**Happy Trip**

**Software Requirements Specification**

**Version 1.3**

**Produced by:**

**Pratian Technologies (India) Pvt. Ltd**

**Bangalore**

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 04 May 2013 | 1.0 | Initial Draft | Priyanka Grover |
| 13-May-2013 | 1.1 | Changes incorporated as per review by Jerry & Ayaskant | Priyanka Grover, Jerry Kurian |
| 14-May-2013 | 1.2 | Updated Database Diagram & Use case Diagrams | Priyanka Grover, Nitesh |
| 19-May-2013 | 1.3 | Updated as per Review | Priyanka Grover |

Table of Contents

[Revision History 2](#_Toc356390527)

[Table of Contents 3](#_Toc356390528)

[1 Introduction 5](#_Toc356390529)

[1.1 Definition 5](#_Toc356390530)

[1.2 Specification Objectives 5](#_Toc356390531)

[1.3 Intended Audiences 5](#_Toc356390532)

[1.4 References 5](#_Toc356390533)

[1.5 Specification Overview 6](#_Toc356390534)

[2 System Overview 7](#_Toc356390535)

[2.1 System Usage 7](#_Toc356390536)

[2.2 Context 8](#_Toc356390537)

[2.2.1 Human Actors 8](#_Toc356390538)

[2.2.2 External Data Repositories 9](#_Toc356390539)

[2.2.3 External Hardware 9](#_Toc356390540)

[2.2.4 External Networks 9](#_Toc356390541)

[2.2.5 External Systems 9](#_Toc356390542)

[3 System Operational Requirements 10](#_Toc356390543)

[3.1 Use Case Diagrams 11](#_Toc356390544)

[3.1.1 HappyTrip System diagram 11](#_Toc356390545)

[3.1.2 User Module 11](#_Toc356390546)

[3.1.3 Admin Module 13](#_Toc356390547)

[3.2 Functional Requirements: 13](#_Toc356390548)

[3.2.1 User Module 13](#_Toc356390549)

[3.2.1.1 Registration 13](#_Toc356390550)

[3.2.1.2 Login 14](#_Toc356390551)

[3.2.1.3 Profile 14](#_Toc356390552)

[3.2.1.4 Book Ticket – [Both registered and non-registered users] 15](#_Toc356390553)

[3.2.1.5 Cancellation 19](#_Toc356390554)

[3.2.1.6 My Trips (Booking History) 19](#_Toc356390555)

[3.2.2 Admin Module 20](#_Toc356390556)

[3.2.2.1 Maintain Airline information 20](#_Toc356390557)

[3.2.2.2 Add Flight information 21](#_Toc356390558)

[3.2.2.3 Maintain City Information 21](#_Toc356390559)

[3.2.2.4 Add Route Information 22](#_Toc356390560)

[3.2.2.5 Add Schedule Information 23](#_Toc356390561)

[3.2.2.6 View Day’s Inventory 23](#_Toc356390562)

[3.2.2.7 Error Handling 24](#_Toc356390563)

[3.2.2.8 24](#_Toc356390564)

[3.2.3 GUI Requirements 25](#_Toc356390565)

[3.2.3.1 Home Page 25](#_Toc356390566)

[3.2.3.2 Search Results 25](#_Toc356390567)

[3.2.3.3 Confirm Booking 26](#_Toc356390568)

[3.2.3.4 Passenger Details 26](#_Toc356390569)

[3.2.3.5 Payment 27](#_Toc356390570)

[3.2.3.6 Payment confirmation - E-Ticket 27](#_Toc356390571)

[3.2.3.7 Registration 28](#_Toc356390572)

[3.2.3.8 Sign in 28](#_Toc356390573)

[3.2.3.9 View/Edit Profile 29](#_Toc356390574)

[3.2.3.10 My Trips 29](#_Toc356390575)

[3.2.3.11 Cancel – Confirm Box 30](#_Toc356390576)

[3.2.3.12 Cancel Success Page 30](#_Toc356390577)

[3.2.3.13 Admin – Login 31](#_Toc356390578)

[3.2.3.14 Admin – Add Airline 31](#_Toc356390579)

[3.2.3.15 Admin – Add Flight 32](#_Toc356390580)

[3.2.3.16 Admin – Add City 32](#_Toc356390581)

[3.2.3.17 Admin – Add Route 33](#_Toc356390582)

[3.2.3.18 Admin – View/Edit Airline 33](#_Toc356390583)

[3.2.3.19 Admin – View/Edit City 34](#_Toc356390584)

[3.2.3.20 Admin – Add Schedule 34](#_Toc356390585)

[3.2.3.21 Admin – View Inventory 35](#_Toc356390586)

[3.2.3.22 Admin – Successful Add 35](#_Toc356390587)

[3.2.3.23 Admin – Failure in Add 36](#_Toc356390588)

[3.2.3.24 Page Validations 36](#_Toc356390589)

[3.2.4 Non Functional Requirements 37](#_Toc356390590)

[3.2.4.1 Browser compatibility 37](#_Toc356390591)

[3.2.4.2 Performance Requirements 37](#_Toc356390592)

[3.2.4.3 Session Management & Security 37](#_Toc356390593)

[3.2.4.4 Usability 37](#_Toc356390594)

[4 Architecture and Design 38](#_Toc356390595)

[5 Database Design 40](#_Toc356390596)

[6 Known Issues 41](#_Toc356390597)

[7 Future Enhancements 42](#_Toc356390598)

[8 References 43](#_Toc356390599)

**Table of Figures**

[Figure 1: Happy TripContext Diagram 7](#_Toc352064669)

[Figure 2: m TripHigh Level Design Diagram 9](#_Toc352064670)

# Introduction

The Software Requirements Specification (SRS) for the application Happy Tripis intended to detail out the requirements design & implementation approach outlined for the same.

## Definition

This document will be referred as the absolute final requirements specification and it is this requirements work product that formally specifies the requirements of the Happy Trip.

## Specification Objectives

This software requirements specification has the following objectives:

* To provide an overview of the application’s context and capabilities.
* To formally specify the associated:
* Operational requirements
* Informational requirements
* Architecture and Design
* To document any future planned enhancements
* To document any open issues, major things to be completed, and assumptions.

