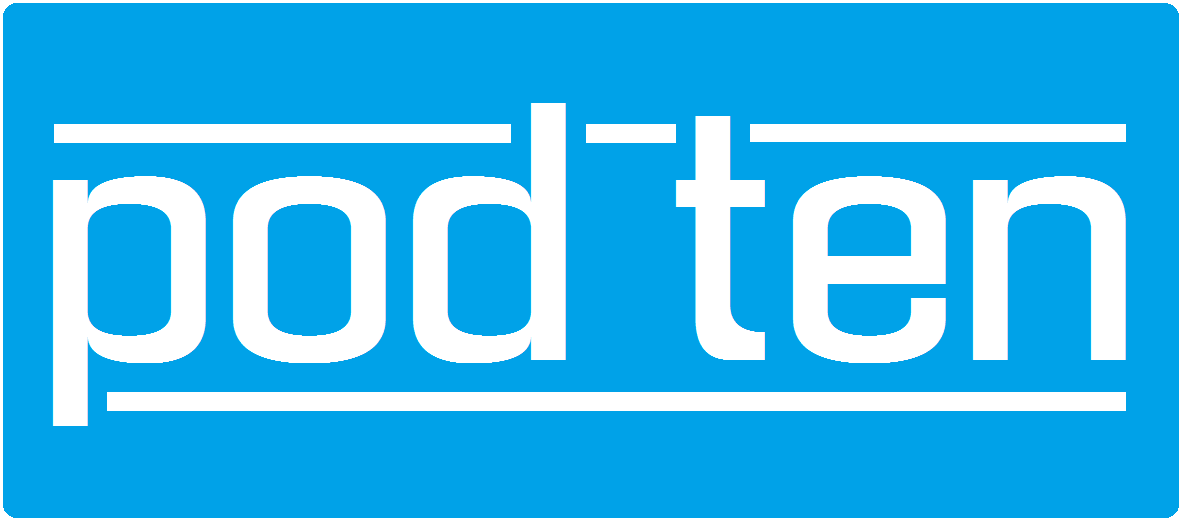
**AirPlan – Online Flight Booking System**

AirPlan Software Requirements Document

EECE 419 Assignment #2

Document Ref.: EECE419-Pod-10

Version 1.0- September 25, 2012



**CIRCULATION / DISTRIBUTION LIST**

|  |  |
| --- | --- |
| **NAME** | **STUDENT#** |
| Anuj Mehta | 23610082 |
| Ryan Randhawa | 62768080 |
| Kevin Chow | 44692093 |
| Nathaniel Sham | 65635088 |
| Yu Han | 43088087 |
| Louie Yuen | 75658096 |
| Yonathan Indrajaya | 42087072 |
| Scott Pearson | 43821081 |
| Shirley Gong | 63769087 |

**DOCUMENT IDENTIFICATION SHEET**

|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
| Date of first issue: 22/09/2012 | | Current version number : Version 1.0 | |
|  | | | |
| Prepared by : | EECE 419 Pod 10 | | Date : 21/09/2012 |
|  |  | |  |
| Verified by : | Yu Han | | Date : 22/09/2012 |
|  |  | |  |
| Approved by : | Anuj Mehta | | Date : 25/09/2012 |
|  |  | |  |
| Status : | IN PROGRESS | | Applicability Date: 25/09/2012 |
|  | | | |

**Copyright notice**

This document may not be reproduced (even partially) or communicated to third parties without the written authorization of Pod 10 Project Management.

DOCUMENT CHANGE RECORD

|  |  |  |
| --- | --- | --- |
| **Date** | **Description** | **Affected sections** |
| 22/09/2012 | First Issue | All |
| 25/09/2012 | Version 1.0 | All |
| 25/11/2012 | Version 2.0 | All |

Contents

1. INTRODUCTION 2

1.1. Purpose 2

1.2. Scope 2

1.3. References 2

2. SPECIFIC REQUIREMENTS 3

2.1. General Description 3

2.2. UML Diagram 4

2.3. Functional Requirements 5

2.4. Non-Functional Requirements 6

3. USE CASES 8

3.1. Customer Use Cases 8

3.2. Airline staff and management use cases 14

4. APPENDIX 18

4.1. Software and Hardware Requirements 18

4.2. User Interfaces Sketches and Box 19

# INTRODUCTION

## Purpose

The purpose of this document is to describe the requirements and specifications of AirPlan- a web based flight reservation and management system. It explains the functional features of the system, discusses the details of the user interface, design constraints and related constraints. This document is intended to be used as a guideline to designers, developers and testers that are responsible for developing the AirPlan system.

