Software Requirements Specification

for

Airline Ticket booking and conformation system

Version 1.0 approved

Prepared by:

Bhavani Kalal : 21CS30027
Voddula Karthik Reddy : 21CS30058
Parvathaneni Venkatsai Mokshith : 21CS10050

Indian Institute of Technology, Kharagpur

27/3/2022

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

The Purpose of this Software Requirements Specification document is to present a detailed description of Airline Ticket Booking and Conformation system This document will explain the purpose and features ,interfaces different types of users and their interfaces ,constraints under which the system will operate .

The project focuses on building an hassle free online flight booking website which accounts for safety of passengers keeping mind about recent cowid pandemic.

1.2 Document Convention

Typographical Conventions :

All abriveated words are written in capital letters. following are the abreviations used in this document.

ATBCS	Airline ticket booking and conformation system
V	Version
OS	Operating System
FFM	Frequent flier miles

1.3 Intended Audience and Reading Suggestions

The intended audience for this project would be software developers, engineers, and designers who are interested in building flight management systems with special functionalities for passenger safety, particularly in the context of the current pandemic situation. It may also be of interest to airline companies and travel agencies who are looking to enhance their booking systems and offer additional safety measures to their customers

1.4 Product Scope

ACTBS *is a website used to book aeroplane tickets online in a seamless manner and intends to provide safe journey to its travelers by verifying their cowin certificates if they are above 15 years. The software must have applealing interface inorder to attract more customers to the website,regular customers should be rewared ,for which* FFM *system is adopted accordingly .*

1.5 References

https://www.npmjs.com/package/node for npm

https://www.mongodb.com/docs/ for mongodb

https://www.npmjs.com/package/react for react

2. Overall Description

2.1 Product Perspective

ATBCS V 1.0 *is new self-contained product which is a component of* ATBCS ,which *requires users to login ,register ,find flights and needs to verify cowin cerificates of the passengers eligable* ATBCS V 1.0 implements login,register, backend part of find flights also initial backend implementation of verifying cowin certificates.

2.2 Product Functions for ATBCS

1. *Must allow users to create an account.*
2. *Verify the user through email.*
3. *Must allow users to login.*
4. *Must be able to search flights.*
5. *Must be able to verify cowin certificates.*
6. *Must allow registered users to book tickets after successful verification.*

2.3 User Classes and Characteristics

Visitors : Allowed to search flights from a source to a destination at a given date with given number of passengers ,class of choice . Allowed to view the search result ,flights with their time taken to travel , flight number ,travel cost per adult in chosen class.

Members : A member is a Visitor who is registered with the airline after succesful login he will able to do functions of a user plus being able to enter details of each passenger, upload their cowin certificate if applicable, proceed to check out ,pay the required sum using card and gets a conformation email regarding same.A member must give details of passengers while booking a ticket, must upload cowin certificate of passengers if applicable. A member can use already accumiliated frequent flier miles to avail discounts on ticket fare,also a member receives ffm after sucssesful purchase of the ticket.

User: Any one using the program at present

2.4 Operating Environment

The software can be used by any user with access to a device with the internet. The Portal will have a server from which it will retrieve all the information all the information and the database will be maintained at the server. The software can run on any modern device which can load a modern web browser and has a reliable internet connection.

2.5 Design and Implementation Constraints

1. For now, multi-city is not given as an option, though in the results it shows them too to simply the search function.
2. For now , only visitors and members are the user classes to simply the login page and navigation bar.

2.6 User Documentation

All user documentation would be made available via the README of the repository.

