# *Software Requirement Specification (SRS) Document*

**For**

# FLY HIGH AIRLINE SYSTEM

*Version 1.0*

*Survey of Software Engineering(CS5391.251)*

03/30/2017

Prepared by:

Patrick Ring
Akhila Vemuganti
Aishwarya Rana
Kritika Karki

Table of Contents

# 1. Introduction

## 1.1 Purpose

The purpose of this Software Requirement Specification (SRS) document is to list specification and requirements for ***Fly high airlines*** website. From this document, the customer will be able to understand what the final product would entail. The audience of this SRS document would be the professionals developing this website and the clients who invested on this website.

## 1.2 Scope

The ***Fly high airlines*** system is a web-based system for the users which helps them to create an account, provide their information, search for the flights they want, and check them out by paying for the available tickets via Credit/Debit card or accumulated mileage. It should be globally available and free to access. ***Fly high airlines*** provides information about availability of flight tickets and their price based on the search criteria provided by the users. It requires active connection to perform the search and display the results to users, and also to complete the payment process after a flight choice has been made. It also has the capability of letting users update their account information, for instance updating their address, phone number or credit/debit card details.

## 1.3 References

[1] Pressman, Roger S. Software engineering: a practitioner's approach / Roger S. Pressman. — 7th ed. p. cm *New York: McGraw* Hill, 1953

[2] Software Engineering Standards Committee of the IEEE Computer Society (1998) Available at: http://www.cse.msu.edu/~cse870/IEEEXplore-SRS-template.pdf [02/ 02/ 2017].

## 1.4 Overview

The following sections of the SRS are organized as follows:

- Section 2 will give an overview of each of system's functionalities. This section also introduces different types of stakeholders and their interaction with the system.
- Section 3 will provide the specific requirement for each function of the airlines system with more technical details. It will also provide the non-functional requirements of the system.

# 2. Overall Description

This section will give an overview of the whole system. The system will be explained in its context to show the ways a customer can use the system and introduce the basic functionalities. It will also describe the type of users that can utilize the system and the functionalities available for each type. At the end, the constraints for the system will be presented.

## 2.1 Product perspective

***Fly high airlines*** system is not part of any larger system, however, it depends on external emailing system (to inform users about their account as well the flights) and credit card system (to help users make a payment). The system will interact with a database to store and access customer as well as flight information. The user will interact with the system via a website.

## 2.2 Product functions

*Fly high airlines* system has following functions:

### 2.2.1 Account Registration

#### 2.2.1.1 Description:

To set up an account the users should provide their first name, last name, mailing address, billing address, phone number, and email address. The users are given the freedom to pick their own password. On account creation, the users will be assigned a frequent flyer number which can be used by the users, in combination with the password, to access their accounts.

**Rationale:**

This function will allow the user to enroll in the website by creating an account.

### 2.2.2 Account Login

#### 2.2.2.1 Description:

The account login function shall allow registered users to enter their frequent flyer number and password. Once the frequent flyer number and password are verified, users will be able to access their account and perform desired action.

**Rationale:**

This function will authenticate the user. Authenticated users are allowed to access more functionalities than non-users.

### 2.2.3 Forgot Password

#### 2.2.3.1 Description:

This function will allow a user to reset password in case the user forgets it.

**Rationale:**

This function allows users to recover password. The system sends their respective frequent flyer number with new password to the user in their registered email address.

### 2.2.4 Flight Search

#### 2.2.4.1 Description:

The function will display a list view of the search results matching closely to the search made by the user with information such as “To”, “From”, “Departure Date”, “Arrival Date”, and “Number of passengers”. User can search flight either for one way or two way.

**Rationale:**

It allows users to search for specific flight(s) based on some search criteria.

### 2.2.5 Book Flights

#### 2.2.5.1 Description:

The book flights option shall allow users to temporarily save the flights in a placeholder that are being considered for purchase until finished purchasing. A page should display a list of flights along with the information about the ticket price, arrival and departure time, origin, destination, and the total amount that must be paid.

**Rationale:**

The function will allow user to book desired flight(s) if the tickets for the specific flight(s) are available.

### 2.2.6 Reserve Seats

#### 2.2.6.1 Description:

The reserve seats function will allow the users to select a seat in a preferred location inside the aircraft. A user must enter the booking confirmation number for a flight. A page should display the seats' layout inside the craft showing the seats that are available for selection and seats that are unavailable for reservation in that flight.

**Rationale:**

This function will allow those users who do not like the system assigned seats to reserve preferred seat(s). The user must be logged in the system to use this function.