## Scope

The scope of this document is to discuss the required features of the AirPlan flight reservation and management system. It introduces the reader to the system by providing its general description and a few user interface sketches. A UML diagram is also provided to present a visual representation of the use cases for each type of user.

The document then discusses the functional and non-functional requirements of the system. It goes on to describe the possible use cases for each type of user- passengers, airline employees and airline management. The document delves into the details of each use case by explaining its flow of events, its expected outcome, and its pre and post conditions. The use cases are fairly comprehensive, as they provide step by step flows for user behaviour and explain how each possible scenario can be handled.

## References

Google. (s.d.). *Chrome system requirements*. Retrieved le September 23, 2012, from Google Support:

http://support.google.com/chrome/bin/answer.py?hl=en&answer=95411

# SPECIFIC REQUIREMENTS

## General Description

AirPlan is a flight reservation and management system that allows customers to search for and reserve flight tickets through an intuitive web interface and airline staff and management an easy to use flight management tool. The system will allow passengers to search for flights, reserve seats and buy tickets, select flight preferences, and view boarding passes. It will also allow passengers to modify reservations and manage loyalty points. The airline staff and management will be able to view the manifest of each flight along with each passenger’s seating location and flight preferences. The management will also have access to statistical information such as the percentage of flights delay, the percentage of flights that were full, the percentage of flights that were more than half empty, seat availability information and destination specific information. Overall, AirPlan aims to be a one stop shop for all flight reservation and management issues.

## UML Diagram

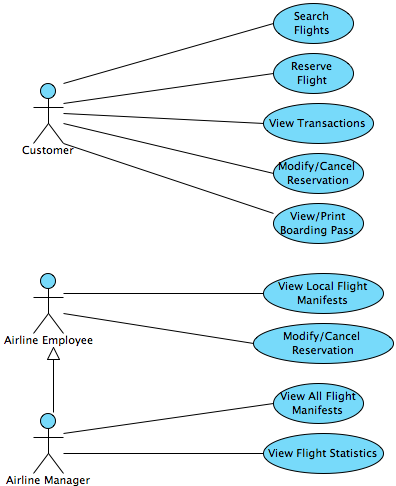


Figure 1: UML User Diagram

## Functional Requirements

|  |  |
| --- | --- |
| **SR. Id.** | **Software requirements description** |
|  | **Customer Requirements** |
| F.000 | The customer shall be able to search for a flight or a set of flights by inputting required information such as from and to locations and date of travel, and optional information such as return flight date. |
| F.001 | The customer shall be able to make flight reservations of one or more flights using either their credit cards or loyalty points. |
| F.002 | The customer shall be able to manage payments by viewing all of their past transactions and checking the status of their loyalty points. |
| F.003 | The customer shall be able to logon to modify tickets for their upcoming flights. The customer shall be able to update their ticket preferences such as seat location and meal type. |
| F.004 | The customer shall be able to logon to view and print the boarding passes associated with bookings. |
| F.005 | The customer shall be able to cancel a booking and all of the tickets associated with the booking. |
|  | **Airline Staff and Management Requirements** |
| F.100 | The airline staff shall be able to view the manifest for each flight that they have access to, including seat locations and in flight preferences for each passenger. |
| F.101 | The airline management should be able to view the flight manifests of each flight, including seat locations and in flight preferences for each passenger. |
| F.102 | The airline staff and management shall be able to help customers modify their tickets for upcoming flights. |
| F.103 | The airline staff and management shall be able to help customers cancel their reservations for upcoming flights. |
| F.104 | The airline management shall be able to view real time statistics for each flight. |

## Non-Functional Requirements

### Security requirements

|  |  |
| --- | --- |
| **SR. Id.** | **Software requirements description** |
| SEC.0 | The system shall store all user data in secure data centers and no unauthorized access shall be granted to anyone outside or within the company.   * User data shall be stored in external databases which shall only be accessible to authorized team members. * Only data that is absolutely essential shall be requested of the user. * User data shall not be sold to external parties and shall not be used to display targeted advertisements. |

