King Saud University

College of Computer and Information Sciences

Department of Software Engineering

SWE 444 Project

Book Flights

2nd Phase

Instructor: Prof. Mohammad Abdullah-Al-Wadud

|  |  |  |
| --- | --- | --- |
| # | Name | ID |
| 1 | Hamad Aloqaily | 436105244 |
| 2 | Abdullah Aljamhour | 437106288 |
| 3 | Ibrahim Alsuhaim | 437100964 |

Table of Contents

[1.Introduction 2](#_Toc21344612)

[2.Requirements 3](#_Toc21344613)

[2.1.Functional requirements 3](#_Toc21344614)

[3.Use Case Model 4](#_Toc21344615)

[4.Design and Architecture 5](#_Toc21344616)

[4.1.Architecture Design 5](#_Toc21344617)

[4.2.Detailed Design 6](#_Toc21344618)

[Class Diagram 6](#_Toc21344619)

[Database design 7](#_Toc21344620)

[5.The plan 8](#_Toc21344621)

[6.List of scenarios implemented in Phase 2 9](#_Toc21344622)

[7.Contribution of each member 9](#_Toc21344623)

[8.URL 10](#_Toc21344624)

# Introduction

In recent years, many people travel to other countries for several reasons, including treatment, study, tourism and so on. Therefore, our project aims to serve these people through the Internet service without the need to go to the ticketing centers to book a flight ticket. In addition to that our website offers many services for the traveler for instance, the user can choose the appropriate trip, time and choose the features of that trip for example he can choose the meal and the seat in the plane. Also, the user can cancel his/her flight ticket before 24 hours of departure time.

# Requirements

## Functional requirements

1. The system shall let the user register.
2. The system shall let the user and admin login.
3. The system shall let the user and admin logout.
4. The system shall let the admin add flights.
5. The system shall let the admin delete flights.
6. The system shall let the user search for tickets.
7. The system shall let the user choose sorting tickets by price.
8. The system shall let the user filter search result by airline.
9. The system shall let the user choose number of passengers.
10. The system shall let the user book a ticket.
11. The system shall let the user choose specific seat.
12. The system shall let the user choose meal.
13. The system shall show to the user the information about the booked flight (distance, time, class, flight number).
14. The system shall let the user show his/her booked tickets.
15. The system shall let the user cancel his/her ticket before 24 hours from the departure time.
16. The system shall send confirmation email the user after booking.
17. The system shall send reminder email users before 24 hours from flight departure time.
18. The system shall show the distance between the two locations.

