Software Requirements Specification

for

Distributed computer system for managing a company's flights

Version 2.0 approved

Prepared by: Mircea Andrei Pomacu

Robert Cristian Draghici

Ionut Gabriel Buzduga

Organization: University of Craiova,

Faculty of Automation, Computers and Electronics

03/13/2022

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
First Version	12.03.2022	Added first 3 chapters	1.0
First Revision	16.03.2022	Resolve issues with the first 3 chapters	1.0
Second Version	18.03.2022	Added the last chapters	2.0

1. Introduction

1.1 Purpose

The purpose of the document is to describe the requirements of creating an application for managing a company's flights.

1.2 Document Conventions

In this part of the document, we will create a table with the following abbreviations, acronyms and definitions for terms that are used in the creation of the report.

Abbreviations:

Term	**Abbreviations**
Administrator	Admin
Application	App
Information technology	IT
Information	Info
Departure	Dept
Arrival	Arr
N	Not Added/Not Applicable
IDE	Integrated Development Environment
HTML	Hypertext Markup Language
CSS	Cascading Style Sheets
SQL	Structured Query Language
DB	Database

Definitions:

Term	**Definition**
User	Someone who uses the application
Administrator	A person that has the role to manage and control the system of the application
Passenger	A person that books a flight in a company plane at a specific moment
Flight	The movement of a plane from a departure point to a destination point

1.3 Intended Audience and Reading Suggestions

This document represents the report of the application and can be used by the client or developers alike.This project is meant to be a prototype for a flight management system, which means that its functionalities are restricted to work with simulated data.It is more directed to college work, for the students involved to practice organization and teamwork under the guidance of the professor.
To have a good understanding of the project it is suggested reading the entirety of the SRS document.

1.4 Product Scope

The purpose of the web application for managing a company's flights is to facilitate the booking for the flights of our company for people that want to travel from one point to another with the company planes and all the complementary requirements for booking such as: cancel the booking, details about the date and time for the flight. The administrators have the right to edit the time for the flights or to cancel a flight due to explainable reasons.
Above all, we hope to provide a comfortable user experience along with the best features available.

1.5 References

Title	Link
Step-by-step ASP.NET MVC Tutorial for Beginners | Mosh	https://www.youtube.com/watch?v=E7Voso411Vs
React JS - React Tutorial for Beginners	https://www.youtube.com/watch?v=Ke90Tje7VS0
Web Design for Beginners: Real World Coding in HTML & CSS	https://www.udemy.com/course/web-design-for-beginners-real-world-coding-in-html-css/learn/lecture/1509624?start=165#overvie
C# Tutorial - Full Course for Beginners	https://www.youtube.com/watch?v=GhQdlIFylQ8

2. Overall Description

This section will give an overview of the whole system. The system will be explained in its context to show how the system interacts with other systems and introduce the basic functionality of it. It will also describe what type of stakeholders will use the system and what functionality is available for each type. At last, the constraints and assumptions for the system will be presented.

2.1 Product Perspective

This system is a self-contained product and will consist of: a web UI(web server), a database server and an application server. This web portal will be used by our customers to find information about flights from numerous companies and to buy tickets for the flights they find attractive. Nevertheless, the web portal will be used for managing the data corresponding to the clients and the available flights of the airline company and the system as a whole.

Since it is a data-centric product, developed to automate the process of administration of the data, the web application will need to store that data somewhere. For that, a database server will be used. The web portal will not only use the database to get the data, but also to add or modify it.

The web application will need to communicate to a database server and to an application server (figure 1) in order to support the construction of the web pages and then, to administrate the data of the clients and the transactions that they want to realize.

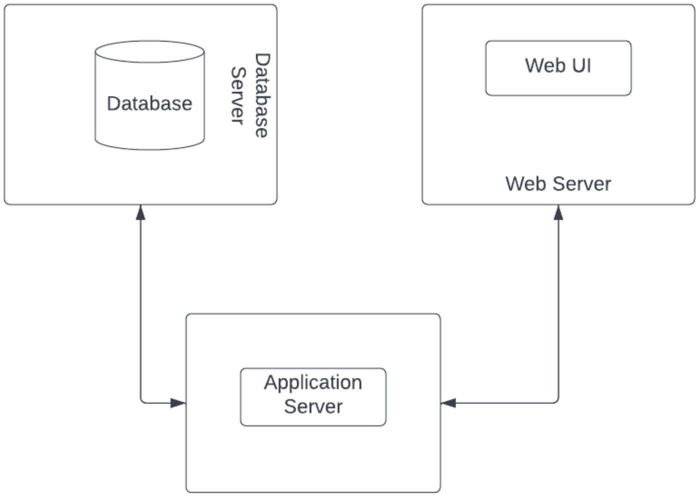


Figure 1

2.2 Product Functions

The following functions consists in the complement of the web app:

User functionalities:

- Book a flight on a certain date from one airport to another.
- Selecting their own seats in the plane.
- Ability to cancel a flight.
- Ability to change the date of the flight with another one in reasonable time.
- Ability to leave reviews about the flight and staff.