### Performance requirements

|  |  |
| --- | --- |
| **SR. Id.** | **Software requirements description** |
| PER.0 | No user action shall have a response time of more than 1 second.   * Users shall be notified in advance if an action is known to have large processing times. |

### Interface requirements

|  |  |
| --- | --- |
| **SR. Id.** | **Software requirements description** |
| INT.0 | The product shall be web based and platform independent.   * The product shall be accessed via a web browser such as Google Chrome on any platform. This includes but is not limited to Windows, OSX and Linux. |

### Operational requirements

|  |  |
| --- | --- |
| **SR. Id.** | **Software requirements description** |
| OPE.0 | All significant transactions initiated by the user shall be logged.   * Each log shall include user id, timestamp and a verbal explanation of the action. Logs may also include prior values and new values of objects. * Read and write access to production logs shall only be given to authorized users. * Logs shall be verbose enough to support investigation of bugs. * Logs shall be written for any action that involves an update to the database. These actions include but are not limited to flight reservations, loyalty point updates, payments, modifications to reservations etc. |
| OPE.1 | The consumer as well as airline employee facing applications will have 99.9% uptime between 6:00AM and 1:00AM Pacific Standard Time.   * Maintenance shall only be done between 1:00am and 6:00am Pacific Standard Time, and visitors to the website shall be notified of this maintenance at least 24 hours in advance. * The integrity of each transaction shall be protected with mechanisms such as fallback and retry etc. |

# USE CASES

## Customer Use Cases

### Use Case 1 – Search Flight

#### Description

This use case describes how a customer can search for flights over the Internet.

#### Pre-Conditions

Customer has created a valid account.

Customer has logged into the system.

Customer’s identity has been verified.

Customer’s current login session is valid.

#### Post-Conditions

**Successful Post-Condition**:

Customer has found the most suitable flights for his/her trip.

**Failed Post-Condition**:

Customer has failed to find a suitable flight because the location is out of scope.

#### Flow of Events

1. Customer selects a starting location.
2. Customer selects a destination.
3. Customer selects a date.
4. Customer selects a return flight date (optional).
5. Customer selects number of passengers.
6. The system shows a list of flight choices with available seats more than customer’s number of passengers.
7. Customer selects the flight he/she is interested in.
8. The system shows more detail about that flight.
9. End use-case.

#### Special Requirements

Customer can stop the search and go back to the previous step at any time.

#### Extension Points

None.

### Use Case 2 – Reserve Flight

#### Description

This use case describes how a customer can reserve flights over the Internet.

#### Pre-Conditions

Customer has created a valid account.

Customer has logged into the system.

Customer’s identity has been verified.

Customer’s current login session is valid.

Customer has found the flights he/she wishes to reserve.

#### Post-Conditions

**Successful Post-Conditions**:

Customer has reserved the flights with preferred class.

Customer has selected preferred seat option on the plane.

Customer has selected a preferred meal plan.

Customer has paid for the booking (using credit cards and/or loyalty points).

**Failed Post-Condition:**

Customer has failed to book a ticket because all tickets are sold out.

#### Flow of Events

1. Customer chooses a flight or several flights and proceeds to booking.
2. Customer enters passenger’s name
3. Customer selects a seat preference.
4. Customer selects a meal plan preference.
5. Customer finalizes the booking.
6. The system reserves a seat on the flight.
7. The system displays an invoice to the customer and redirects the customer to billing module.
8. The system displays payment information and payment method.

*The system disable the option of paying by loyalty points if not enough points under the user’s account, otherwise, displays the available loyalty points for customer to use.*

1. Customer selects payment method as credit cards and enters credit card information.
2. Customer completes the payment transaction.
3. The system updates the booking as paid, books the seats, and makes the ticket available to the customer.

*The system updates the manifest of the flight and meal preference list.*

1. End use-case.

Alternative A:

1. Customer selects payment method as paying by loyalty points.
2. The system deducts the amount from the customer’s loyalty points balance.
3. The system notifies the customer that no point is given and proceeds to final confirmation.
4. End use-case.

#### Special Requirements

Customer can stop the booking process at any time before it is confirmed (step 10).