## Intended Audiences

This system requirements specification has the following intended audiences:

* **Architecture Team& Tech Leads -** to drive and validate the system architectures.
* **Test Leads -** to understand the scope of the application to be delivered.
* **Configuration Mgt. Team -** to drive the design of dev, test environments, as well as version controlling, branching etc.
* **Test Engineers Team -** to generate test artifacts
* **Orchard Team** - to manage project scope and schedule project activities.
* **Dev. Engineers Team -** to drive the design of the software components for implementation (coding and related activities) purpose.
* **Any other stakeholders not explicitly mentioned above.**

## References

Customer Requirement Document:

* Happy Trip Project Requirement.pdf

## Specification Overview

This SRS is organized into the following sections:

* **Introduction:** This introduces the SRS for Happy Trip to its readers.
* **System Overview:** This provides a high level description of Happy Trip system including its definition, functions, context, and typical usage.
* **System Operational Requirements:** This specifies the system’s operational (functional) requirements
* **Architecture and Design:** This specifies required architecture and design to be treated as requirements.
* **Future Enhancements:** This section will specify the planned future enhancements.
* **Appendices:** This defines ancillary information including open issues and TBDs, etc.

# System Overview

Happy Trip is one of the leading companies in the world which provide services in the Travel domain. Happy Trip has decided to launch a portal for users to book flight tickets online for travel across India. This portal will also have administrative module to maintain flights & their schedules, inventory etc.

This portal would essentially have 2 modules:



1. User Module – All user related operations, like, features to book tickets online, maintaining user’s profile, and user related booking history management will be part of this module
2. Admin Module – To manage airlines, flights, routes, cities, schedules & flight inventory (view only) on the portal

## System Usage

* Search flights
* Book flight tickets, with Baggage Insurance Option
* Maintain registered user’s profile
* Maintain airline information
* Add Flight Information
* Maintain cities
* Add route information
* Add Schedules for flights
* Happy miles credit & debit management (Only for registered users) -HappyMiles Integration
* Cancellation of ticket (Only for registered users)
* Payment gateway Integration – For Booking & Cancellation
* View booking history
* Inventory for each scheduled flight (Read-Only seats availability)

## Context

The subsection uses context diagrams to document the context of the Happy Trip in terms of the external actors, data repositories, hardware, networks, software, and systems with which it interacts.

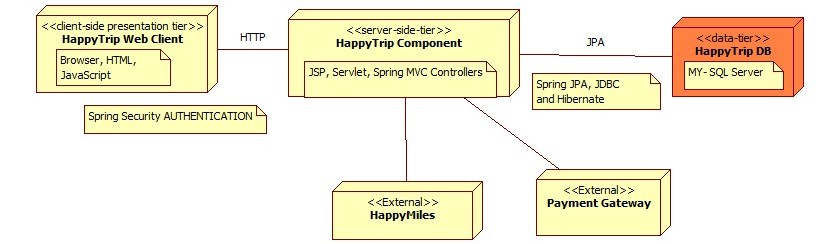


Figure 1: HappyTrip Context Diagram

### Human Actors

HappyTrip interacts, either directly or indirectly, with the following significant human actors

* Guest Users:
  + Who uses the system to search flight with desired details, book ticket (with or without insurance)
  + Who is interested in registering to the portal



* Registered Users:
  + who uses this system to search flight with desired details, book ticket (with or without insurance), to earn miles for registered airlines of HappyMiles, views booking history & can cancel a booking whose departure is atleast 3 hours ahead in future.
  + Who uses this system to manage his profile
* Admin:
  + Who uses the system to manage airlines, flights, routes, cities, schedules & flight inventory (view only) on the portal



### External Data Repositories

HappyMiles DB: This Database will store the business rules & policies implemented for airlines registered with the HappyMiles service. This will include, for each airline, number of miles earned, minimum transaction amount & base exemption, if any.

### External Hardware

None

### External Networks

None

### External Systems

1. Happy Miles
2. Credit Card Payment Gateway

# System Operational Requirements

The section of the SRS specifies the system’s operational requirements in terms of a High Level Design diagram.

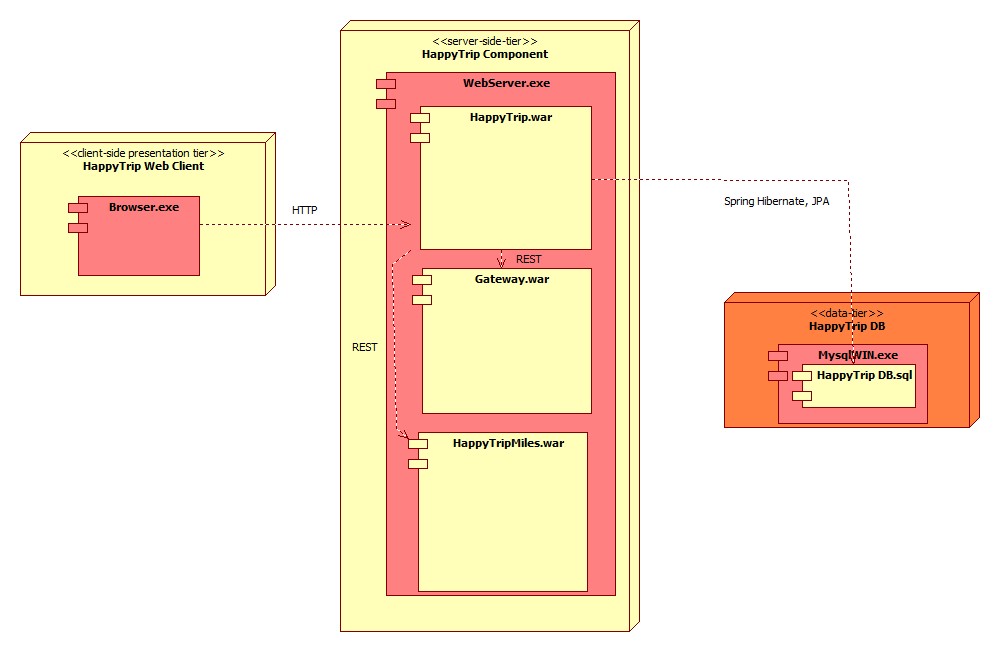
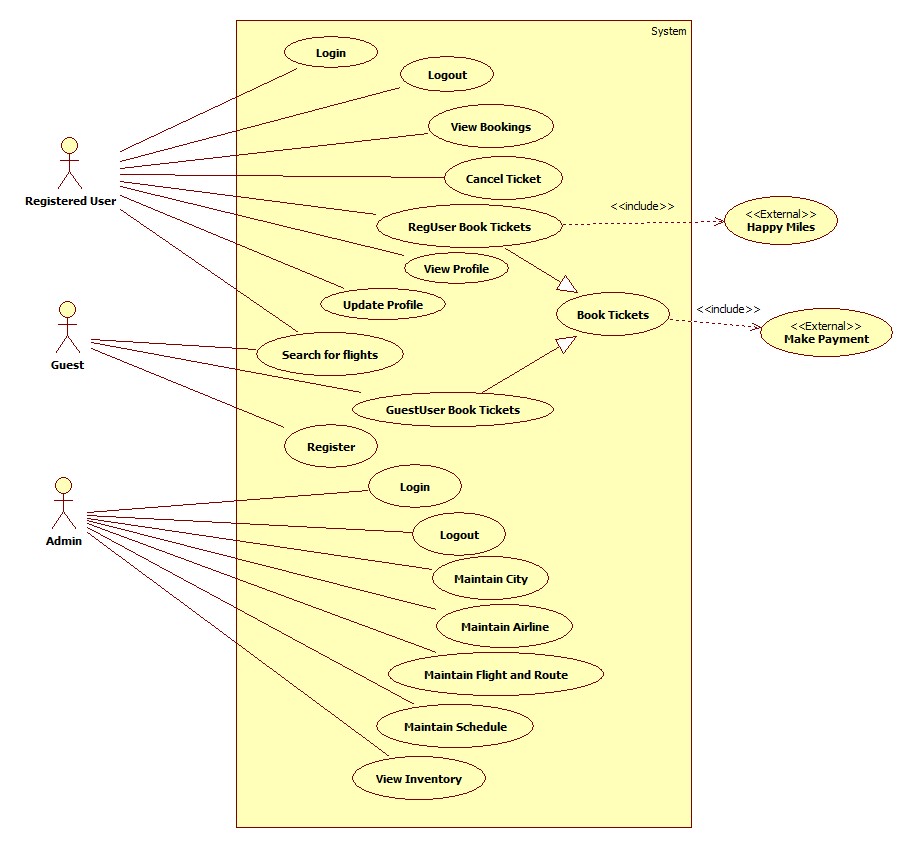


