**FlyAway Travel Booking Project Specification**

|  |  |
| --- | --- |
| **Version # and Author** | **Version History** |
| 1.0 – Binu Thankachan | October 1,, 2021 - Created document with the following sections: |
|  |  |
|  |  |

Contents

[Sprints planned and the tasks achieved in them 3](#_Toc84092023)

[Project Epic: 3](#_Toc84092024)

[User Stories and Sprint Assignments: 4](#_Toc84092025)

[Algorithms and flowcharts of the application 5](#_Toc84092026)

[Modules developed for the application 7](#_Toc84092027)

[Core Java concepts used in the project 8](#_Toc84092028)

[Link to the GitHub repository 9](#_Toc84092029)

[Conclusion on enhancing the application and defining the Unique Selling Points 10](#_Toc84092030)

# Sprints planned and the tasks achieved in them

## Project Epic:

Create a dynamic web application for FlyAway a Travel Booking Portal that lets people book flights on their website.

The application will:

* + Provide a search form in the homepage to allow entry of travel details, like the date of travel, source, destination, and the number of persons.
    - Based on the travel details entered, it will show the available flights with their ticket prices.
      * Once a person selects a flight to book, they will be taken to a register page where they must fill in their personal details.
      * In the next page, they are shown the flight details of the flight that they are booking, and the payment is done via a dummy payment gateway.
      * On completion of the payment, they are shown a confirmation page with the details of the booking.
  + Provide an admin backend with the following features:
    - An admin login page where the admin can change the password after login, if he wishes
      * A master list of places for source and destination
      * A master list of airlines
      * A list of flights where each flight has a source, destination, airline, and ticket price

## User Stories and Sprint Assignments:

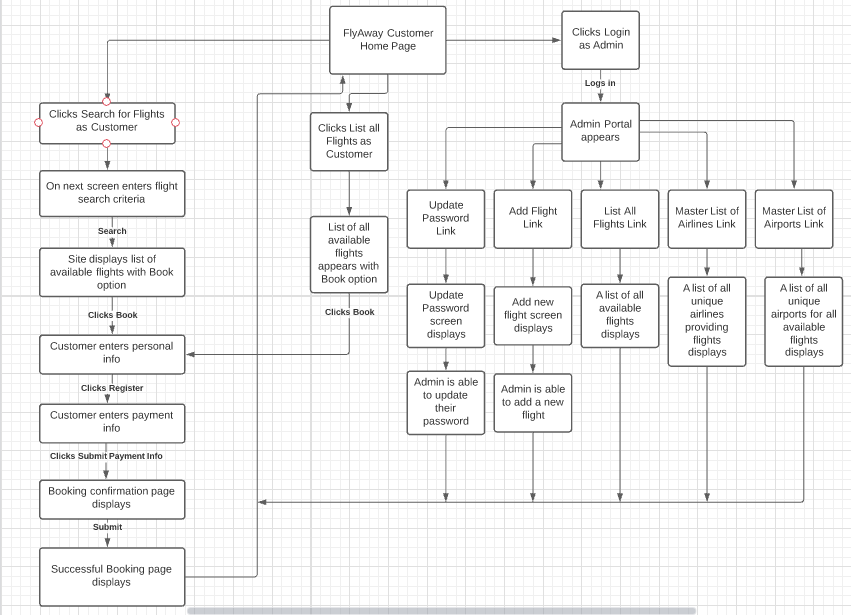
|  |  |
| --- | --- |
| **User Story** | **Planned Sprint** |
| As a customer I want a FlyAway travel booking home page that I can access with links to available options | 1 |
| As an Admin I want a FlyAway travel booking portal that admins need to login to and cannot be accessed by those that aren’t admins and containing links to available options | 1 |
| As a customer I want a flight search page accessible by anyone that accepts the following search criteria and displays available flights with prices matching that criteria:   * Departure Date * Source airport * Destination airport * Number of travelers | 2 |
| As a customer I want the ability to select a flight to book from the available flights listed on the customer Flight Search page. I should be given:   * a page to enter my customer details * a page to enter my payment details * a confirmation page where I can confirm the booking * the flight should be booked after I confirm the booking and a page should display booking details | 3 |
| As an Admin I want an admin login page where the admin can change the password after login, if he/she wishes | 4 |
| As an Admin I want a page that displays a master list of source/destination airports | 4 |
| As an Admin I want a page that displays a master list of airlines offering available flights | 4 |
| As an Admin I want a page that displays a list of all available flights where each flight has:   * a source airport * a destination airport * the airline * ticket price | 4 |

# Algorithms and flowcharts of the application

Develop a dynamic FlyAway Airline Flight Booking web application with the following:

* A customer landing page that displays links to the following available options

1. Search for available flights
   1. Each flight displayed will have a “Book” option that will allow the customer to book the flight
2. List all available flights
   1. Each flight displayed will have a “Book” option that will allow the customer to book the flight
3. Login as an Admin
   1. The user will be presented with an Admin login page where they can enter their user name and password
      1. Unsuccessful login and an error page will be displayed
      2. Successful login will display:
         1. Update password link
            1. This will update their password in the user table in the database
         2. Add new flight link
            1. This will add a new flight record to the flight table in the database
         3. List all available flights link
            1. This will display a page showing all flights listed in the flight table in the database
         4. Master list of Airlines link
            1. This will display a page showing the unique set of airlines from all available flights in the flight table in the database
         5. Master list of Airports link
            1. This will display a page showing the unique set of source/destination airports from all available flights in the flight table in the database.



# Modules developed for the application

1. Java modules developed were:
   1. Data transfer objects
   2. Servlet Controller class
   3. DAO classes, both interfaces and implementation classes
   4. JSP view files
2. The ControllerServlet class is a servlet that controls handling of user url request mappings and routing of those request mappings to the appropriate handler methods for processing.
3. DAO interfaces and implementation classes segmented and abstracted out the data access mechanism for the web application and the various database files it needed to communicate with.
4. The data transfer object classes encapsulated the fields, instantiation and getters/setters for for each of the database files the web application communicated with.
5. JSP view files were developed for each of the web displays needed by the user for each interaction with the website.
6. Maven POM file to handle dependency management

# Core Java concepts used in the project

The core concepts used in this project were:

* Object Oriented Programming
  + Utilizing classes to represent entities
* Exception handling
  + Intercepting and processing application errors
* Code modularity
  + Separating processing via separate classes
* Collections
  + List, ArrayList, Set, HashSet
* Data Structures
  + ArrayList used to hold names of files
  + HashSet used to get a list of unique airports and airlines
* JDBC
  + To control communication and data access between Java layer and Microsoft SQL database
  + Statement and PreparedStatement and query language
* Servlet
  + To serve as a controller and router of requests to handler methods
* JSP
  + To provide a view layer for the web application

# Link to the GitHub repository

<https://github.com/bthank/FlyAwayAirlineBookingProject>

# Conclusion on enhancing the application and defining the Unique Selling Points

Future application enhancements could include:

* Adding a better user front end that is more appealing to the user
* Adding registration and login functionality for customers in order to use the application
* Adding security to allow authentication and authorization in order to use the application
* Adding roles to govern how users can interact with the website and its data
* Adding email notification capability for booking and when flights become available for user criteria
* Adding real payment processing capability
* Adding additional error handling