### 2.2.7 Flight Status

#### 2.2.7.1 Description:

The flight status function will allow the user to enter a flight number and/or origin and destination airports along with the flight date to check the status of a desired flight. A page should display the flight number and name, scheduled and estimated departure time and arrival time along with status of the flight which can be either "On time", "At the Gate", "In flight" or "Delayed".

**Rationale:**

This function shall allow the users to check the status of a flight.

### 2.2.8 Flight Schedules

#### 2.2.8.1 Description:

The page shall provide the list of all available flights based on "To", "From", "Departure date", and "Departure time" information given by the user.

**Rationale:**

This function provides the details of all available flights based on the user's search criteria.

### 2.2.9 Checkout

#### 2.2.9.1 Description:

The checkout function shall allow registered users to select a payment method for the flight ticket. The users can pay either by their accumulated mileage if the number of miles for that user is greater than 25,000 or by using Credit/Debit card.

**Rationale:**

This function will allow logged in users to confirm the flight(s) on which they will be travelling and make payment for the ticket(s).

### 2.2.10 View Account

#### 2.2.10.1 Description:

The function shall allow users who have logged into the system to view their personal information as well as other information like Frequent Flyer number, Mileages accumulated, Booked flights, and Reserved seats.

**Rationale:**

This function lets the user to keep track of his/her personal details and details related to their travel history.

### 2.2.11 Update Account

#### 2.2.11.1 Description:

The function shall permit users to edit their personal details like first name, last name, email address, phone number, mailing address, billing address, and password.

**Rationale:**

This function lets the user to update/edit his/her personal details in case there has been some changes in the previously stored information.

### 2.2.12 Logout

#### 2.2.12.1 Description:

The Logout function will ask confirmation from the user whether they want to logout of the system. If the user confirms the user will be logged out of the system.

**Rationale:**

After the users have finished performing the desired task in the system, they can need not stay in the system.

### 2.2.13 Contact Us

#### 2.2.13.1 Description:

The contact us function shall display the system's office address, customer service telephone number, fax number, and email address to the users.

**Rationale:**
The function makes it easier for the users to contact the airlines system (if needed) by providing all the contact details at the same place.

#### 2.2.14 Help

**2.2.14.1 Description:**
The help function shall provide the user information on how to use different functionalities of the system as well as answer a list of questions relating to the problems frequently faced by users. The user can specify the functionality/problem they need help with.

**Rationale:**
The function will provide an overview of all the functions mentioned above and their implication on the website to the users.

## 2.3 User characteristics

There is one kind of user for the ***Fly high airlines*** system. The user is the customer (having no prior training) who should be able to use the system by accessing the pages in a step by step manner. The system is anticipated to be used at any time of the day.

## 2.4 Constraints

The application needs an internet connection and a web browser to run.

# 3. Specific Requirements

## 3.1 External Interface Requirements

### 3.1.1 System Interfaces

The following two system interfaces will be applicable:

1. A credit card processing system: The system will access the credit card processing system via its web services API.
2. An emailing system: The system will make use of emails to send users information like their Frequent Flyer Number, Booking Confirmation, Payment Confirmation, and new password.

### 3.1.2 Customer Interfaces

The system will enable customers to access the website via the Internet globally. There will be a web interface accessible to users before the account creation/login function. Functions like Contact Us, Flight Schedule, Flight Status, and Help can be accessible by the user without any registration.

### 3.1.3 Software Interfaces

The system requires the user to access the fly high website using a web browser.

## 3.2 Functional Requirements

### 3.2.1 Account Registration

**Description**: A user can enroll with the website by creating a personal account.
**Inputs**: The following information is required:

i. First Name
ii. Last Name
iii. Email Address
iv. Password with length of 6 characters with at least one number.
v. Confirm Password
vi. Date of Birth
vii. Verification Code: Verification code should be the same as shown on screen.
viii. Phone number.

The following inputs will be optional: Middle Initial, Address Line1 (mailing), Address Line2, State, Country, Zip code, and Billing address.
**Source**: All the data will be entered by the user.
**Output**: An account will be created for the user. User is assigned a frequent flyer number and user information is saved.
**Destination**: A new database entry will be made at the web server with all user account details.
**Requires**: None
**Pre-Condition(s)**: The user is presented with the web-interface to create a new account. The user is not currently registered with the website.
**Post-Condition(s):** New user account created on web-server and assigned a frequent flyer number.