2.7 Assumptions and Dependencies

- **Assumptions:**

  -- Only the admins will have the access to the central database
  -- Only admins can add or delete flight schedules
  --

- **Dependencies:**

  -- Node.js: Backend Framework
  (Version: 18.15.0)
  -- MangoDB: Database
  -- mongoose: Database connection
  (Version: 7.0.2)
  -- Express.js: Web application Framework
  (Version: 4.18.2)
  -- React: Frontend Framework
  (Version: 17.0.2)

3. External Interface Requirements

3.1 User Interfaces

Member Interface:

1. He can view flights which travel from source to destinationon a particular day.

2. He can get the ticket rates.

3. He can book tickets.

4. He can view his previous bookings.

5. He can edit the booking which is alive i.e., cancel thetickets

Visitor Interface:

1. He can view flights which travel from source to destination on a particular day.

2. He can get the ticket rates.

3.2 Hardware Interfaces

Since the software is a web application, it only requires a basic hardware interface such as laptop, mobile, or computer where the user can run the software.

3.3 Software Interfaces

5 databases would be maintained:

- Airport database: This database would contain the essential information about airports. Their name, code and other basic information will be stored here.
- Flights database: This database would contain the essential information about flights. Their name, number of seats in respective classes and other basic information will be stored here.
- Schedules database: This database would contain the date wise schedules. Their ID, date, array of flights will be stored here.
- Sessions database: This database would contain the session of the user. Their ID, expired time, and some other basic information will be stored here.
- Users database: This database would contain the essential details of the users. Their ID, name, DOB, email, password, phone number, ffm, and other information regarding their previous bookings are stored here.

3.4 Communications Interfaces

1) User -frontend communication happens through the front end UI using a web browser 2)

front-end backend communication happens thorough HTTP requests.

3) Backend - database communication happens through mongoose 4)

customer - company communication happens through email.

4. System Features

4.1 Search Flights

This feature allows visitors to search flights based on their preferences This feature is of high importance as it is required to find flights inorder to book tickets.

Response Sequences:
1)The visitor is should enter their date of travel ,source ,destination,number of passengers ,class of travel in the search form to get the flights falling under their specifications.

2) Each flight shown in the search result has duration of journey and the price per adult.

3) If the visitor chooses specific flight then he/she would be directed to passenger information form if the visitor is logged into his/her account.

Functional Requirements:

1)The software must maintain a flight schedule database for each day.

2)The data should be stored in such a way that quiring on the data is easy and fast to give the visitor a smooth experiance.

3)The software should be able to determine if a user is currently logged in.

4.2. REGISTER WITH THE SOFTWARE

This feature allows the visitor to register with the software and create his /her account.The priority of this feature is high.

Response Sequences:

1) The visitor should enter his/her name as per official documents,email,phonenumber,email,date of birth,password.

2) Then a mesage of conformation email has been sent would be displayed

3) The visitor must then open his/her entered email and click on the link provided

4) Then if the link is valid he/she would be getting a success message and would beredirected to login page.

Functional Requirements:

1) The system must store user date in a database
:

2) Password of the user must be stored as encrypted form in the database for securityreasons

3) The system must be able send a conformation email to the user4) The system must be

able to verify the link.

4.3 Login

This feature allows the user to sign-in login into the software , this feature is of high priority because it is required for the user to login inorder to proceed with booking

Response sequences :

1)The user would be asked to enter his/her email and password .

2)upon valid credentials he/she would be redirected to home page.

3)Upon invalid credentials the software would display invalid credentials.

Funcitional Requirements :

1) The software should be able to access the user database.

2) The software should be able to encrypt the entered passowrd and compare with encryptedpassword.

4.4 Uploading Passenger Details and verifying Cowin certificates.

This feature allows the user to enter details of passengers like their name,date of birth and upload their cowin certiciates.System would then verify the certificate for the safety of the passengers. The priority of this feature is high for security reasons and to avail discounts for infants.

Responce Sequences:
1) The user would be asked to enter the passenger details and will be asked to upload cowin cerificates of passengers if they are above 15 years of age.
<TBD>^1

Funcitional Requirements :
<TBD>^2

4.5 Payment Portal

This portal allows the user to pay for the ticket and coform his/her ticket .This functionality is of high priority.

Response Sequence :

1) The user would be directed to this page once his/her cowid certicicates are verified,.
2) The user would be asked to pay the sum for his/her ticket

3) upon succesful payment the user would success message on the screen and would receive aconformation email.