No changes can be made to the manifest until the customer makes the booking (step 10).

No loyalty points are issued at this stage (loyalty points are given after customer prints out the boarding pass).

#### Extension Points

None.

### Use Case 3 – View Transaction History

#### Description

This use case describes how customers can view all of their past transactions and check their loyalty point balance.

#### Pre-Conditions

Customer has created a valid account.

Customer has logged into the system.

Customer’s identity has been verified.

Customer has made reservations in the past.

#### Post-Conditions

Customer has viewed his/her past transactions.

Customer has viewed his/her loyalty points.

#### Flow of Events

1. Customer selects view past transactions.
2. The system displays a list of transactions
3. The system displays the most updated balance of loyalty points.

*Pending loyalty points are not included in the balance (pending loyalty points are points gained from flights that the customer has paid for but has not printed the boarding pass for).*

#### Special Requirements

None.

#### Extension Points

None.

### Use Case 4 – Modify Booking

#### Description

This use case describes how a customer can modify tickets associated with their booking over the Internet.

#### Pre-Conditions

Customer has created a valid account.

Customer has logged into the system.

Customer’s identity has been verified.

Customer has reserved a ticket and has a confirmed booking.

#### Post-Conditions

**Successful Post-Conditions**:

Customer has modified ticket for their booking.

The system has updated the information correctly.

**Failed Post-Condition:**

Customer has failed to modify a ticket for their booking because the boarding pass has already been printed.

#### Flow of Events

1. Customer selects a booking he/she has made.
2. Customer makes changes to a ticket for that booking.
3. Customer confirms and saves the modification.
4. The system updates the information.

*The system updates the manifest of the flight and meal preference list.*

1. End use-case.

#### Special Requirements

Customer can stop the modification at any step before confirmation (step 3).

#### Extension Points

None.

### Use Case 5 – Cancel Booking

#### Description

This use case describes how a customer can cancel an entire booking over the Internet.

#### Pre-Conditions

Customer has created a valid account.

Customer has logged into the system.

Customer’s identity has been verified.

Customer has made a booking.

The boarding pass for the booking has not been printed.

#### Post-Conditions

**Successful Post-Conditions**:

Customer has cancelled his/her booking.

Each ticket for that booking has been cancelled.

The system has updated the information correctly.

**Failed Post-Condition:**

Customer has failed to cancel his/her booking because the boarding pass has already been printed.

Some tickets for the booking remain valid.

#### Flow of Events

1. Customer selects a booking they have made.
2. Customer cancels the booking.
3. Customer confirms the cancellation.
4. The system updates the information.

*The system updates the manifest of the flight and issues a refund.*

1. End use-case.

#### Special Requirements

Customer can stop the cancellation at any step before confirmation (step 3).

#### Extension Points

None.

### Use Case 6 – View and Print Boarding Passes

#### Description

This use case describes how a customer can view and/or print his/her boarding pass for the purchased flight tickets.

#### Pre-Conditions

Customer has created a valid account.

Customer has logged into the system.

Customer’s identity has been verified.

Customer has paid for his/her flight reservation.

#### Post-Conditions

Customer has confirmed the purchase.

Customer has viewed his/her boarding pass.

Customer has printed his/her boarding pass.  
Customer has been granted any associated points for that booking.   
Customer may no longer modify or cancel his ticket or booking.

#### Flow of Events

1. Customer logs in.
2. Customer goes to the Manage Bookings page.
3. Customer selects a ticket he/she has purchased.
4. Customer views the boarding pass.
5. Customer prints the boarding pass.
6. The system records the transaction.

*Customer will not be able to make any changes to this reservation.*

1. The system updates the customer’s loyalty points balance.

*Loyalty points are only granted for the segment paid by credit cards.*

1. End use-case.

#### Special Requirements

Customer should not be able to make modifications or cancel the booking once a boarding pass has been printed.

#### Extension Points

None.

## Airline staff and management use cases

### Use Case 1 – View Flight Manifest (Airline Staff)

#### Description

This use case describes how airline staff can view a flight manifest.

#### Pre-Conditions

User has logged in using an account with staff -level privileges.

#### Post-Conditions

Flight manifest is shown for selected flight.