### 3.2.2 Account Login

**Description**: The function allows the registered users to access their account information and utilize other user specific functions of the fly high airlines.
**Inputs**: The following information is required from the user:

i. Frequent Flyer Number: The users enters their assigned frequent flyer number.
ii. Password: The password should match the corresponding frequent flyer number for successful login.

**Source**: The user will provide the information needed for this function.
**Output**: The user will be logged in successfully once correct username and corresponding password is entered or if users enters incorrect password message such as “Invalid Username or Password” is displayed.
**Destination**: Error message will be displayed in the same login page.
**Requires**: The user needs to have a valid account.
**Pre-Condition(s)**: The user is displayed with the login page on the website.
**Post-Condition(s)**: The user should be logged in or fail to log in.

### 3.2.3 Forgot Password

**Description**: Function that allows the user to fetch the password in case the user forgets it.
**Inputs**: Email address on file.
**Source**: Database will be searched for the Frequent Flyer Number entered and if found, a new password will be sent to the associated email address.
**Output**: New password can be set-up.
**Destination**: User account table in the database on the web-server.
**Requires**: A valid email address used while creating the user account on the website.
**Pre-Condition(s)**: The user is displayed with the forgot password option on the web-interface.
**Post-Condition(s)**: The user can access the website with the new frequent flyer number and password sent to their corresponding email address.

### 3.2.4 Search Flight

**Description**: Function that allows the user to search for flights.
**Inputs**: The user should enter data into the following fields:

i. One-way or Two-way option
ii. To
iii. From
iv. Departure Date
v. Arrival Date (in case of two-way)

**Source**: Database will be searched for the flights matching the entered fields' value.
**Output**: A page should be displayed containing list of flights matching users' criteria.
**Destination**: The output of flight search need not be stored.
**Requires**: Either one-way or two-way option must be selected. Also requires valid dates and name of cities.
**Pre-Condition(s)**: The user need not be registered or logged in. A page should be displayed with the above-mentioned field where the user can input their search criteria.
**Post-Condition(s)**: A page showing results of the search is displayed.

### 3.2.5 Book Flights

**Description:** The function offers a way for users to book a desired flight.
**Inputs:** Information including To, From, dates and number of passenger will be collected from the user.
**Source:** The information is provided to the user by the website.
**Output:** The order is successfully placed by the user.
**Destination**: The order details are recorded and stored in the database on the web-server.
**Requires**: Only account holder can book and purchase the flights.
**Pre-Condition(s)**: The user needs to be registered to be able to place an order.
**Post-Condition(s)**: A booking order is placed by the user.

### 3.2.6 Reserve Seats

**Description:** The function offers a way for users to select seat of their preference and then reserve it.

**Inputs:** Booking number.
**Source:** The information is provided to the user by the website.
**Output:** The preferred seat is successfully reserved.
**Destination**: The reserved seat is recorded and stored by associating it with the booking information on the web-server.
**Requires**: Only account holder can reserve seats on the flight that they have booked, hence they must have an account and should be logged into their account.
**Pre-Condition(s)**: The user must first book a flight and must be logged in their account.
**Post-Condition(s)**: User desired seat is noted, retained and assigned to the user on the day of flight.

### 3.2.7 Flight Status

**Description:** The function offers a way for users to search for particular flights and check their status.
**Inputs:** Either flight number, if the user has book a flight or name of source city/airport, name of destination city/airport and date of the flight must be entered by the user on the webpage form.
**Source:** The information is provided to the user by the website.
**Output:** A list is displayed to the user showing a specific flight (given the flight number) or different flight flying from source city to destination city on the specified date with their flight status.
**Destination**: The output of this search need not be stored.
**Requires**: A valid flight number or valid name of cities/airport along with date greater than or equal to the current date.
**Pre-Condition(s)**: The user is presented with the form to enter relevant information to perform the search.
**Post-Condition(s)**: A page showing results of the search is displayed.

### 3.2.8 Flight Schedules

**Description:** The page shall provide the list of all available flights based on user's search criteria.
**Inputs:** The user should be able to enter data in the below fields:

i. To
ii. From
iii. Departure Date
iv. Departure Time

**Source**: The information will be provided by the users.
**Outputs:** The page should display the below information:

i. Flight number (and/or)
ii. To
iii. From
iv. Departure Date
v. Departure Time

vi. Arrival Date
vii. Arrival Time
viii. Seats Availability
ix. Option to select a particular flight.

**Destination**: The output of this search need not be stored.
**Requires**: Valid name of cities along with a valid departure date and time.
**Precondition**: A page should be displayed to allow users to enter search criteria.
**Post-condition:** The flight details based on the search criteria should be displayed.