Functional Requirements:

1) The system should be able to proccess the payment and verify the state of the payment .

2) The system should be able to send conformatin email to the customer.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

1.ATBCS is a web-based application which can work on decent internet connection connection.

2. This can work on popular operating systems such as Windows , Linux or Mac without any glitches.

3. The system should be able to send requests and perform computations on the data received from it for the application to work smoothly .

4. This application is compatible with any recent versions of browsers.

5.2 Safety Requirements
1. Users should logout after their usage so as to avoid unnecessary load on the server.losing access to their account.

2. For now, reset password feature hasn't been added so user must remember their password to avoidscs

5.3 Security Requirements

1. Password-based logins for all users and all the passwords are secured using hashing before storing in database.
2. A token is sent when user logs in and checked for authentication every time it sends a request to the server.

5.4 Software Quality Attributes

1. Object Oriented Design principles has been implemented to change the software in future according to our future requirements.
2. The software can be used on any device using a browser.
3. Extensive testing of the software will be done and there will be minimal bugs and errors.

6. Other Requirements

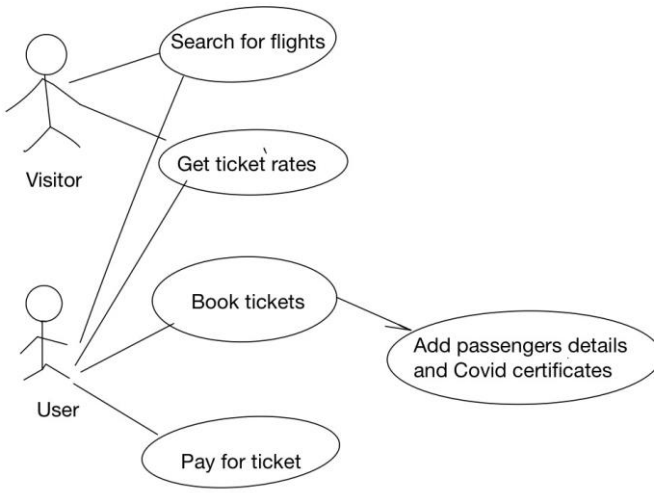
GUI should be effective, interactive and intuitive to increase user experience. The total application should be as sleek as possible to decrease loading time and to improve user experience.

Appendix A: Glossary

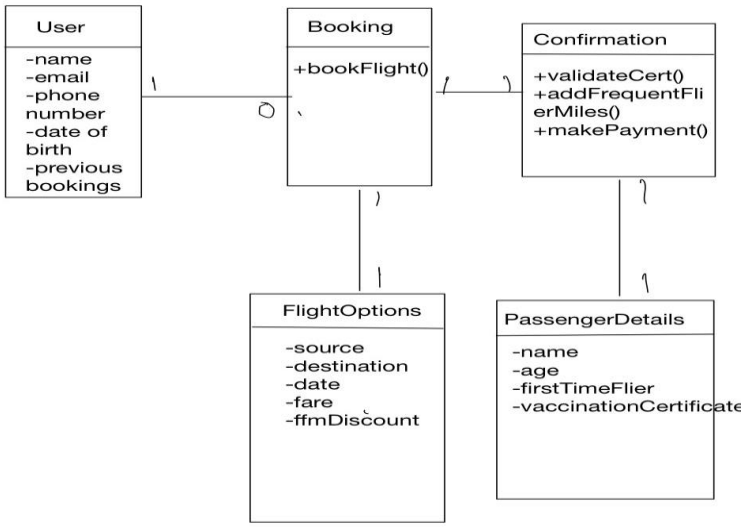
GUI: Graphical User Interface which is a form of user interface that allows users to interact with electronic devices through graphical icons.

SRS: Software Requirements Specification, A document that completely describes all of the functions of a proposed software.

Appendix B: Analysis Models



Use case diagram



Class Diagram

Appendix C: To Be Determined List

TBD^1 : Response sequence of Verifying cowin certificate
TBD^2 : Fucntional Requirements of verifying cowin certificate.