#### Flow of Events

1. User selects view flight manifest option from menu.
2. User is presented with a list of flight manifest available to user.
3. User clicks on one of the flights listed.
4. The system presents manifest of the selected flight.
5. End use case.

Alternative flow A

1. User clicks on individual seat with special request/meal preference.
2. The system pops out a window containing the individual information.
3. End use case.

#### Special Requirements

User can go back to previous steps at any time.

#### Extension Points

None.

### Use Case 2 – View Flight Manifest (Airline Managers)

#### Description

This use case describes how airline managers can view a flight manifest.

#### Pre-Conditions

User has logged in using an account with manager-level privileges.

#### Post-Conditions

**Successful Post-Condition**:

Flight manifest is shown for selected flight.

**Failed Post-Condition:**

Error message is shown to the user.

#### Flow of Events

1. User selects view flight manifest option from menu.
2. The system displays a search criteria to find the desired flight manifest.

*Criteria includes flight code, date (range), flight origin/destination.*

1. User enters the search criteria.
2. System displays list of matching flight based on search criteria.
3. User selects one of the flights listed.
4. The system presents manifest of the selected flight.
5. End use case.

Alternative flow A

4. System cannot find matching flights based on search criteria. User is prompted to enter different search criteria.

5. End use case with failure condition.

#### Special Requirements

User can modify selection and/or go back to previous steps at any time.

#### Extension Points

None.

### Use Case 3 – View Flight Statistics

#### Description

This use case describes how airline managers can view flight statistics.

#### Pre-Conditions

User has logged in using an account with manager-level privileges.

#### Post-Conditions

**Successful Post-Conditions**:

User is presented with flight statistics.

**Failed Post-Condition:**

An error message is shown to the user.

#### Flow of Events

1. User selects to view flight statistics option from the menu.
2. User selects types of statistics to be shown.

*Types: Historical analysis and real-time view of flights.*

1. System processes user selection from database, “Please wait” message is shown to user.
2. Statistics of the selected type is shown to the user.
3. End use case.

Alternative flow A

1. There is no response from database within 1 minute. System will re-try up to 3 times.
2. If there is still no response, display message “Data unavailable at this time – please retry later”.
3. End use case.

#### Special Requirements

User can modify selection and/or go back to previous steps at any time.

#### Extension Points

None.

### Use Case 4 – Modify Ticket Reservation on Behalf of Customer

#### Description

This use case describes how airline staff and airline management can modify ticket reservation on behalf of customers.

#### Pre-Conditions

User has logged in using an account with staff or manager-level privileges.

User has been provided with customer’s email address.

Customer has a booking with one or more tickets.

#### Post-Conditions

**Successful Post-Conditions**:

Ticket for the booking was successfully modified.

System updates related information properly.

**Failed Post-Condition:**

Modification failed since the customer has already printed the boarding pass for the booking.

#### Flow of Events

1. User selects customer service option from menu.
2. User enters customer’s email address.
3. System displays reservation details associated with customer’s email address.
4. User selects a ticket to modify.
5. User confirms and saves the modification.
6. The system updates information in database.
7. End use case.

Alternative flow A

1. System notifies user that no modification can be made since customer has printed a boarding pass.
2. End use case with failure condition.

Alternative flow B

3. System cannot find matching reservation associated with customer’s email address.

4. Use case resumes at step 2.

#### Special Requirements

User can modify selection and/or go back to previous steps before confirmation.

#### Extension Points

None.

### Use Case 5 – Cancel Booking on Behalf of Customer

#### Description

This use case describes how airline staff and airline management can cancel bookings and all associated tickets on behalf of customers.

#### Pre-Conditions

User has logged in using an account with staff or manager-level privileges.

User has been provided with customer’s name and/or confirmation number.

Customer has one or more valid bookings.

#### Post-Conditions

**Successful Post-Conditions**:

Flight booking was successfully cancelled.

All tickets associated with the booking were cancelled.

System updates related information properly.

**Failed Post-Condition:**

Cancellation failed since the customer has already printed a boarding pass.

#### Flow of Events

1. User selects customer service option from menu.
2. User enters customer name and/or booking confirmation number.
3. System displays reservation details associated with customer name or confirmation number.
4. User modifies/cancel the reservation details.
5. User confirms and saves the modification/cancellation.
6. The system updates information in database.
7. End use case.