Admin functionalities:

- Ability to change the date or time of a flight in reasonable time.

- Ability to respond to reviews in time.
- Ability to see the personal information about every passenger of a flight.
- Ability to ban some passengers/users from the platform for misbehaving in a flight.

Common functionalities:

- Access to the application must be protected with the user and password for Administrators/Users.
- Ability to see the remaining free seats in a flight.

2.3 User Classes and Characteristics

There are 2 types of users that interact with the web app:

- Users/Customers that are using the app for the main purpose, to book flights from one point to another don’t need to have specific IT knowledge. This means that the users need to have access to a web browser and have basic knowledge on how to use web sites and how to reserve tickets online.

- Admins manages the entire web app. The administrator manages the information about each flight and passenger from that flight. This type of users’ needs to have the knowledge to resolve any issue about the IT implementation of the app.

2.4 Operating Environment

Any person who will use this application must have the following:

- Internet access
- Microsoft Windows 7 at least installed

One of the next modern browsers:

- Google Chrome (version 96.0.4664)
- Mozilla Firefox (Firefox 91 and 91 ESR)

Back-end server:

- Microsoft IIS

2.5 Design and Implementation Constraints

Constraints:

- Visual Studio IDE 2022 Community Edition
- HTML5
- CSS
- C# programming language version 8.0 or newer
- SQL Server 2019
- HTTPS protocol
- .NET Core 6.0

2.6 User Documentation

This word doc represents the software requirements documentation for the project. This document will replace the user guide and other documentation.

2.7 Assumptions and Dependencies

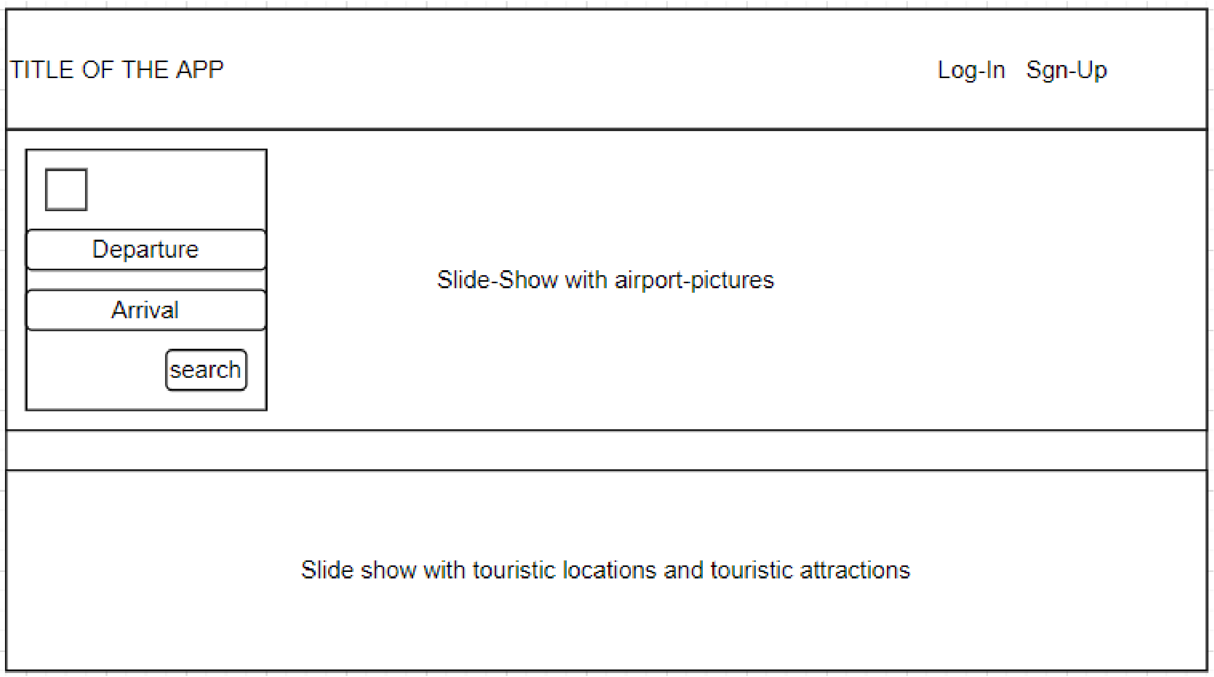
One assumption about the product is that it can be used on every PC that has access to a web browser.

Another assumption is that the GUI will look exactly the same on every PC, despite having different Operating System versions or using different browsers.