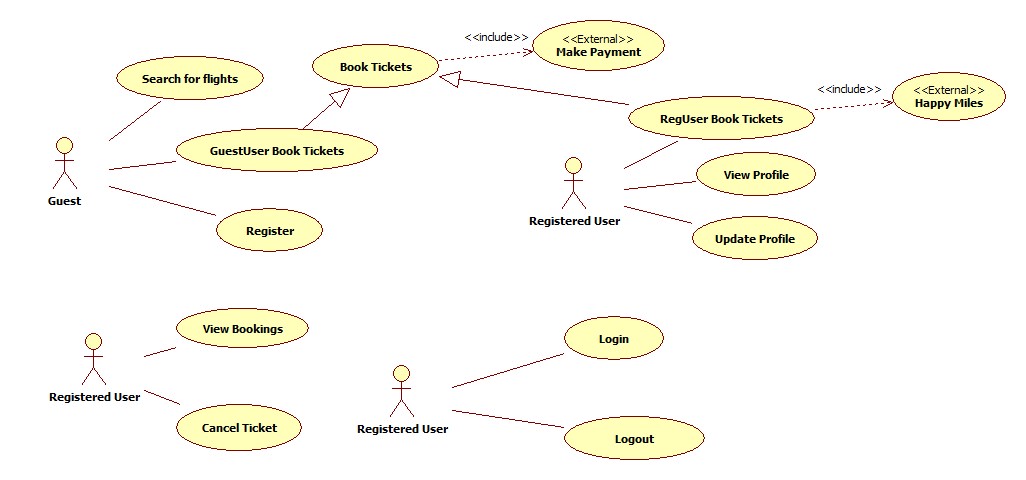
Figure 2: HappyTrip High Level Design Diagram

## Use Case Diagrams

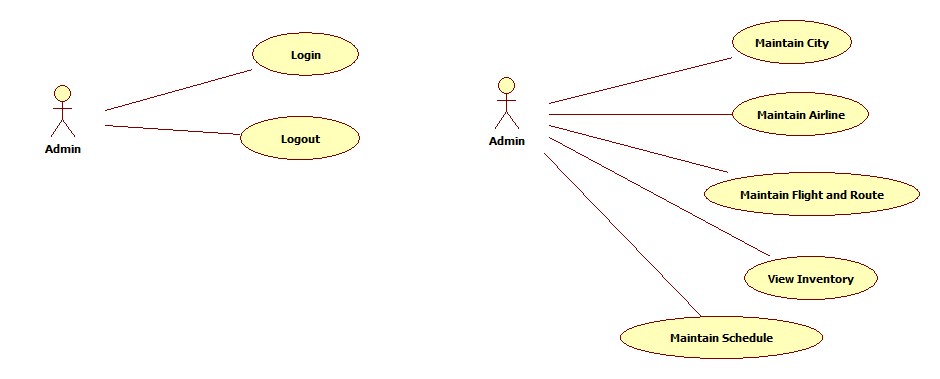
### HappyTrip System diagram



### User Module



### Admin Module



## Functional Requirements:

### User Module

#### Registration

****

User can register on the portal by entering information like

1. Full Name \*
2. Email Id \* [Also the login id]
3. Password \*
4. Gender
5. Date Of Birth \*

Fields marked \* are mandatory

* Login Id has to be a unique email ID
* Once user registers, he will be redirected to the sign-in page, to login using the previously entered email Id & password.
* The user will be logged onto the system & can select to view his profile, booking history (immediately after registration this section will not have data)

#### Login

****

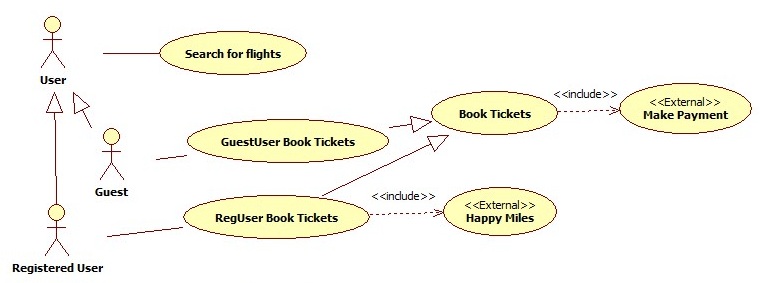
1. Registered User can login to the portal using his/her login id and password given during the time of registration
2. After logging in user is redirected to search page, where he/she can enter travel related information and search for flights available to book tickets
3. This user has access to “My Trips” section
4. Also can view and edit profile information, which also holds total Happy Miles earned by the user