# Use Case Model

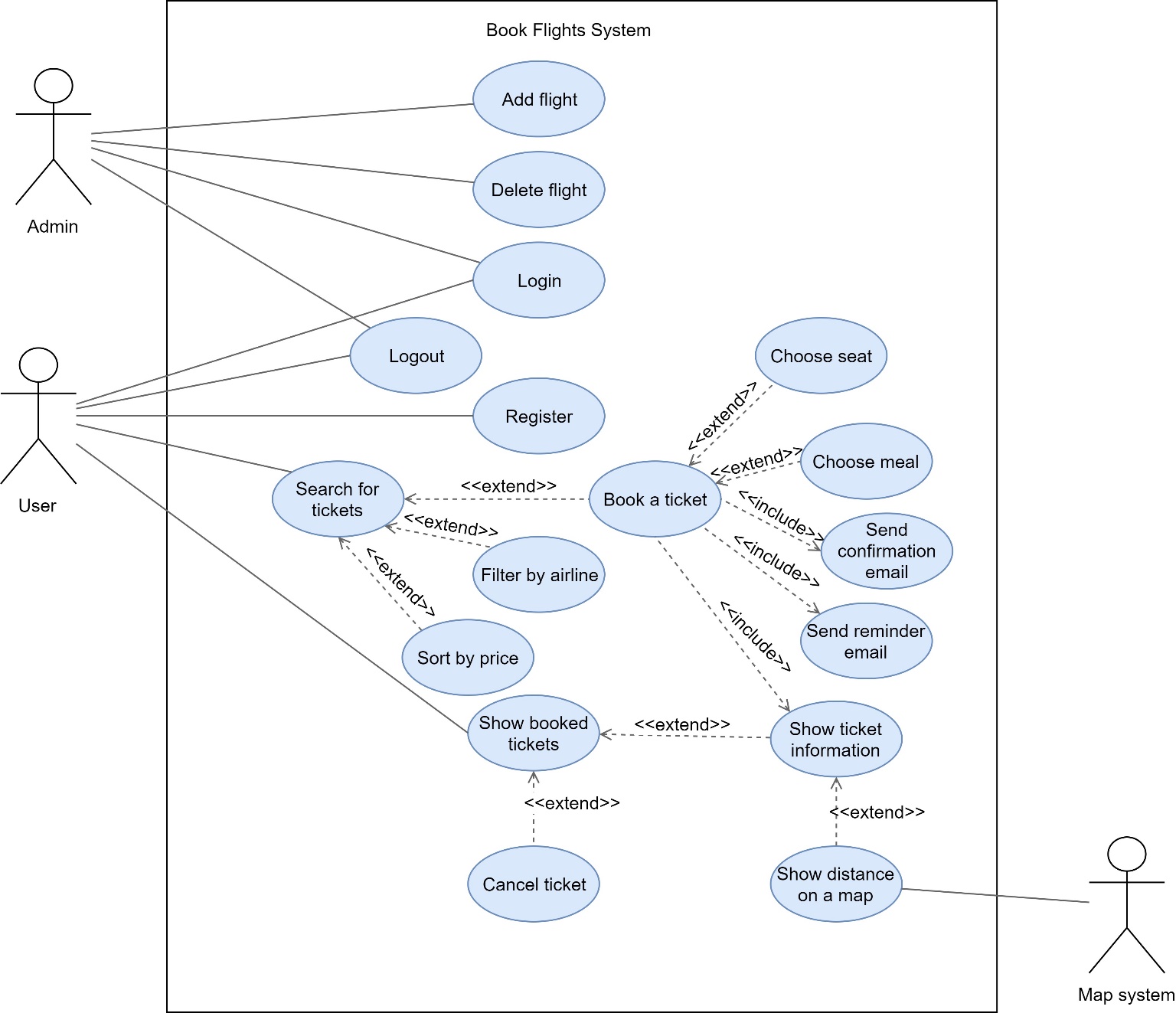


Figure 1 Book Flights System Use Case diagram

# Design and Architecture

## Architecture Design

Client-Server

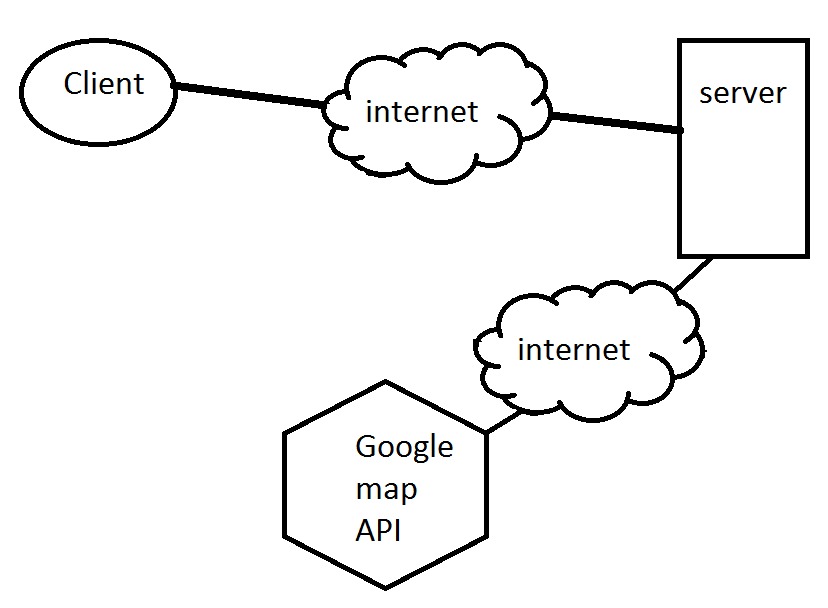


Figure Client - Server architecure for book flights system

We used client-server architecture because our system is centralized, also it helps us with the scalability of the system to make it easier in the future to expand our system to support multiple users by using multiple servers.

The alternative architecture is layered architecture it has many advantages that will help our system like simplicity and security, but we didn't choose it because it will affect the performance and it's very hard to scale up.

