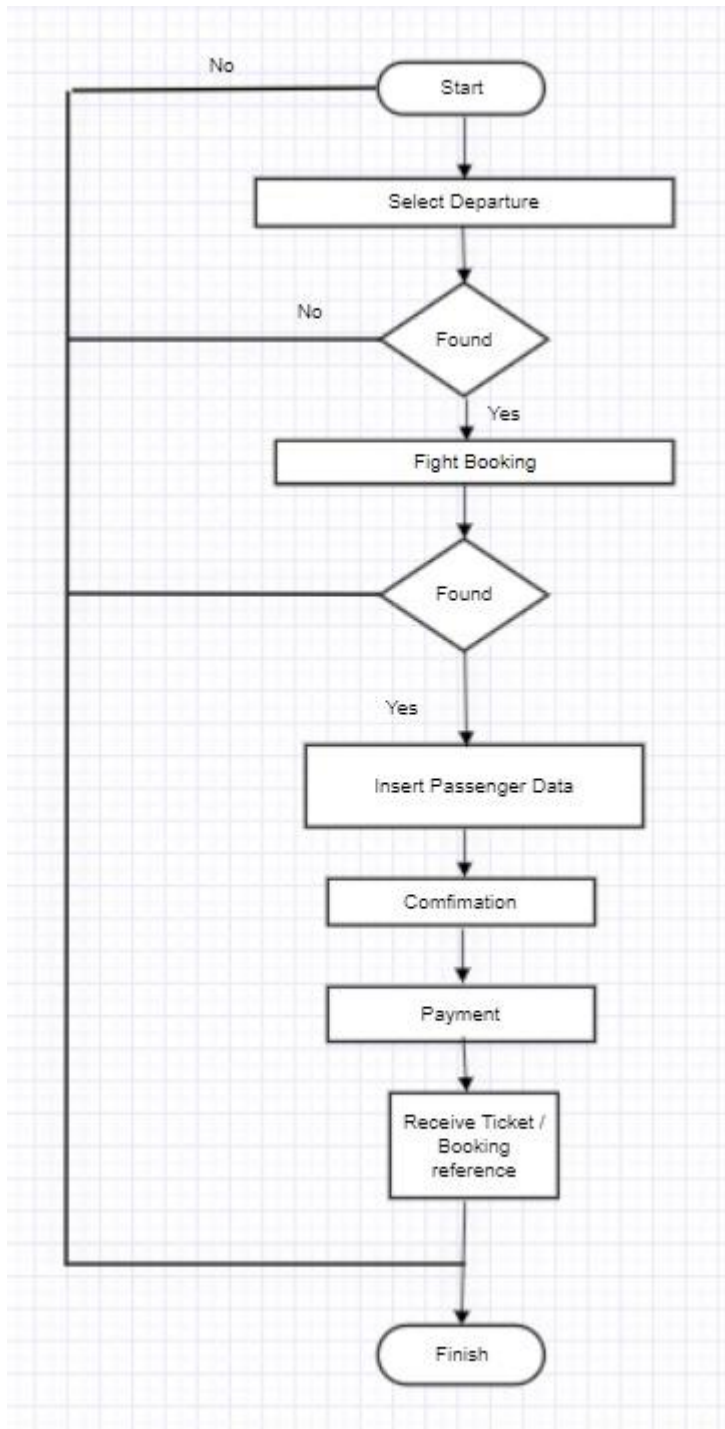
FBS_Flight_ServiceLocations_DB

Field	Type	Length
FBS_ServiceLocation_Id(uk)	Int	20
FBS_LocationName(pk)	Varchar	50
Created_By	Varchar	50
Modified_By	Varchar	50
Created_Date	date	50
Modified_Date	date	50

8.0 CLASS DIAGRAM



9.0 Flowchart



10.0 PROJECT STRUCTURE

10.1 API EXPECTATION

Create API using

- Visual Studio 2019 IDE
- .NetCore / C# 5.0
- SQL Express 2019
- Entity Framework
- Swagger Endpoint for Testing