#### Profile

****

* 1. Once user registers with happytrip.com, a profile is automatically created, with basic data entered during registration
  2. User can view & update his/her profile information, upon logging into the portal
  3. Profile information would include
     1. Email Id [Which is also the login id, cannot be changed]
     2. Full Name
     3. Gender
     4. Date Of Birth
     5. Address
     6. Pin Code
     7. Phone number – Can be selected either Mobile/Landline
     8. Miles Earned
  4. Full Name, email-id, gender, date of birth and password is collected during registration and the rest can be added anytime by the user
  5. The relevant information from profile is displayed during booking, prefilled in the passenger details page, only for the registered user

#### Book Ticket – [Both registered and non-registered users]



* 1. User can book tickets on the portal with or without registration
  2. First, user has to enter travel information and to search the flights available for his/her search criteria
  3. Information to be given on search page
     1. Radio button for Direction Of Travel – One Way or Return – Default selection: One way
     2. From City
     3. To City
     4. Date Of Travel
     5. No Of Seats – In terms of No. of Adults, Children & Infants – default selection: 1 Adult only
     6. Class of Travel- Default Selection is “Economy”
  4. When user selects “Return” option, the system should display “Return Date” field as well
  5. Checks should be done to ensure that the user don’t attempt to book tickets beyond 3 months from the current date.
  6. Checks should also be done to ensure that there are seats available in the scheduled flights that match the search criteria, else user should be shown message for no results found
  7. Checks should also be done to ensure that there are flights operating between from and to destinations
  8. For a roundtrip, if there are no results found for one of the either of the legs (going & coming back) of the trip, then the user should be shown message for that part of journey & the other part of journey should still be available for proceeding to book
  9. Checks should be done to ensure that only those flights that depart atleast 3 hours from the time of search are listed
  10. If there are scheduled flights found for the above mentioned information, then all those flights along with each one’s details should be listed
  11. Information to be displayed for each flight would be
      1. From City
      2. To City
      3. Date of Travel
      4. Flight Name
      5. Airline Name
      6. Departure Time of Flight
      7. Arrival Time of Flight
      8. Cost of the ticket /per person
  12. Results should be sorted in the order of price, with Lowest price on top From here user would select an appropriate flight for each of direction of his/her journey
  13. Upon selection, user can click “Book” button & he should be navigated to next page where he would be able to enter the below mentioned information to book the ticket
      1. Title (Mr, Ms)
      2. Full Name
      3. Gender (Male/Female)
      4. Date Of Birth
      5. Mobile No
  14. And an option to select Baggage insurance, which is calculated as follows
      + 1. Baggage insurance is calculated for each traveler
        2. This is 5% of the booking amount for each ticket
        3. The minimum fee is INR 250/- and maximum is 500/-
        4. This amount should be added to the total ticket price
  15. If registered user has logged in, then the application should prefill all these details from his/her profile.
  16. Registered user can edit/update the prefilled information on the page
  17. User must enter the details of additional passengers who will be travelling along. For each additional passenger, the following details must be provided.
      1. Title
      2. Name
      3. Gender
      4. Date of birth
  18. All fields mentioned above would be mandatory & Baggage insurance is one option, which is applicable for all travelers
  19. After entering all the information, a confirmation screen should be provided where user is shown all the details regarding the ticket to be booked
      1. Full Name – First passenger’s name for guest user transaction & Registered user’s name for registered user
      2. Date Of Birth
      3. Date Of booking
      4. Departure Time of Flight
      5. Flight Name
      6. Class of Seat
      7. Route
      8. No Of Passengers
      9. Total Insurance Amount
      10. Price of tickets
      11. Total Price of the booking, including insurance
  20. If the user confirms these details, he should be proceeded towards the page to Accept payment from the user
  21. Payment is made via a payment gateway, which would be hosted on a separate environment & HappyTrip is integrated with it to validate the user details
  22. In this page, user has to enter the payment related information like
      1. Credit Card No – 16 digit number
      2. CVV No – 3 digit number
      3. Credit Card Type (Visa/MasterCard)
      4. Card Holder Name
      5. Expiry Date – Month and Year
      6. Total Amount to be debited should be displayed
  23. If payment process fails, user has to be informed about the same and should be asked to retry the booking
  24. If payment is successful, a unique booking no is generated and displayed to the user along with detailed travel information as below:
      1. Full Name – Same as mentioned above, while booking confirmation
      2. Date of Birth
      3. Date of booking
      4. Booking Reference No
      5. Date of Travel
      6. Departure Time of Flight
      7. Flight Name
      8. Class of Seat
      9. Route
      10. No Of Passengers
      11. Total Insurance Amount
      12. Happy Miles Earned
      13. Price of tickets
      14. A list of passengers, with the following information displayed row-wise for each traveler in this booking
          1. Title
          2. Name
          3. Gender
          4. Date of Birth
      15. Total Price of the booking, including insurance
  25. This Data of successful transaction should be stored for the booking & it is reflected in the My Trips section as well
      1. For a return journey, two reference numbers will be generated & my trips will show two separate entries for the same, wherein each will have their status managed separately.
  26. Only a registered user is eligible for Happy Miles based on the business rules defined by Happy Miles Program
  27. Happy Miles are not calculated for guest user, so their E-Ticket will display it as Zero
  28. For any airline that is not listed in HappyMiles program, HappyMiles earned for the transaction will be Zero
  29. The Happy Miles Application will compute the number of Happy Miles and this varies for each airline
      1. The Happy Miles calculation rules for each airline depends on the following factors:
         1. Minimum Booking value
         2. Miles per rupee spent
         3. Base exempted amount
      2. The HappyMiles calculation is done as per the formula
         1. If Total Booking amount > Minimum Booking amount specified for the airline, then
            1. (Booking Value – Base Exempted amount) \* Miles per rupee spent
         2. Else, HappyMiles earned are Zero
      3. For example, if the rules for calculating miles for Jet Airways is:
         1. Minimum booking value is Rs10000/-
         2. 2 (Two) Miles for every rupee
         3. Base exempted amount is Rs.5000/-
         4. Then the miles earned for a transaction of Rs 10000 will be 10000 HappyMiles
      4. For example, if the rules for calculating miles are for Kingfisher is:
         1. Minimum booking value is Rs.5000/-
         2. One Mile per for every rupee
         3. Base exempted amount is Rs.1000/-
         4. Then the miles earned for a transaction of Rs 10000 will be 9000
      5. The participating airlines & their business rules will be provided during deployment

#### Cancellation

**Reg User**

**Cancel flights**

* 1. Only, a registered user can cancel an upcoming journey from his booking history (“My Trips” section)
  2. A flight can only be canceled if at the cancellation time, the time remaining for its departure is atleast 3 hours, else this option will not be available for the cancellation
  3. A cancelation fee of INR 750/- is charged for each passenger per ticket (irrespective of the class), in the booking transaction being cancelled. So for a return journey, where onward & return are treated separately in My Trips, the cancellation charges will also be applicable to them separately and not as one transaction.
  4. If the booking amount is less than or equal to Rs 750, nothing should be refunded back
  5. Baggage Insurance charged in the booking, is non-refundable, hence it will be forfeited for each passenger
  6. Once user confirms cancellation of the booking, he should get a cancellation process success message on the next page, with the amount refunded for the transaction.
  7. Happy Miles associated with the cancelled transaction should be reversed in the user’s account
  8. Refund payment is credited back into the Credit card (used while booking) account of the user and can only be tracked by making bookings subsequently with the refunded amount added to account balance.