## Detailed Design

## Class Diagram

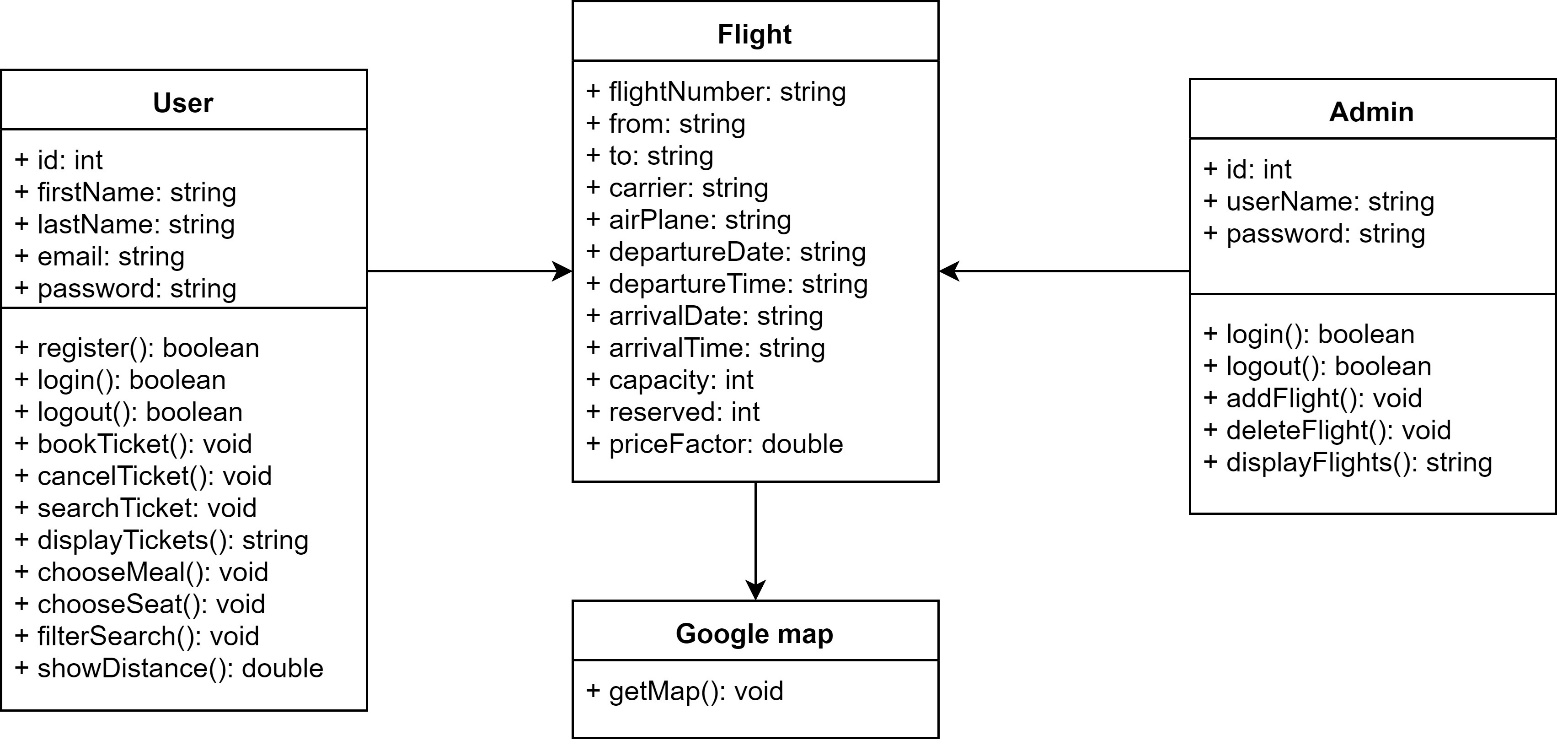


Figure 3 book flights class diagram

## Database design

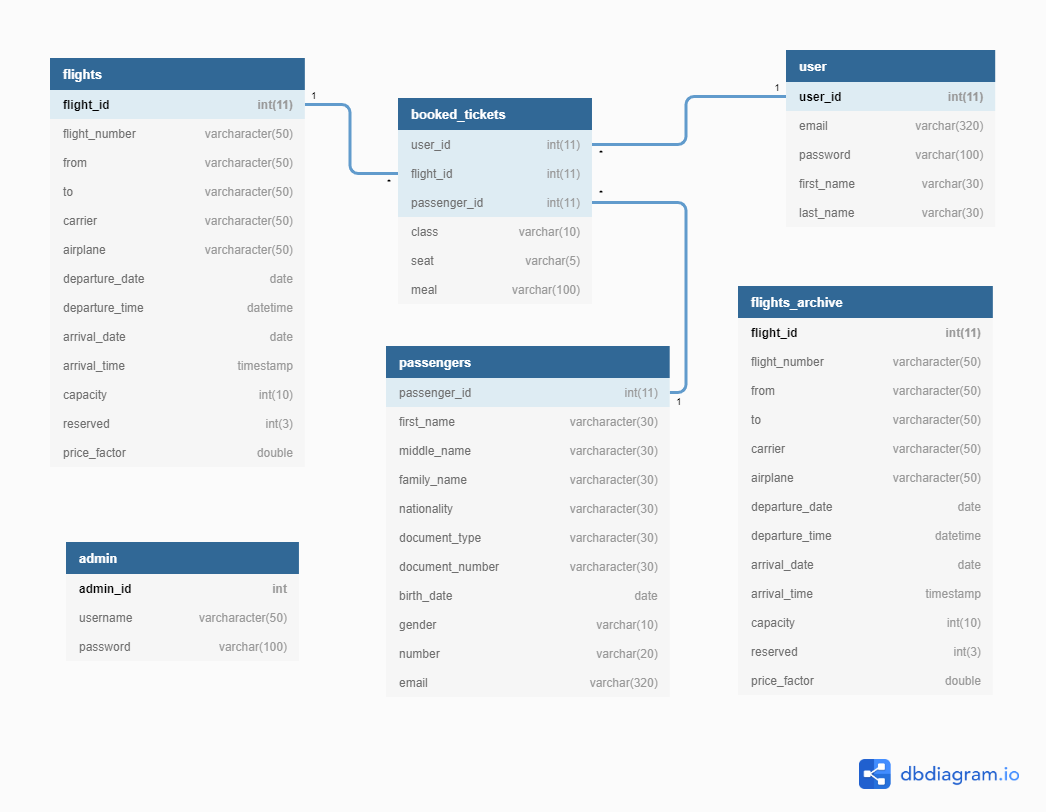


Figure Database design for book flights system

# The plan

|  |  |
| --- | --- |
| Missions | **Phase** |
| 1. Register 2. Login 3. Logout | **Phase 1** |
| 1. Admin add flights 2. Admin delete flights 3. Search for tickets | **Phase 2** |
| 1. Book a ticket 2. Show booked tickets. 3. Send confirmation email. | **Phase 3** |
| 1. Choose meal and seat features. 2. Show ticket information. 3. Cancel booked ticket | **Phase 4** |
| 1. Filter by airline and Sort by price 2. Show distance on map 3. Send reminder email. | **Phase 5** |

Figure 5 The plan table for Book Flights System

# List of scenarios implemented in Phase 2

|  |  |
| --- | --- |
| Use case | Scenario |
| Add flight | 1. Admin login to the system using username and password. 2. dashboard is displayed with add flight form. 3. Admin enters the required flight fields. 4. The flight is added to the database. |
| Delete Flight | 1. Admin login to the system using username and password. 2. dashboard is displayed with delete flight form. 3. Admin enters the required flight id to be deleted. 4. Confirmation is altered. 5. Admin confirm deletion. 6. The flight is deleted from the database. |
| Search for tickets | 1. User in the index page. 2. User clicks Book A trip button. 3. Search form is displayed. 4. User fills required fields. Ex: from (RUH), to (JED), dep: 2019-12-10, return: 2019-12-17. 5. Available flights are displayed. Otherwise no flight message is displayed. |

# Contribution of each member

|  |  |
| --- | --- |
| **Name** | **Task** |
| Hamad Aloqaily | Add flight |
| Abdullah Aljamhour | Delete flight |
| Ibrahim Alsuhaim | Search for tickets |

# URL

<https://book-flights.herokuapp.com/>

**Note:** Admin can access login page via this URL <https://book-flights.herokuapp.com/admin.php>

Admin username: Admin

Password: 123123123

**Source code:** <https://github.com/IbrahimAlSuhaim/Booking-Flights>