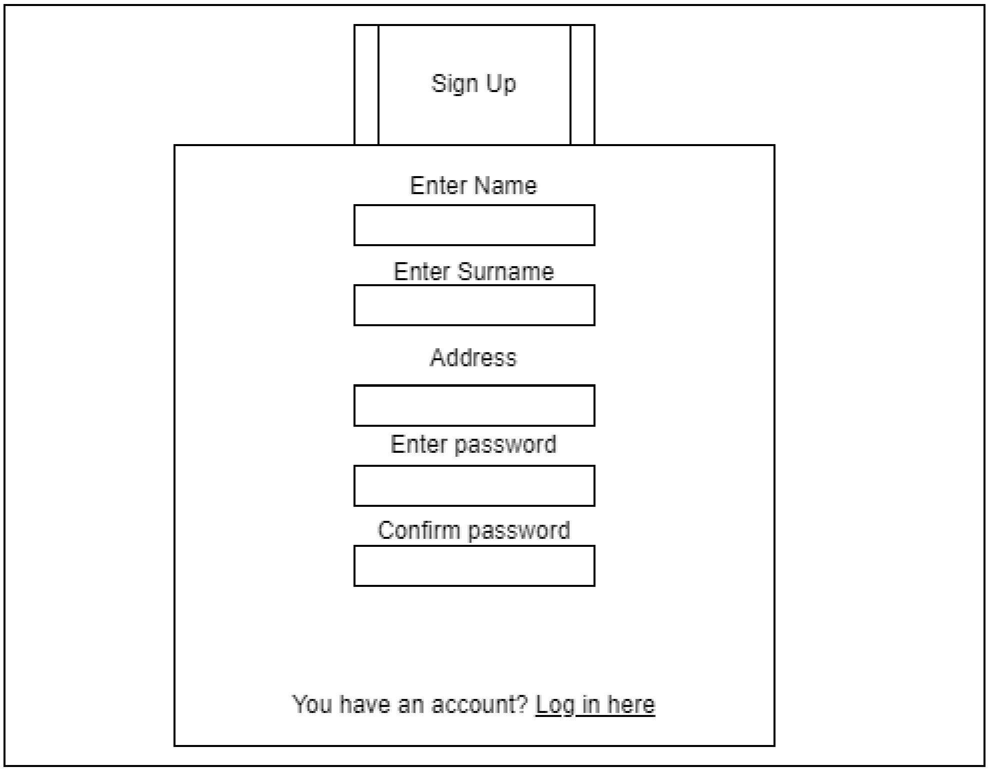
3. External Interface Requirements

3.1 User Interfaces

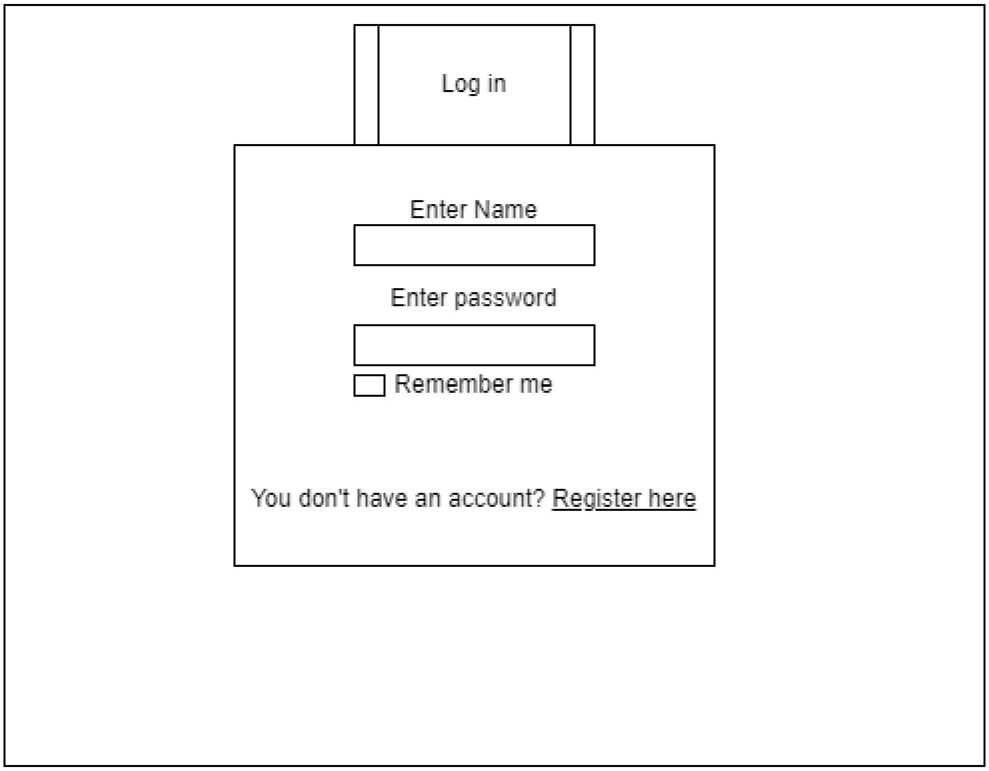
Main page:


TITLE OF THE APP
Log-In
Sgn-Up
□
Departure
Arrival
search
Slide-Show with airport-pictures
Slide show with touristic locations and touristic attractions

