**Requirements Specification**

**Functional requirements**

**Agent**

R1. FlyinTravel should store all the data about user login credentials.

R2. FlyinTravel should store all the flights information and user details in database.

R3. FlyinTravel should allow user’s for searching flights followed by search criteria.

R4. FlyinTravel should allow the Agent to update the flight database.

R5. FlyinTravel should allow Agent to make booking for flights available.

R6. FlyinTravel should allow Agent to modify and cancel the booking with respective booking reference ID.

**Traveler**

R7. FlyinTravel should allow the Traveler to create a personal account in the system.

R8. FlyinTravel should allow Traveler to manage personal account information.

R9. FlyinTravel should allow the Traveler to make a flight booking from selecting the search results.

R10. FlyinTravel should provide the Traveler different payment options for flight booking.

R11. FlyinTravel should allow the Traveler to modify the booking.

R12. FlyinTravel Should allow the Traveler to cancel the booking.

R13. FlyinTravel should allow Traveler to manage the bookings.

**Bank**

R14. FlyinTravel should provide access to the Bank for validating the payment made with Credit/Debit/Interac.

**Non-Functional Requirements**

**Performance Requirements**

R15. FlyinTravel should allow parallel processing for serving various user’s requests.

R16. FlyinTravel should be accessed from anywhere and at anytime.

**Usability Requirements**

R17. FlyinTravel is more user friendly and strive for effective user interaction by generating alerts and allowing user to avoid common errors.

R18. FlyinTravel doesn’t need any system training in order to use.

**Security Requirements**

R19. FlyinTravel should only provide access to Agent to update the flight database.

R20. FlyinTravel should be provide standard security for performing the payment transactions.

**Legal Requirements**

R21. FlyinTravel should satisfy with the certified quality assurance standards.

R22. FlyinTravel should protect and never disclose the user personal information.

**Use Case Descriptions**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Use Case name** | **General Description** |
| 1. | Update Flight DB | The Agent update the flights information in the database. |
| 2. | Create Account | The user can create a personal account in the system. |
| 3. | Login | The user can login into personal account with valid account credentials. |
| 4. | Search Flights | The user can search for the flights based on search criteria. |
| 5. | Flight Booking | The user can book a flight from the search results. |
| 6. | Make Payment | The Traveler needs to make payment for the booking in process. |
| 7. | Cancel Booking | The user can cancel the booking with the respective booking reference ID. |
| 8. | Modify Booking | The user can make modifications to the booking. |
| 9. | Earn Points | The Traveler with personal account will earn points for the miles travelled with the previous bookings. |
| 10. | Payment via points | The Traveler can redeem their points as a payment for the current booking. |
| 11. | Payment via Credit/Debit/Interac | The Traveler can make payment for the booking with a Credit/ Debit/ Interac transaction. |
| 12. | Validate Payment | The bank will validate the Credit /Debit /Interac details of the booking payment. |

|  |  |
| --- | --- |
| Use case ID | UC 001 |
| Name | Update\_Flight DB |
| Actor(s) | Agent |
| Description | The Agent will update the flights information in the database. |
| Preconditions | 1. The Agent needs to login into the system. 2. The Agent must get all the required flight information. |
| Basic Flow | 1. The Agent account provide the Update\_flight DB option for the Agent. 2. Upon selecting the option, the Agent should enter all the required information of the flights to be updated. |
| Post Conditions | Flight database updated successfully. |
| Exception | If the updated information is any match with other flights or mandatory fields of flight left blank, an error message will be displayed. |

|  |  |
| --- | --- |
| Use case ID | UC 002 |
| Name | Create Account |
| Actor(s) | Traveler |
| Description | The Traveler is facilitated to create a personal account in the system. |
| Preconditions | 1. The Traveler must not have any existed account in the system. 2. The Traveler should disclose the necessary information in the process of creating account. |
| Basic Flow | 1. The Traveler request the system for creating account. 2. The system redirects the Traveler to signup form page where the personal information should be disclosed. 3. The Traveler submit after entering the valid Username and Password as the credentials for the personal account 4. The system validates the Username, Password and the personal information with the existed accounts in database, and then store them in the database. |
| Post Conditions | Account Created successfully. |
| Exception | If any of the information provided by the Traveler existed in the system or any mandatory fields left blank, then an error message will be displayed. |

|  |  |
| --- | --- |
| Use case ID | UC 003 |
| Name | Login |
| Actor(s) | Traveler, Agent |
| Description | The User Login into the personal account with login credentials. |
| Preconditions | 1. The User should have an existed account in the system database. 2. The User need to provide the login credentials. |
| Basic Flow | 1. The User open the login page in the system. 2. The system requests the user to provide the login credentials. 3. The User then needs to enter the Username and Password and submit to the system. 4. The system then verify the credentials provided with the stored credentials in the database. 5. The User will given access into the personal account upon valid verification of the credentials. |
| Post Conditions | Login successful. |
| Exception | If the credentials are verified as not valid, an error message will be displayed and allow user to enter them again. |

|  |  |
| --- | --- |
| Use case ID | UC 004 |
| Name | Earn Points |
| Actor(s) | Traveler |
| Description | The Traveler will earn points based on the number of miles travelled with the previous bookings. |
| Preconditions | The Traveler must login into the personal account. |
| Basic Flow | 1. The Traveler made a successful booking with the system through logging into personal account. 2. The system calculates the miles involved in the journey of that booking. 3. The number of miles will be rewarded as points earned for the Traveler. |
| Post Conditions | Earned points will be displayed. |
| Exception | NA |