Alternative flow A

1. System notifies user that no cancellation can be made since customer has printed the boarding pass.
2. End use case with failure condition.

Alternative flow B

3. System cannot find matching reservation associated with customer name or confirmation number.

4. Use case resumes at step 2.

#### Special Requirements

User can modify selection and/or go back to previous steps before confirmation.

#### Extension Points

None.

# APPENDIX

## Software and Hardware Requirements

Computers running AirPlan should meet the following hardware and software specifications in order to use the software as intended:

* Google Chrome version 21 or higher
* 250 MB free disk space
* 512 MB RAM
* An AirPlan account (free to create)
* A valid email address (for all correspondence)
* Access to a printer (for boarding passes)

In addition, computers should meet the following minimum operating system and processor requirements:

|  |  |
| --- | --- |
| **Minimum Operating System Version** | **Minimum Processor** |
| Windows XP Service Pack 2 | Intel Pentium 4 or later |
| Mac OS X 10.5.6 | Intel Processor |
| Ubuntu 10.04+  Debian 6+  Fedora Linux 14+ | Intel Pentium 3 or AMD Athlon 64 |

## User Interfaces Sketches and Box

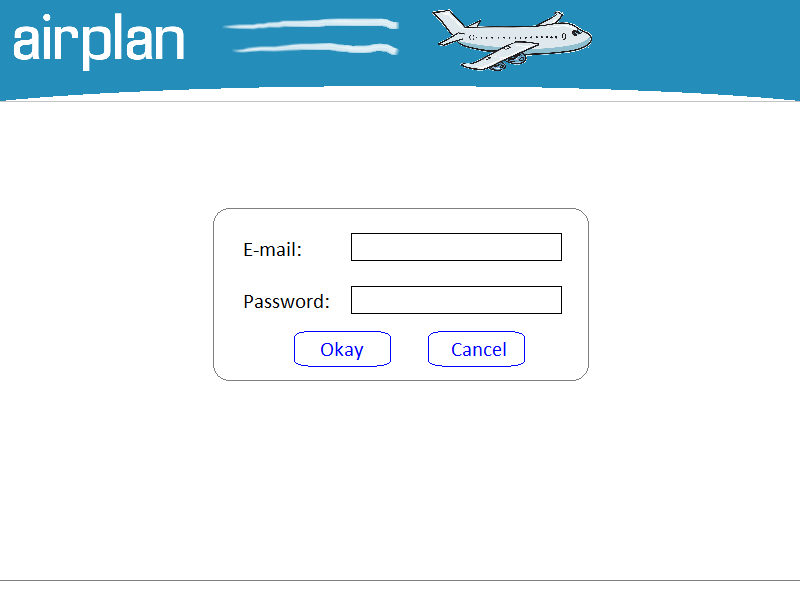
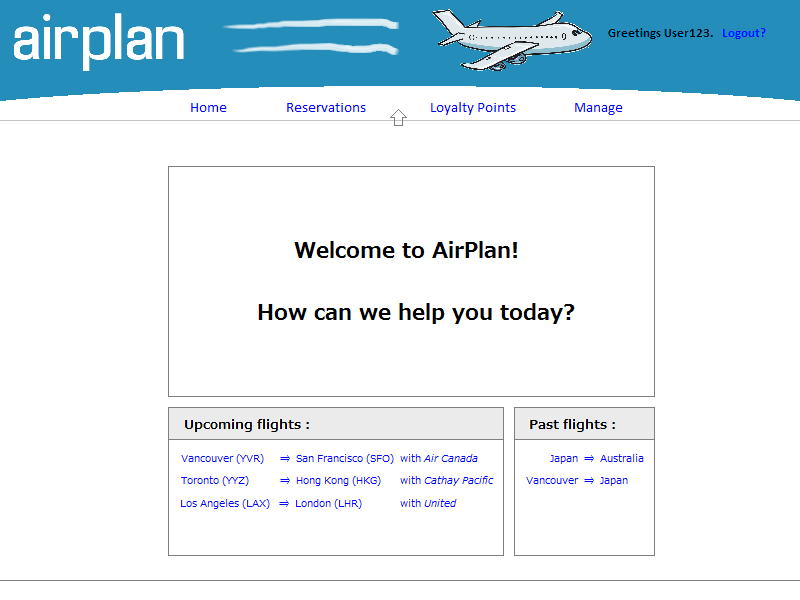
 