#### My Trips (Booking History)

**RegUser**

**View Booking History**

1. The Booking History can be viewed only by a registered user in My Trips section
2. This page will show details of the journeys completed and the upcoming journeys
3. The Booked Trips listing for the user displays details for each booking transaction of the logged on user, as follows:
   * 1. Booking ID
     2. Flight Information – Airline Name, From City & To City
     3. Date of Departure
     4. Total amount paid for this booking
     5. Seat Class
     6. Status of the booking – Flown, Cancelled, Cancel (Link to cancel booking)
4. “Cancel” Option should be shown only for an upcoming journey, which is scheduled to depart atleast 3 hours later than the current time

### Admin Module

Admin will be able to manage airlines, flights, schedules & inventory data through the admin module. Admin can login using his/her login id and password. These operations have to be carried out only by the admin and travel user should not have the access to any of the admin modules

#### Maintain Airline information

****

* 1. Admin should be able to maintain airlines with the following modules
     + Add a new airline
       1. Admin can add an airline by specifying the following fields
          1. Name
          2. Code
       2. Both these fields are unique for each airline and any new airline to be added, **should not** have a name or code, matching to an existing airline record
       3. Once added, this airline should be available in the Add Flight module’s dropdown for Airlines
     + View & Edit an existing airline
       1. Admin should be able to View all the airlines existing in the system, and the details displayed should be same as above (Name & Code)
       2. Admin should be able to Edit an existing airline, by selecting the airline & submitting it for an update
  2. The list of existing airlines should be sorted alphabetically

#### Add Flight information

****

* 1. Admin should be able to add a new Flight with following details
     + Airline\* - List of existing airlines in the system, alphabetically sorted in ascending order - First Airline name should be selected by default
     + Name of the flight\* – Duplicate flight names, should not be allowed (uniqueness is maintained across airlines and not as a composite airline-flight pair)
     + Class Wise Capacity – Only for Economy and Business classes
       1. Capacity for Economy can be maximum 200
       2. Capacity for Business can be maximum 50
       3. Validations should be done to mandate capacity field to be entered for atleast one class & the capacity can only be a positive number, greater than 0
  2. Once added successfully, the flight should be available for selection in the Schedule module

#### Maintain City Information

****

* 1. Admin should be able to add a new city as follows
     + State - List of existing States in the system, alphabetically sorted in ascending order - First State name should be selected by default
  2. The admin needs to only select the state & mention the city name (no duplicate city name should be allowed for a state)
  3. Admin should be able to view the list of existing cities along with each one’s corresponding state name
  4. Admin should be able to edit a city, by selecting it & confirming to update
  5. Once updated, the new city name should be reflected in Route & schedule modules
  6. The new city name should also be reflected in all the booking transactions executed in the system

#### Add Route Information

****

* 1. Prior to scheduling a flight, a Route needs to exist between Origin & Destination Cities.
  2. Admin should be able to add a new route, by selecting Origin city & Destination city (Both cannot be same)
  3. Once added, this new route should be available in the Schedule Flight module for selection
  4. Duplicate routes should not be allowed

#### Add Schedule Information

****

* 1. Admin should be able to schedule a flight on a route existing in the system by specifying the following details
     + Flight\* – List of existing flights in the system, alphabetically sorted in ascending order - First flight name should be selected by default
     + Route\* – List of existing routes in the system, alphabetically sorted in ascending order - First flight name should be selected by default
     + Distance covered in the route\*
     + Departure Date\*
     + Departure Time of Flight – 24 hr format
     + Arrival Date\*
     + Arrival Time of Flight – 24 hr format
     + Cost of Business Class (Per Ticket)
     + Cost of Economy Class (Per Ticket)
  + A specific flight on a route, with specified departure & arrival details, cannot be scheduled again on the same route with exactly same departure & arrival details.
  + \*\*Please note: Cost for a class of travel is same for all types of passengers – Adults, Children, Infants

#### View Day’s Inventory

**View Inventory**

**Admin**

* 1. For all the scheduled flights in a day, the admin should be able to view the inventory on a daily basis, for departures scheduled between 0000hrs to 2300 hrs of the day
  2. This will display the following details for each scheduled flight
     1. Airline Name
     2. Flight Name
     3. Route
     4. Departure Time
     5. Arrival Time
     6. Capacity in Economy class
     7. Capacity in Business class
     8. Seats Available in Economy class
     9. Seats in Business class
  3. This is purely a read-only list, which should be updated each time a booking is made in a scheduled flight of the current day or a booking is cancelled, for a scheduled flight for the current day