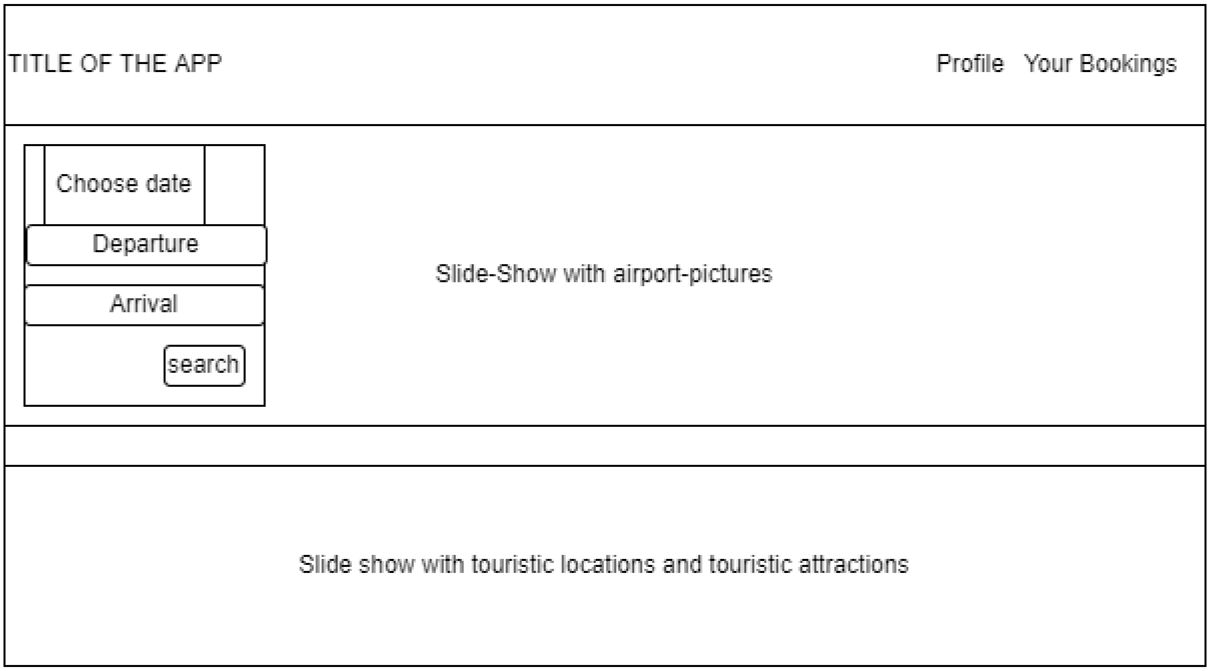
Register page:


Sign Up
Enter Name
Enter Surname
Address
Enter password
Confirm password
You have an account? Log in here

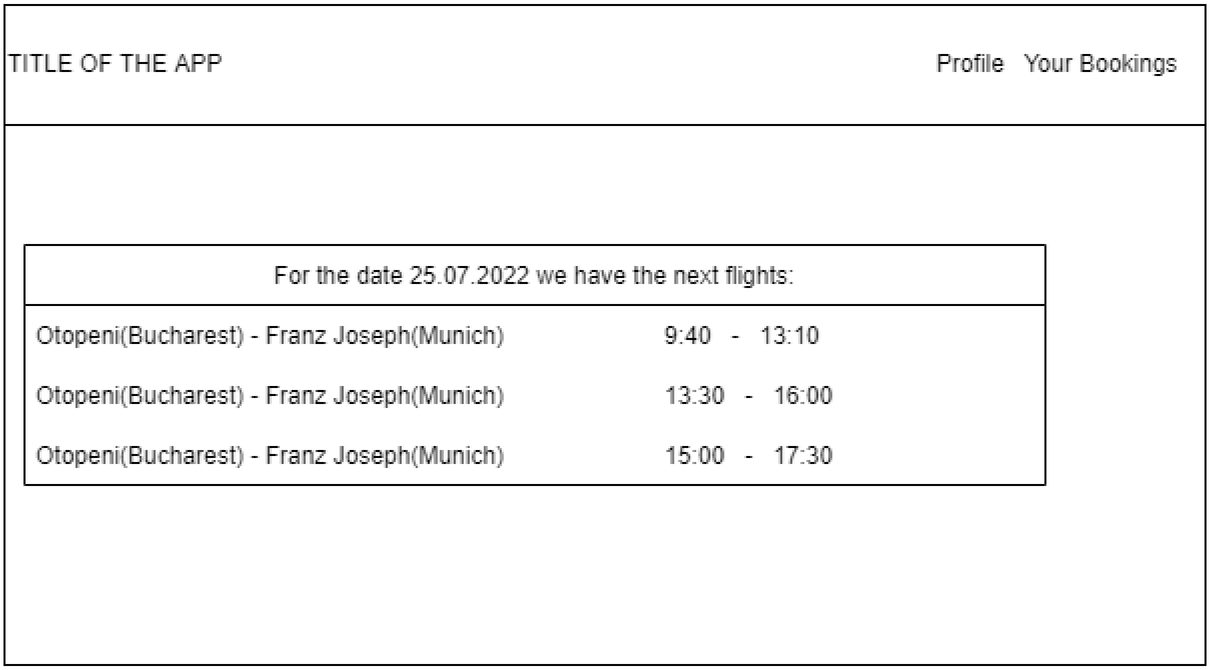
Login page:


Log in
Enter Name
Enter password
□ Remember me
You don't have an account? Register here

Main User page:


TITLE OF THE APP
Profile
Your Bookings
Choose date
Departure
Arrival
search
Slide-Show with airport-pictures
Slide show with touristic locations and touristic attractions

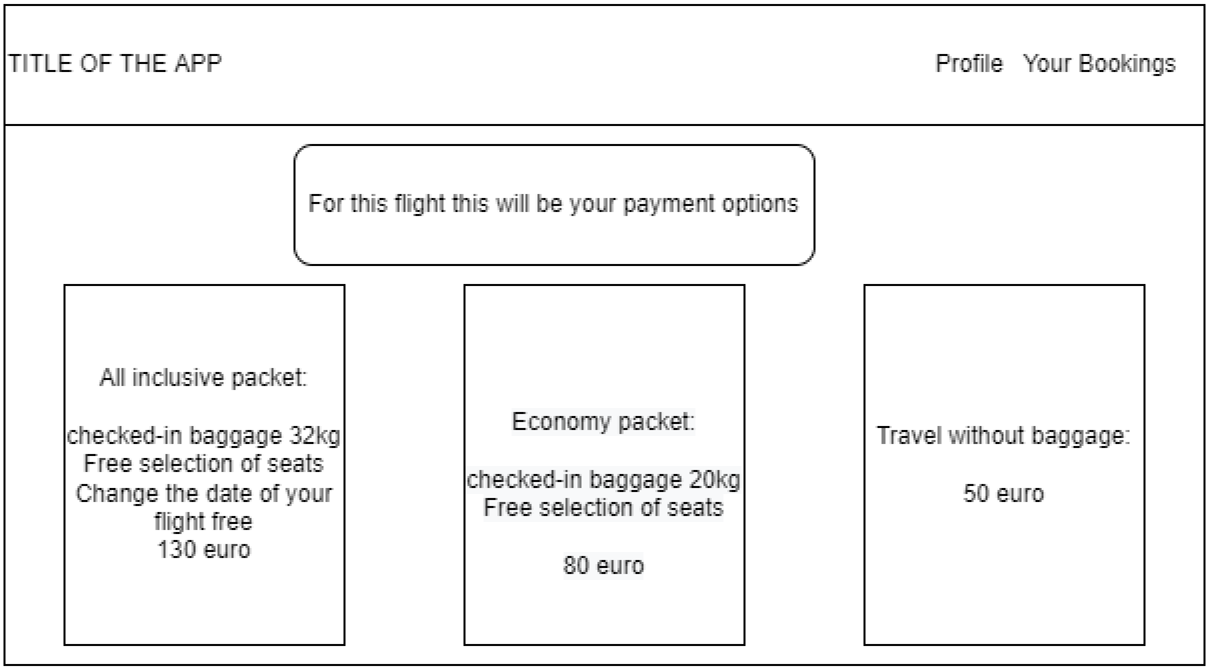
Flight list:



For the date 25.07.2022 we have the next flights:

Otopeni(Bucharest) - Franz Joseph(Munich)	9:40 - 13:10
Otopeni(Bucharest) - Franz Joseph(Munich)	13:30 - 16:00
Otopeni(Bucharest) - Franz Joseph(Munich)	15:00 - 17:30

After you select a flight a price list is shown:



For this flight this will be your payment options

All inclusive packet:

checked-in baggage 32kg
Free selection of seats
Change the date of your flight free
130 euro

Economy packet:

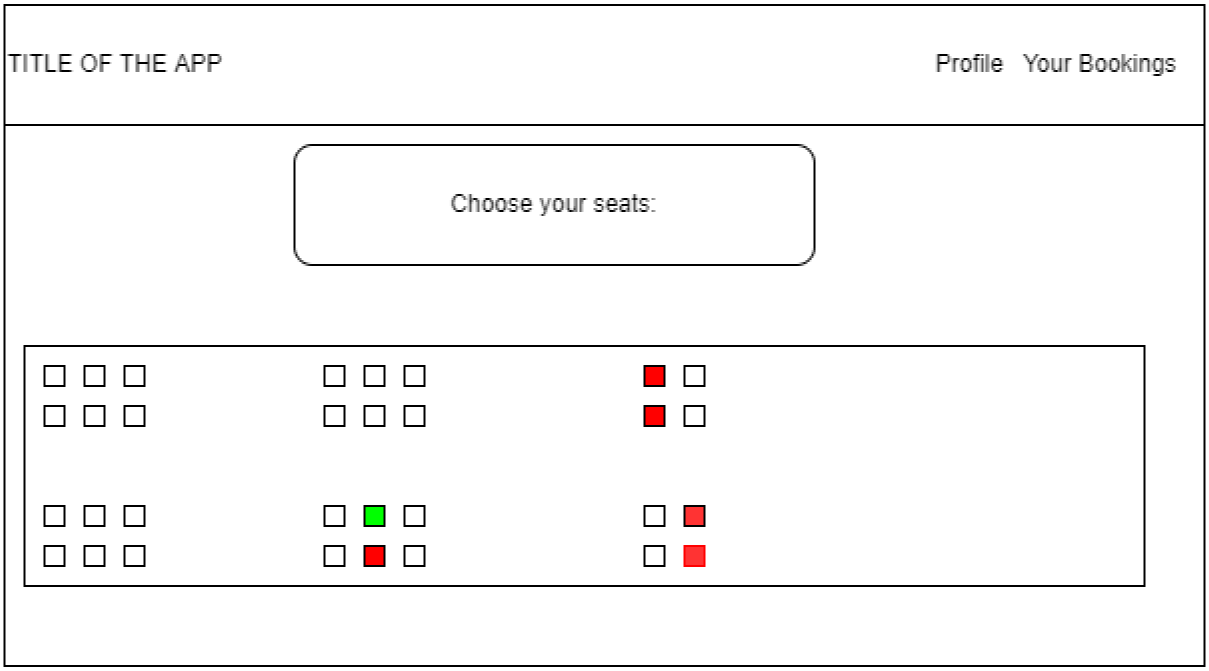
checked-in baggage 20kg
Free selection of seats