Figure 2: Login Page Figure 3: Home Page

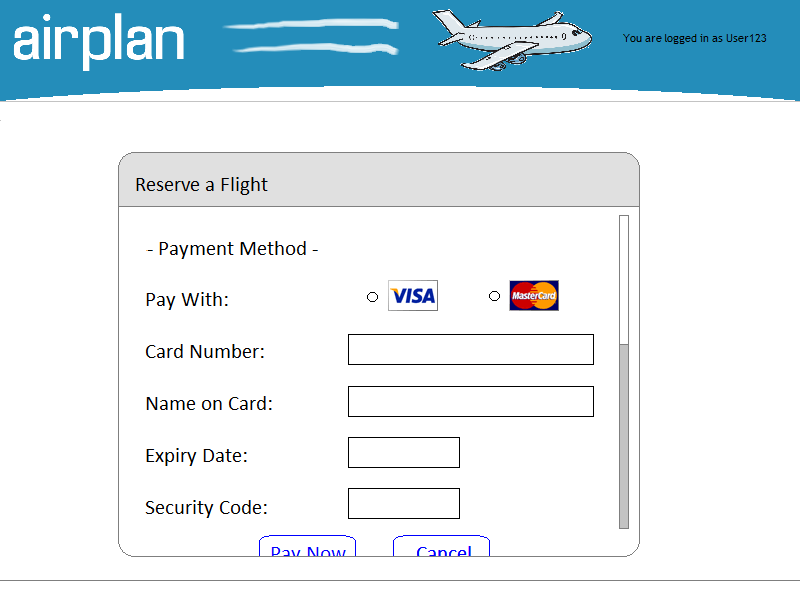
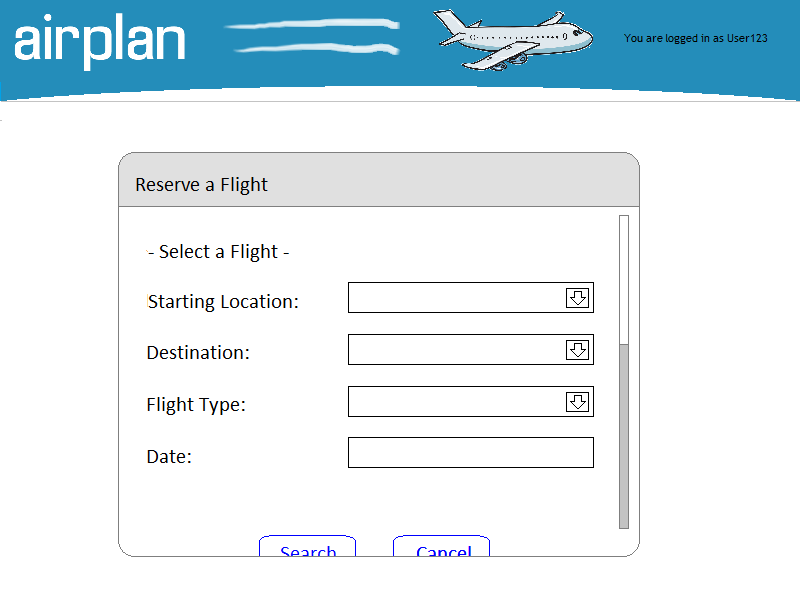
 