#### Error Handling

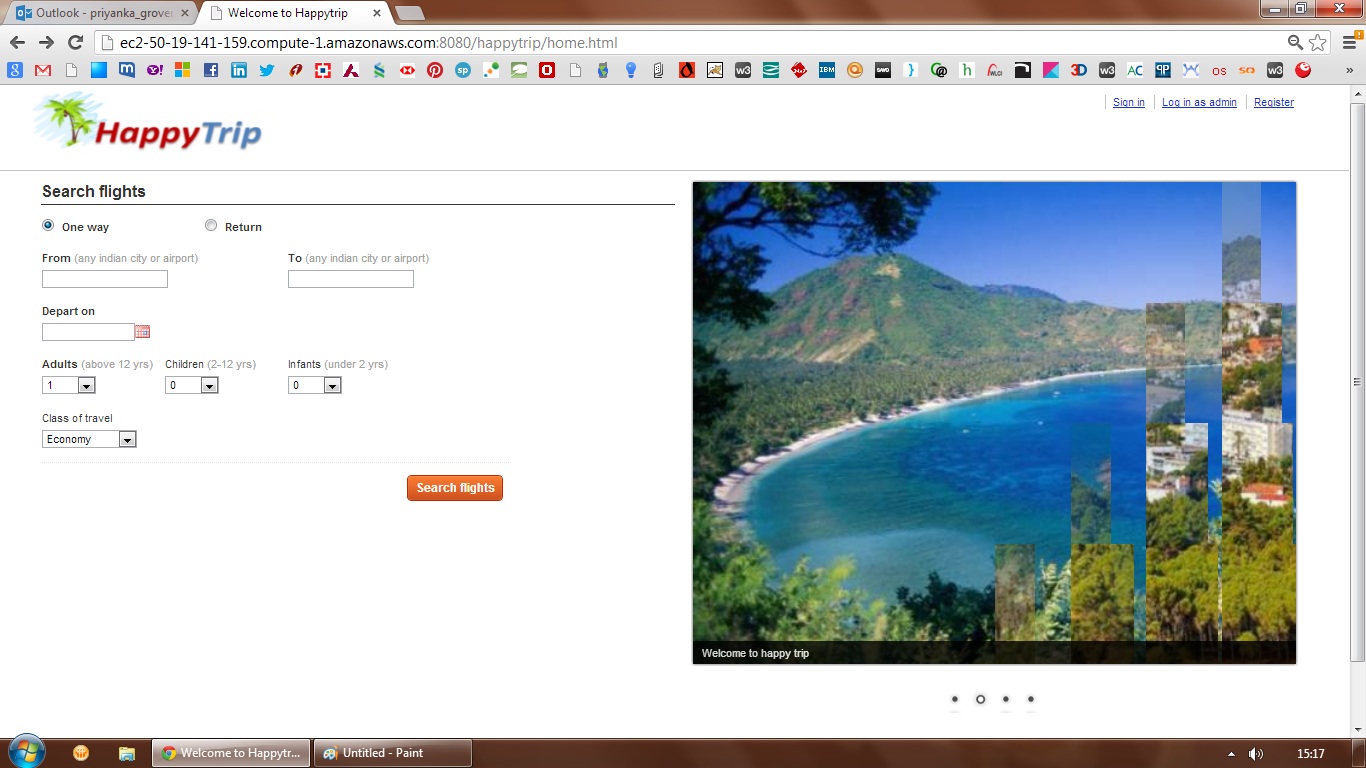
#### 

* All Validation Errors and exceptions occurring in the application should be handled & a message should be displayed by the system for the same.
* The system should have graceful exits defined for errors & exceptions

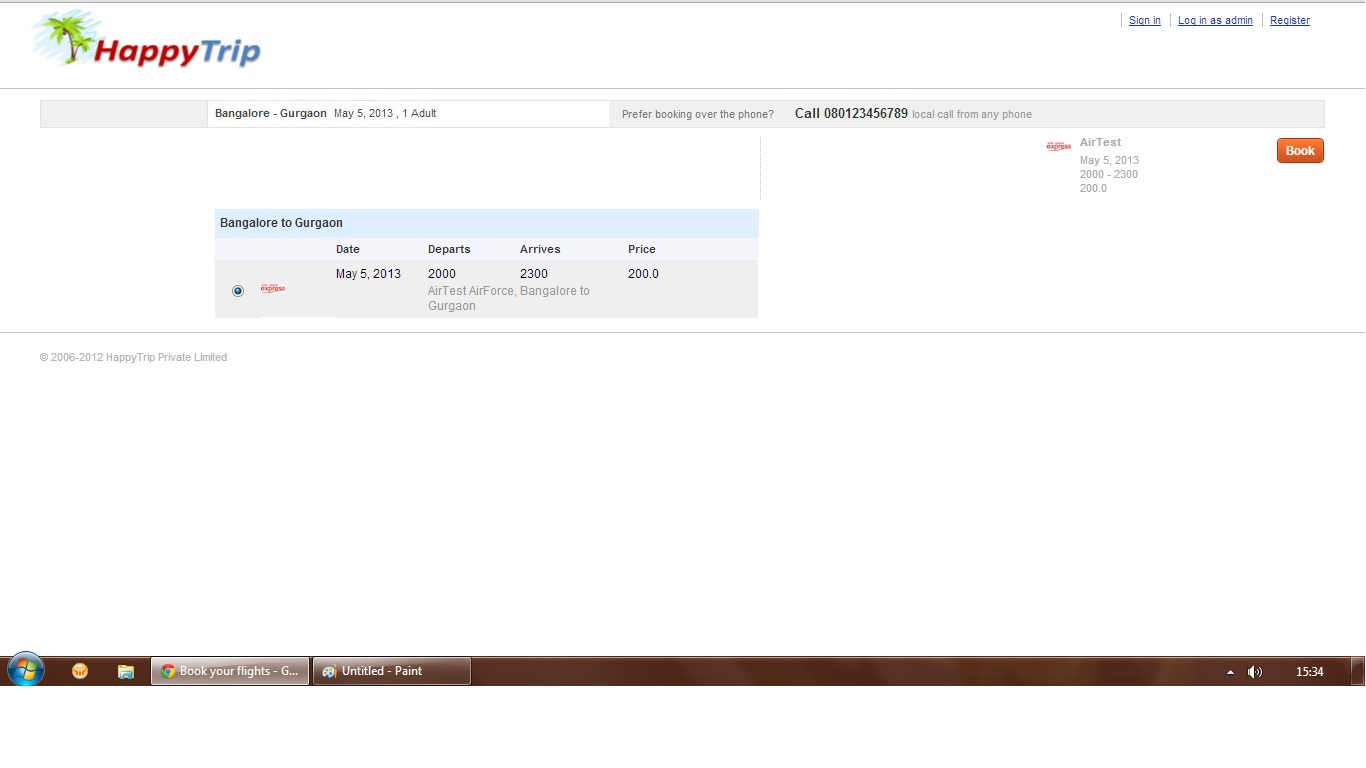
### GUI Requirements

The GUI requirements for this application are specified at each page level, by using the screen design prototypes as below:

#### Home Page



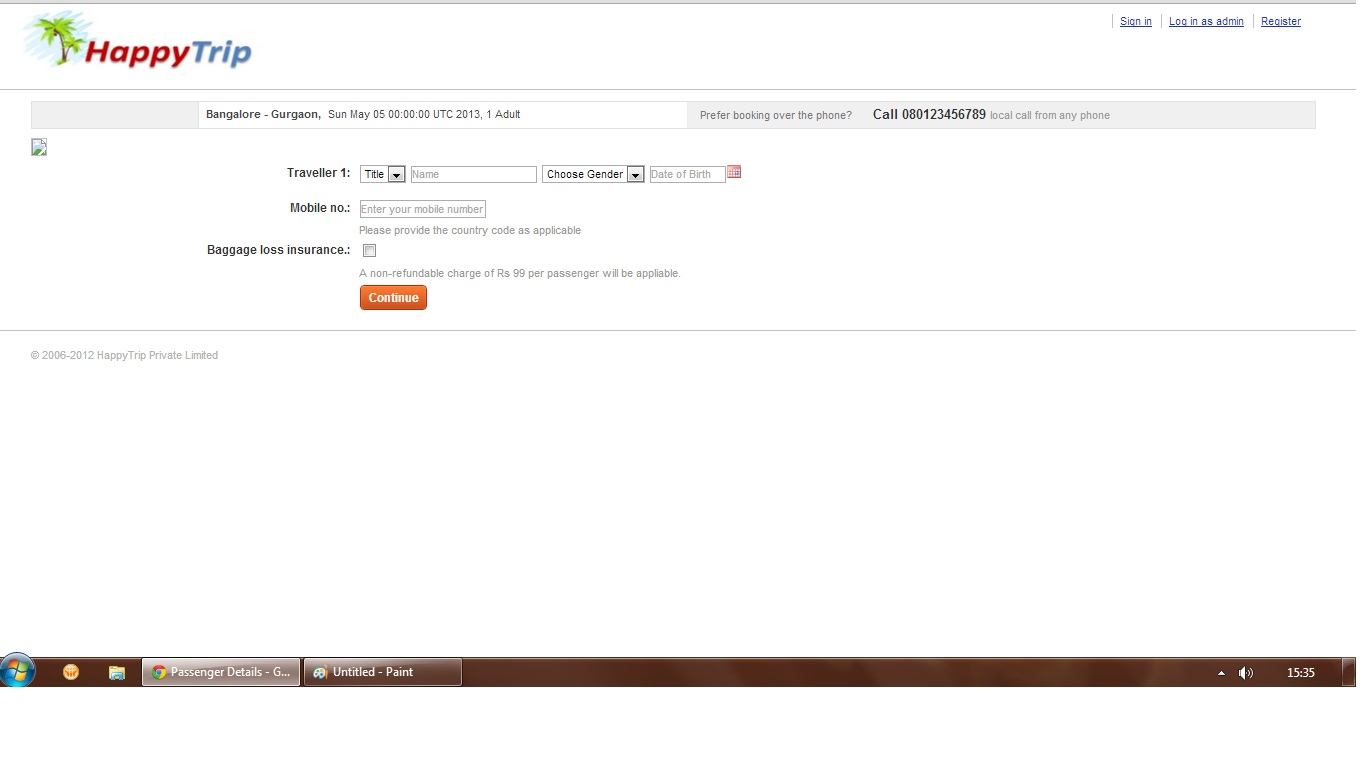
#### Search Results



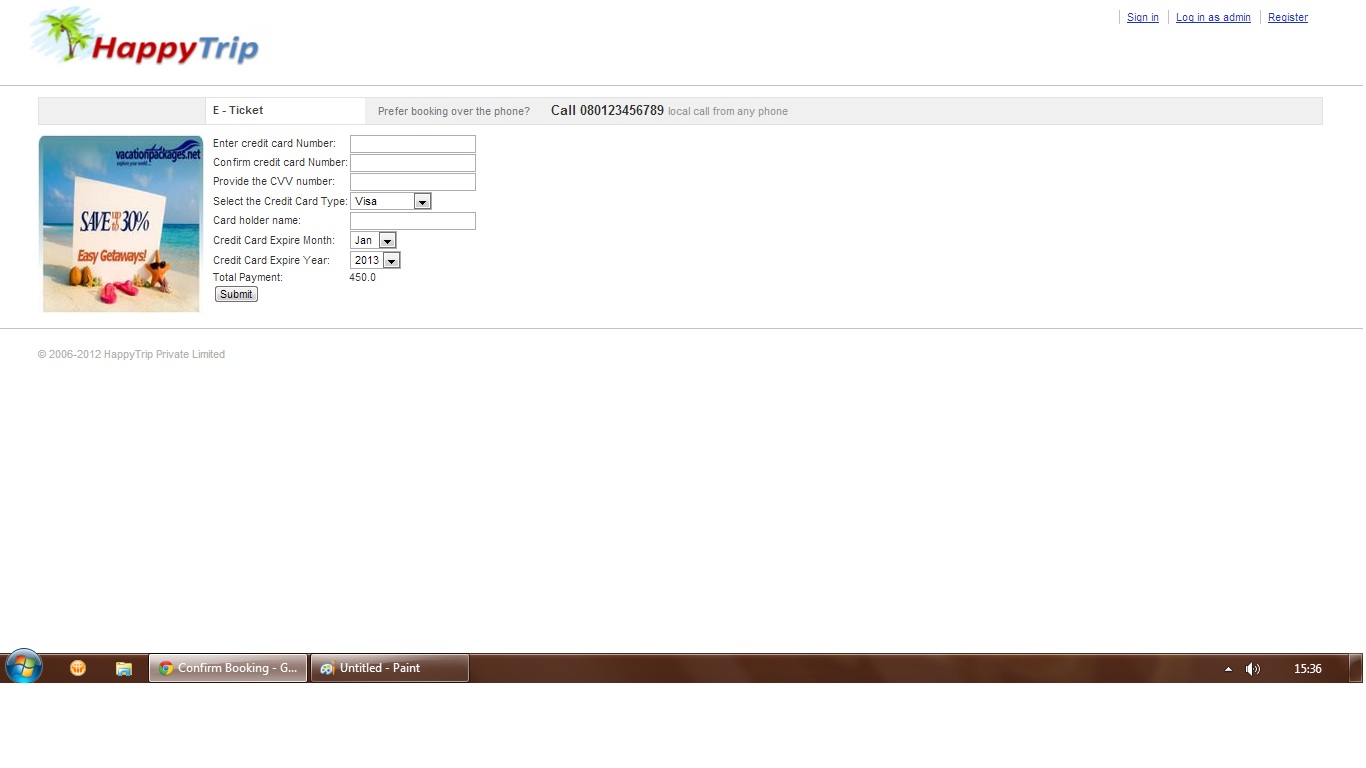