80 euro

Travel without baggage:

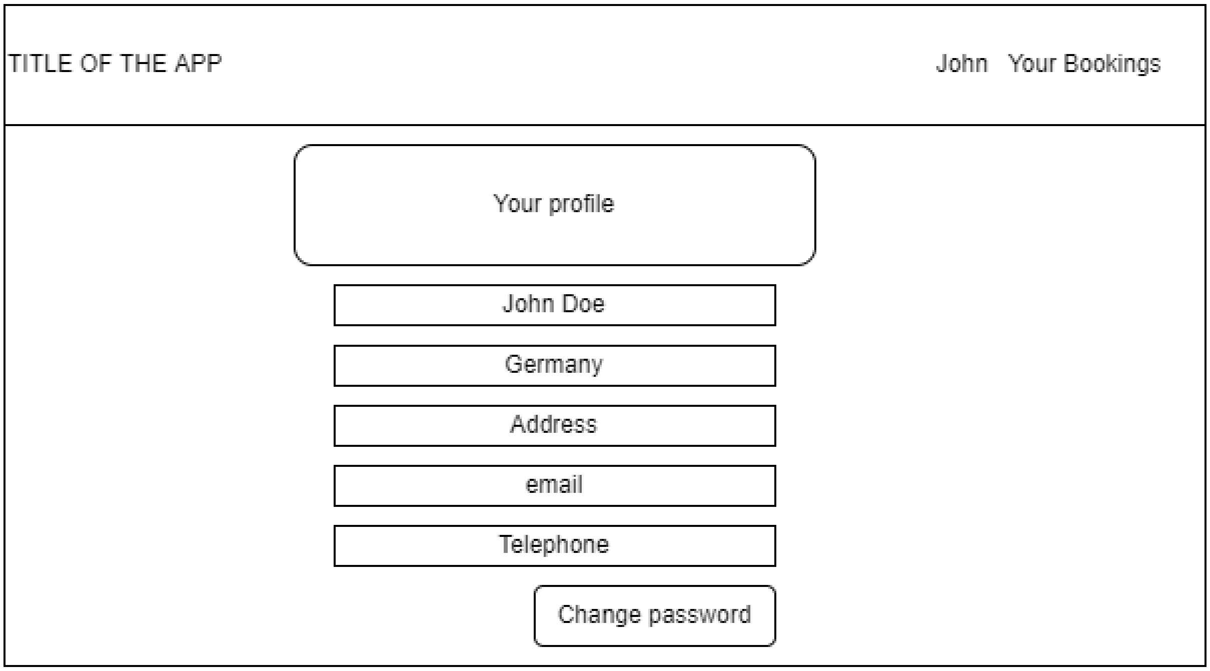
50 euro

After you select a packet, you can choose your seat and the number of seats you want to select. Maximum 3 persons. The users need to add their credentials too.



The white seats are not taken, the red ones are already booked, and green are the ones that the user wants to book. From the admin point of view, all the booked seats are red and when you press a seat he/she can see the credentials of that person.

Profile page:



