King Saud University

College of Computer and Information Sciences

Department of Software Engineering

SWE 444 Project

Book Flights

3rd 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 (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.
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.

# Use Case Model



Figure 1 Book Flights System Use Case diagram

# Design and Architecture

## Architecture Design

Client-Server

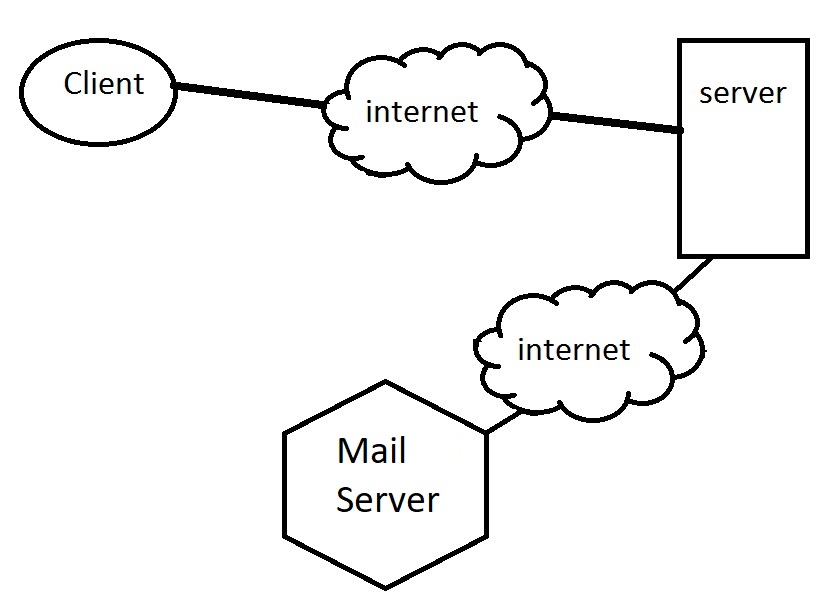


Figure 2 Client - Server architecure for book flights system

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

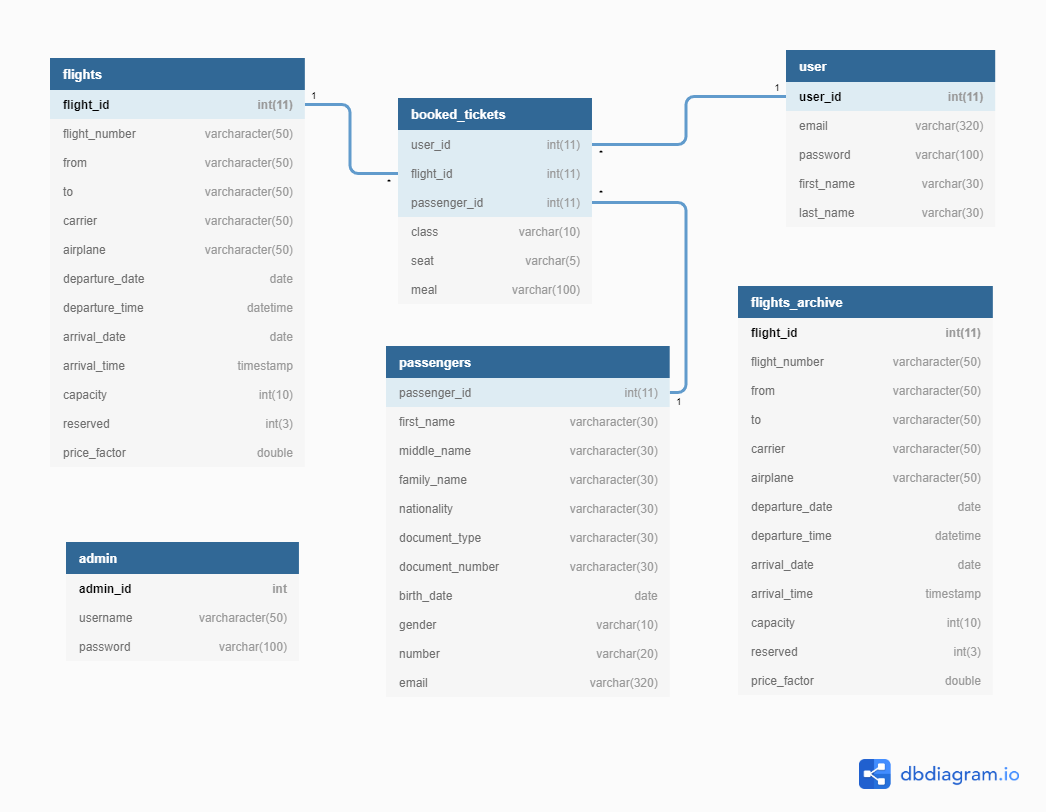


Figure 4 Database design for book flights system

# 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 ticket(s) 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. 2. Sort by price. 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. |

# List of scenarios implemented in Phase 3

|  |  |
| --- | --- |
| Use case | Scenario |
| book ticket(s) | 1. Steps mentioned in use case Search for tickets. 2. After that, the user selects the departure flight, and if the directionality is return, the available return flights are displayed, and the user selects one. 3. Form of passengers is displayed, the user fills required fields and clicks GO! 4. The user will be directed to seat page, which is not implemented in this phase yet (in next phase). 5. The user clicks on go to payment. *The payment process isn’t required in this project so the user will be directed to confirmation page directly.* 6. Confirmation page is displayed. |
| send confirmation email | * After book ticket(s) use case a confirmation email should be sent to the user to his registered email. |
| show booked tickets | 1. User is logged in to the system. 2. In the header, the user clicks on My tickets link. 3. List of tickets booked by that user is displayed. Otherwise no tickets message is displayed. |

# List of scenarios implemented in Phase 4

|  |  |
| --- | --- |
| Use case | Scenario |
| Choose meal and seat features. | * After the user filled the passengers form in book tickets use case the user will be redirected to seat page to choose seat to every passenger. * We used seats map from <https://github.com/omarmahili/seatchart.js> and applied some modifications on it. * The user can choose preferred meal if he/she wants on the same page. * After choosing seats, the user clicks GO! Button. * If the directionality is return, then user also have to choose the seats in that return flight. |
| Show ticket information. | * The user on the bookedTickets page. * If the user has booked tickets, then all his/her booked tickets will show and to every ticket there is details link. * The user clicks that link, then the information of that ticket is displayed. |
| Cancel booked ticket | * The user on the bookedTickets page. * If the user has booked tickets, then all his/her booked tickets will show and to every ticket there is delete link. * The user clicks that link, then a confirmation will show up. * The user confirms the deletion, then the ticket will be canceled. |

# Contribution of each member

|  |  |
| --- | --- |
| **Name** | **Tasks** |
| Hamad Aloqaily | Add flight, show booked tickets, Show ticket information. |
| Abdullah Aljamhour | Delete flight, send confirmation email, Cancel booked ticket. |
| Ibrahim Alsuhaim | Search for tickets, book ticket(s), Choose meal and seat features. |

# 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>