### 3.2.9 Checkout

**Description**: This functionality provides the administrator with the orders they need to process and complete.
**Inputs**: The user will be providing payment method either by card or accumulated mileage and confirming email address for order confirmation.
**Source**: The information will be provided by both- the user and the web-server using database.
**Output**: A record of the order successfully placed is provided to the administrator.
**Destination**: A database entry will be made with the order placed and their invoice details.
**Requires**: It requires successful payment completion by user.
**Pre-Condition(s):** It requires that the user has successfully placed an order.
**Post-Condition(s):** Administrator receives an email for the orders.

### 3.2.10 View Account

**Description:** The user should be able to view the following details in the page.

i. Frequent Flyer number (read only)
ii. Mileages accumulated (read only)
iii. Reservations (read only)
iv. Personal Information: First Name, Middle Initial, Last Name, Date of birth, Address Line1 (mailing), Address Line2, State, Country, Zip code, Billing address, Phone number, Email Address, Password.

**Inputs:** The user should be able to view the personal information fields.
**Source**: The user should provide the data needed to update the fields.
**Output:** Once the user updates the fields, the data should be saved and displayed on the user account.
**Destination**: None
**Requires**: None
**Pre-Condition(s)**: The account information page is displayed with the fields populated with data.
**Post-Condition(s)**: If the user updates any information, the changes will be reflected to user's' account.

### 3.2.11 Update Account

**Description:** The user should be able to view and update the following details in the page.

i. Personal Information: First Name, Middle Initial, Last Name, Date of birth, Address Line1 (mailing), Address Line2, State, Country, Zip code, Billing address, Phone number, Email Address, Password.

**Inputs:** The user should be able to update the personal information fields.
**Source**: The user should provide the data needed to update the fields.
**Output:** Once the user updates the fields, the data should be saved and displayed on the user account.
**Destination**: None
**Requires**: None

**Pre-Condition(s)**: The account information page is displayed with the fields populated with data and in editable format.
**Post-Condition(s)**: Once updated, the changes will be reflected to user's' account.

### 3.2.12 Account Logout

**Description**: The function allows the registered users to restrict access to their account from their current location and view the default pages.
**Inputs**: The following information is required from the user:

i. Request to logout
ii. Accept logout confirmation

**Source**: The user will provide the information needed for this function.
**Output**: The user will get a confirmation of successful logout once logout is complete
**Destination**: The user will be sent to the login screen.
**Requires**: The user to have a valid account.
**Pre-Condition(s)**: The user is logged into an account.
**Post-Condition(s)**: The user should be logged out.

### 3.2.13 Contact Us

**Description**: The function allows the users to view the Contact details of the airlines.
**Description**: The function allows the users to view the Contact details of the airlines.
**Inputs**: None
**Source**: None
**Output**: The user will view the email address and customer service number details of the airlines.
**Destination**: None
**Requires**: None
**Pre-Condition(s)**: None
**Post-Condition(s)**: None

#### 3.2.14 Help

**Description**: The function allows the users to view the FAQs.
**Description**: The function allows the users to view the Contact details of the airlines.
**Inputs**: None
**Source**: None
**Output**: The user will view the frequently asked questions related to reservations of the airlines.
**Destination**: None
**Requires**: None
**Pre-Condition(s)**: None
**Post-Condition(s)**: None

## 3.3 Non-Functional Requirements

### 3.3.1 Performance

The System is accessible from any web browser. The page load time and process time depends on the internet connection speed and computer systems of the user. Searches are expected to complete within 7 seconds. The number of users across different computer systems is not expected to affect the page load time and processing time.

### 3.3.2 Reliability

The average failure time should be 2 days. In the event of system crash, a backup system will be up within an hour.

### 3.3.3 Availability

The website is available to users 24 hours a day. In case of maintenance, the system should be made available again in no more than 4 hours.

### 3.3.4 Security

- To access personal information or make any changes, users must log into the system.
- Passwords and Credit Card information should be encrypted to prevent theft.

### 3.3.5 Usability

- The system interface should be easy to understand and need not require more than 15 minutes' worth of tutorial to use.
- The user should not need to consult the tutorial every time they use the system interface.
- The users should be able to use the functions provided by the system website with no or max 3 errors.

### 3.3.6 Software Quality Attributes

#### 3.3.6.1 Maintainability

All the system updates should be on system side and users should not be required to perform any maintenance activity.

#### 3.3.6.2 Robustness

The system should run without any failure or crash for 340 days a year. External system crashes should not cause the crash of airlines system.