TITLE OF THE APP

John Your Bookings

Your profile

John Doe

Germany

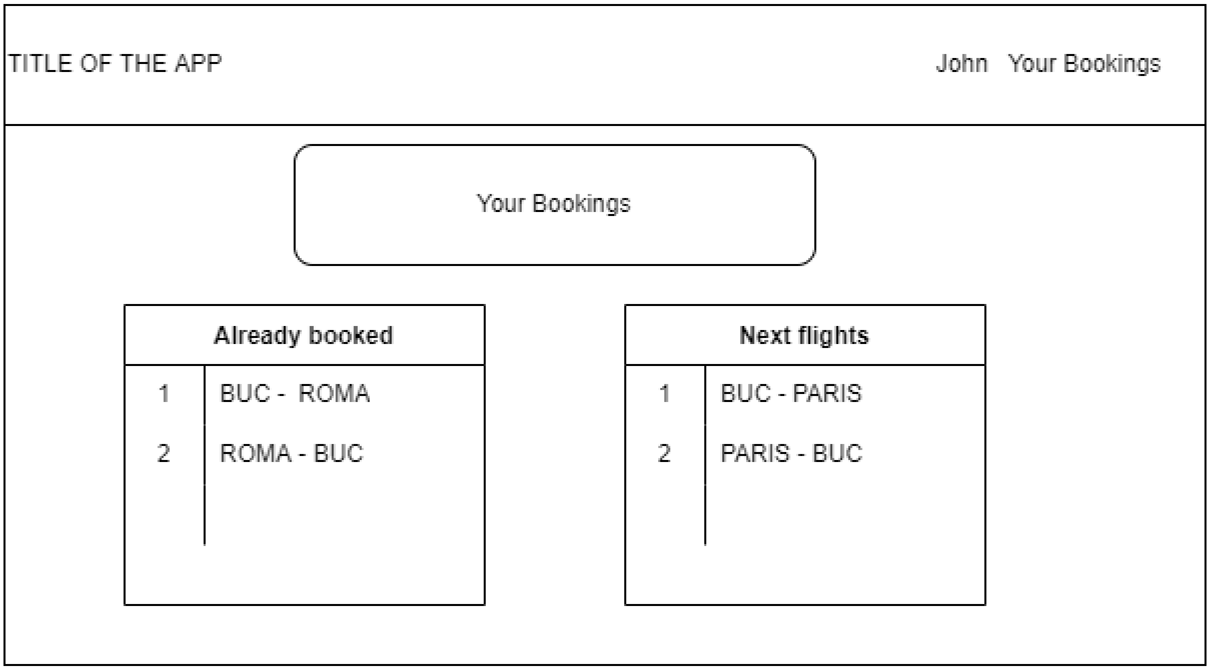
Address

email

Telephone

Change password

Your bookings page:



TITLE OF THE APP

John Your Bookings

Your Bookings

Already booked	
1	BUC - ROMA
2	ROMA - BUC

Next flights	
1	BUC - PARIS
2	PARIS - BUC

3.2 Hardware Interfaces

N

3.3 Software Interfaces

We will be using .NET Entity Framework connected to the Microsoft SQL DB Server. This product is designed to manage the communication between back end and the DB.

3.4 Communications Interfaces

The communication interface will be HTTPS.

4. System Features

4.1 Login

4.1.1 Description and Priority

This feature will be used by the administrators and users alike. In order to login the user must have already created an account.

4.1.2 Stimulus/Response Sequences

The user must enter the username and password to be able to enter in the account. After the information is added in the textboxes and press the login button, then the database is accessed, and the email and the password will be searched. If the user’s email and password match in the database then the user can access his account.

4.1.3 Functional Requirements

- This type of feature can be used by users and admins.
- Email text-box: The user/admin can add their email in order to be identified.
- Password text-box: the user/admin can add their password in order to enter in the account.
- Remember me check-box: If this check-box is selected then at another access of the web page the account is still active.
- Login button: it is pressed if you want to access your account. It will check the email and the password to match.
- If the password or email is incorrect, an error message is displayed.

4.2 Register

4.2.1 Description and Priority

If the user doesn’t already have an account, this feature will be used in order to generate one. Otherwise, the user will not be able to access the application and use its functionalities.

4.2.2 Stimulus/Response Sequences

The user has to enter its name and surname, email address and also choose a password which will be introduced twice for confirmation.The next step is to press the sign up button which stores the information generated for the newly created account. The output is represented by a confirmation email and the ability to enter in the account.

4.2.3 Functional Requirements

- This type of feature will be used by the customers to create an account

- Name and surname text-boxes which represent the full name of the user
- Email text-box: The user can add their email in order to be identified.
- If the email doesn't exist, an error message is displayed.
- Password text-box: the user can add their password in order to create the account.
- Confirm Password text-box:the user needs to retype their password to confirm it
- An error message will be displayed if the passwords do not match
- Sign up button:by pressing this button after entering valid information the account will be created

4.3 Main page for user

4.3.1 Description and Priority

Users can see on the screen the choosing table for the flight and other commercial information regarding the airports and tourist locations.

4.3.2 Stimulus/Response Sequences

The input is represented by the queries regarding the information about flights and dates. The output is represented by the available flight for that specific date.

4.3.3 Functional Requirements

- The date text bar: where the date could be selected for a specific flight.
- the departure text box: choose the boarding airport for the flight.
- the arrival text box: choose the destination airport for the flight.
- A slideshow with airports. If a picture with an airport is selected a short description about that airport should be displayed.
- A slideshow with touristic destinations. If a picture with a destination is selected a short description about that region should be displayed.
- Profile button. If pressed, the user should see his/hers credentials.
- Booking button. If pressed, the user could see his/hers last bookings(means that the user had already had that flight) and next bookings for future flights

4.4 Main page for admin

4.4.1 Description and Priority

After the admin gets logged in the page he/she could see a list with all the flights that will take place in the next month.

4.4.2 Stimulus/Response Sequences

The input is represented by the selection of a specific flight and as a response the DB should display all the persons that have already booked seats for that flight.