Figure 4: Payment Page Figure 5: Flight Search Page

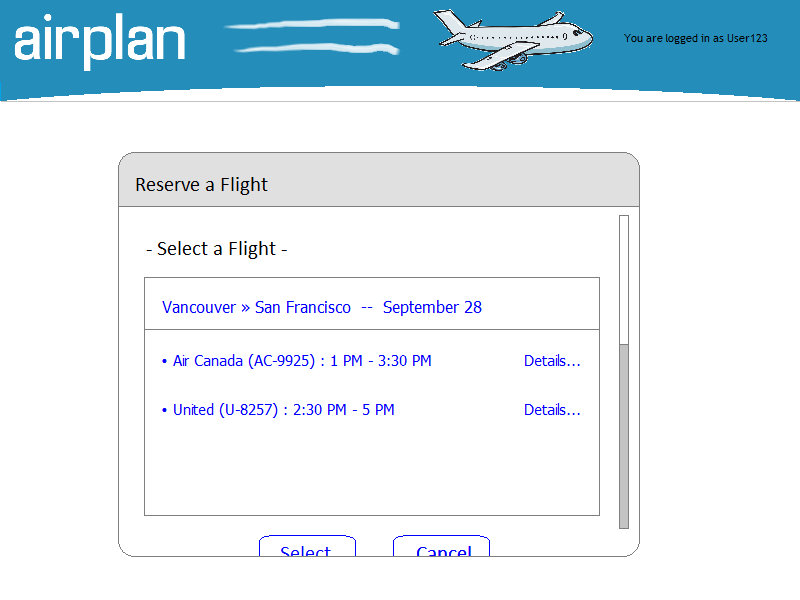
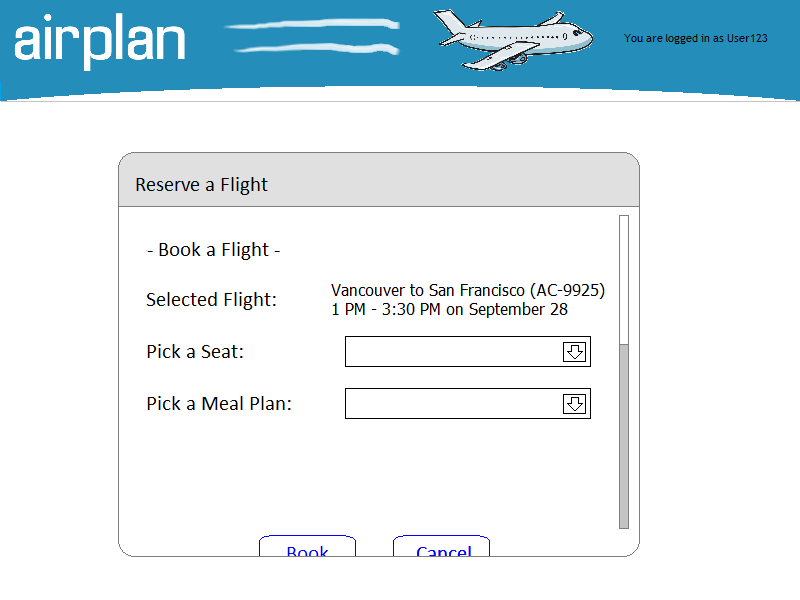
 

Figure 6: Flight Selection Page Figure 7: Reservation Details Page

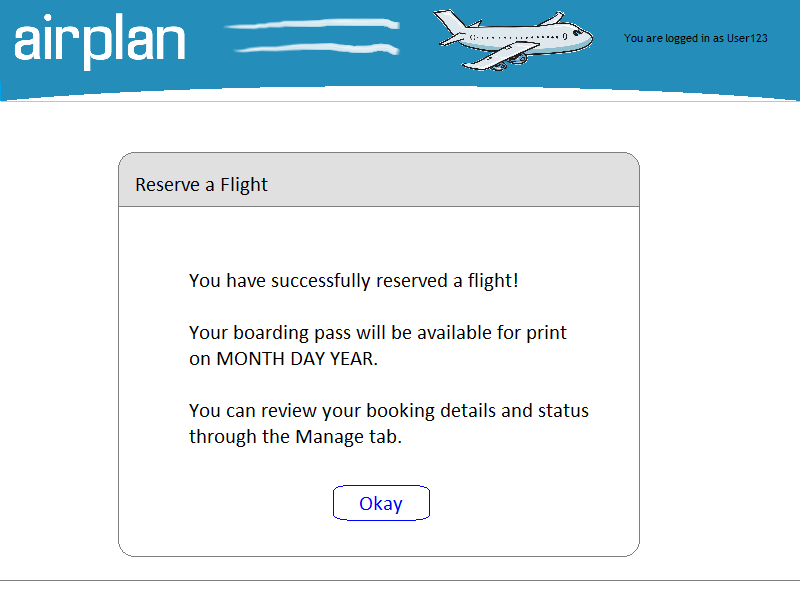


Figure 8: Reservation Confirmation Page