**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](#_Toc84059660)

[Project Epic: 3](#_Toc84059661)

[User Stories and Sprint Assignments: 4](#_Toc84059662)

[Algorithms and flowcharts of the application 4](#_Toc84059663)

[Modules developed for the application 6](#_Toc84059664)

[Core Java concepts used in the project 7](#_Toc84059665)

[Link to the GitHub repository 8](#_Toc84059666)

[Conclusion on enhancing the application and defining the Unique Selling Points 9](#_Toc84059667)

# 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:

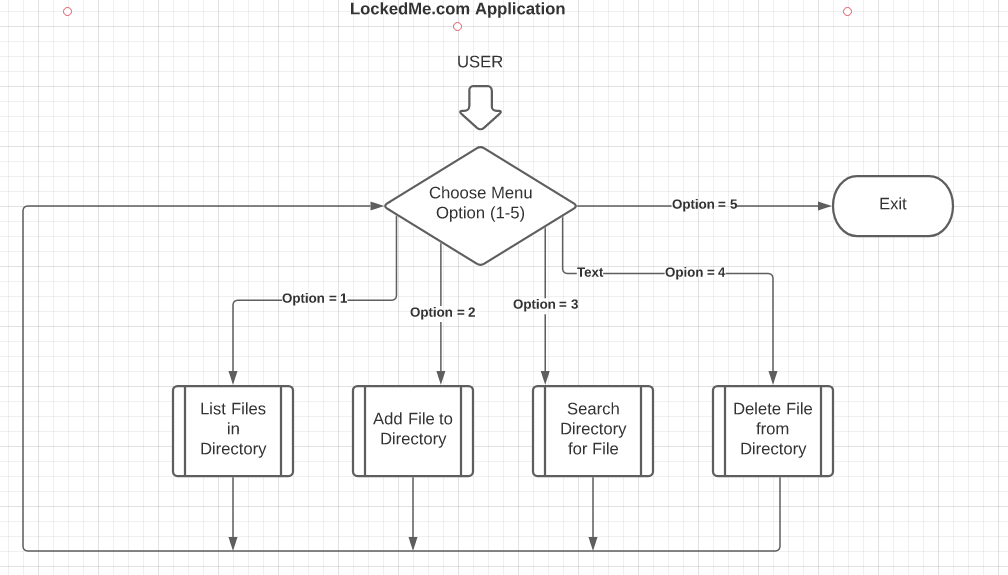
|  |  |
| --- | --- |
| **User Story** | **Planned Sprint** |
| As a customer I want a FlyAway travel booking home page that I can access with links to available options | 1 |
| As an Admin I want a FlyAway travel booking portal that admins need to login to and cannot be accessed by those that aren’t 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 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 the ability to enter my customer details, my payment details and a confirmation page should appear before submitting the booking. I should also be able to book the flight after confirming the booking. | 3 |
| As a user I want to be able to **search for** a file from a computer directory I specify | 3 |
| As a user I want to be able to **delete** a file from a computer directory I specify | 3 |
|  |  |
|  |  |

# 

# Algorithms and flowcharts of the application

Develop a loop that continuously loops until the user chooses to exit the loop

* Within the loop, display a menu allowing the user to choose 1 of 5 options.
  + Option 1 will display the list of all files in a user specified directory
  + Option 2 will add a new file with content to a user specified directory
  + Option 3 will search the user specified directory for a file
  + Option 4 will delete a file from a user specified directory
  + Option 5 will exit the loop



# Modules developed for the application

1. **The LockedMeApplication class** is the main entry point for the application.
2. The **LockedMeConsole class** encapsulates console user prompt handling for the LockedMeApplication.
3. The **LockedMeUserInput class** encapsulates console user input handling for the LockedMeApplication.
4. The **FileManager class** encapsulates file handling functions for the LockedMeApplication.

# Core Java concepts used in the project

The core concepts used in this project were:

* Object Oriented Programming
  + Utilizing classes to represent entities
* File handling
  + Creating, reading, finding and deleting files in the file system
* Terminal I/O
  + Scanner
* Exception handling
  + Intercepting and processing application errors
* Code modularity
  + Separating processing via separate classes
* Collections
  + ArrayList
* Data Structures
  + ArrayList used to hold names of files
  + List.sort() used to sort the ArrayList in ascending order
* Control mechanisms
  + While loop to control user prompts, input and exit
  + Sentinal value used to control loop entry and exit

# Link to the GitHub repository

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

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

Future application enhancements could include:

* Converting the application to a Web based user interface
* Adding file update functionality
* Adding the ability to create folders within directories
* Adding the ability to clone a directory and it’s contents
* Adding the ability to search for text across all files within a directory
* Adding registration and login functionality 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 data