4.4.3 Functional Requirements

- At every flight from that list the admin can see the number of people that have already booked a flight just hoover-ing a flight tab. Based on the number of passengers the admin has the power to cancel a flight if the number is less than 5.
- Clicking a flight on the screen will appear the names of all the passengers and their positions on the seats.

- Clicking on a passenger, the admin can see the name, email and telephone number.
-

4.5 Flight list

4.5.1 Description and Priority

After entering numerous criteria for the desired flight, the user will be shown a list of available flights that met the conditions entered on the main page.

4.5.2 Stimulus/Response Sequences

The input is represented by chosen criteria for the flight (date, cost, destination) and the output is represented by generating the list of flights that correspond.

4.5.3 Functional Requirements

- Profile button. If pressed, the user should see his/hers credentials.
- Booking button. If pressed, the user could see his/hers last bookings(means that the user had already had that flight) and next bookings for future flights
- In the middle there will be a “flights” menu that will show a list of all available flights on the selected date
- every flight will show their: airport of departure, airport for arrival and time of departure and time for arrival

4.6 Tickets selection

4.6.1 Description and Priority

This feature will be used by the user to select a certain type of packet (economy, first class) for the tickets he/she wants to buy/reserve.

4.6.2 Stimulus/Response Sequences

The input is represented by choosing one of the packages and the output is introducing the customer to the seats selection page.

4.6.3 Functional Requirements

- Profile button. If pressed, the user should see his/hers credentials.
- Booking button. If pressed, the user could see his/hers last bookings(means that the user had already had that flight) and next bookings for future flights
- a text box specifying the chosen flight and the available payment options for it
- a price list with different packages, each with their rights and bonuses
- details about each package are shown when hovering over them

4.7 Seats selection

4.7.1 Description and Priority

After selecting a ticket price the user will be sent to the Seats selection page where he/she can select an available seat if they have an All-inclusive or Economy ticket, otherwise the seats will be allocated by the system.

4.7.2 Stimulus/Response Sequences

The input is represented by the ability of the user to click on an available seat and confirm it.This information will be sent to the database, which in response will send back the changed list of available seats.

4.7.3 Functional Requirements

- An interface with the available seats where the color of the seats dictates their availability :
    - ➢ White means available for booking
    - ➢ Red means not available for booking
    - ➢ Green indicates your chosen seats for booking
- A confirmation button to ensure that the user does not click on a wrong seat by accident

4.8 Edit User Profile

4.8.1 Description and Priority

The user can edit the login and contact information and also change the password of the account.

4.8.2 Stimulus/Response Sequences

The input is represented by the newly entered information for each text field which will update the database.The response can be the email sent in case the user requested to change his/hers password

4.8.3 Functional Requirements

- Edit buttons for the following fields:
    - ➢ Name
    - ➢ Country
    - ➢ Address
    - ➢ Email
    - ➢ Telephone
- The Change Password button has the purpose of letting the user change the password of his/hers account by receiving an email from the admin.

4.9 Booking of an user

4.9.1 Description and Priority

The user can see his/her own bookings that have already been used and also bookings in process.

4.9.2 Stimulus/Response Sequences

The input is represented by the reviews that a user can add on processed bookings and the output is represented by lists with processed bookings and in progress bookings

4.9.3 Functional Requirements

- The user will receive 2 lists displayed on the screen. One with the processed bookings and another with in progress bookings.
- The user can select a processed booking and leave a review about the staff and flight

- The user can select an in progress booking and cancel it within a period selected by the admin.
- If the booking is canceled after that period a fee should be applied.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- The web app must retrieve the information from the database quick (1,2 seconds)
- The time response when a seat is taken should be instant. After this step is successfully done the seat should be blocked to be selected by another passenger.
- The adding in the database of a new customer should be in less than 2 seconds.

5.2 Safety Requirements

N

5.3 Security Requirements

- When the user/admin enters the password, it shall not be visible.
- The passwords shall be encrypted in the database.
- Only important credentials can be seen by the admin for every customer such as: name, address, email, telephone. The password can't be seen.
- If a flight is canceled a day before the flight due to unforeseen events created by the company, the passenger will have a discount of 60% on the next flight that it booked via our app.
- Without a confirmation booking sent via email you can't proceed to the payment step.If the user does not receive an email they should contact the admins of the app via email at admins@company-name.org.
- Website data must be backed-up in the database in order to prevent sudden loss of information

5.4 Software Quality Attributes

The following rules shall be respected throughout the implementation code:

- Source code should respect SOLID principles.
- Each new piece of code should be commented on.
- The variables in the code should have meaningful names.
- Simplicity: the code must be easy to understand for any programmer.
- Source code must not contain duplicated code.
- Every feature should have an integrated test that shows us that it has the right functionality.
- The quality of being debuggable: capability of being (easily) debugged.

- Effectiveness: the capability of producing a desired result or the ability to produce desired output.

5.5 Business Rules

N

6. Other Requirements

N

Appendix A: Glossary

N

Appendix B: Analysis Models

N

Appendix C: To Be Determined List

N