#### Confirm Booking

#### 

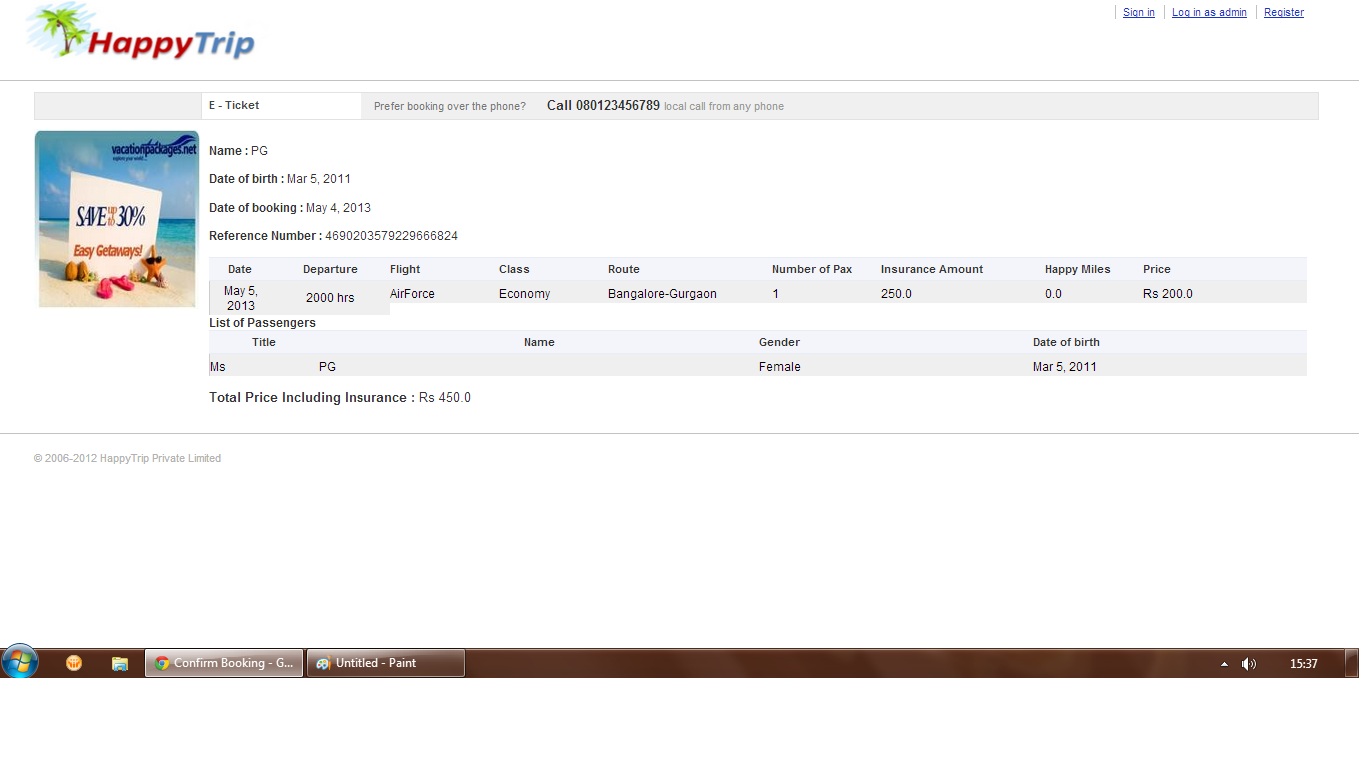
#### Passenger Details



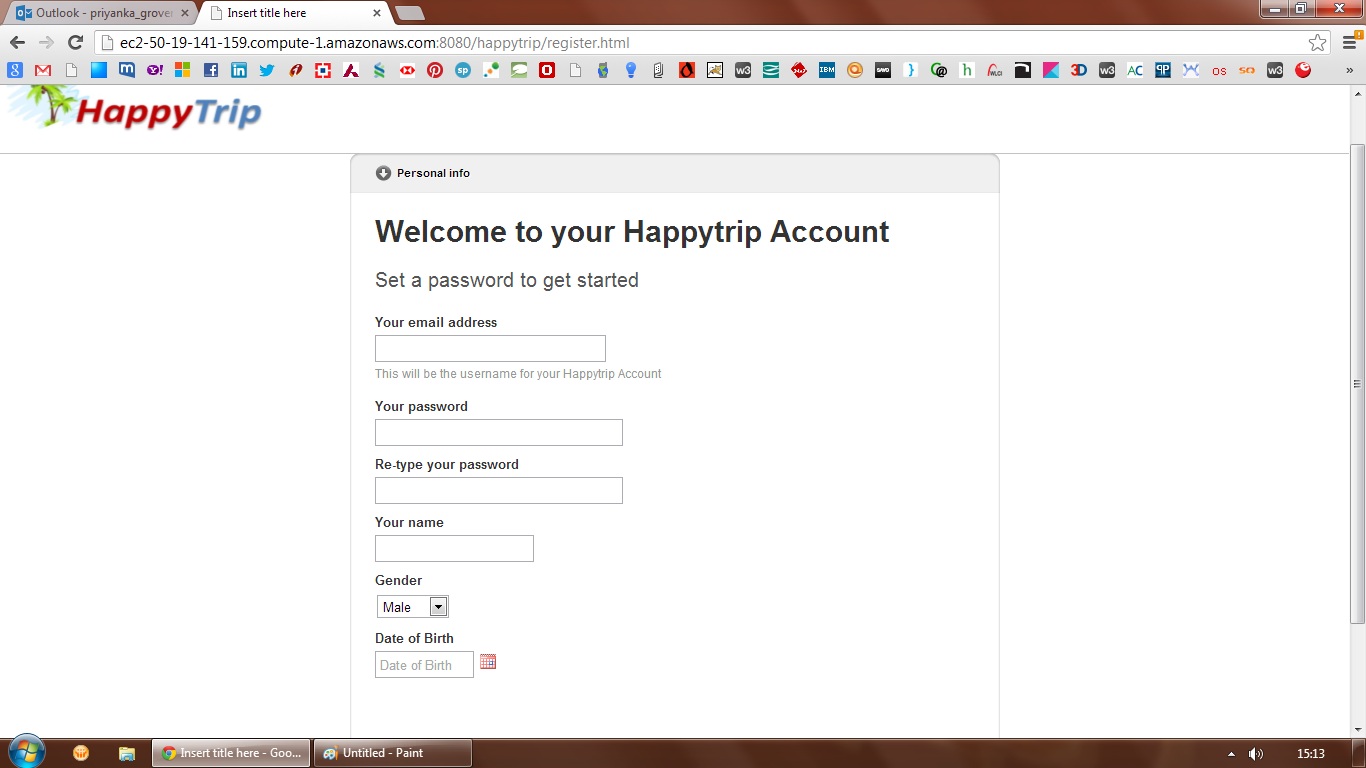
#### Payment



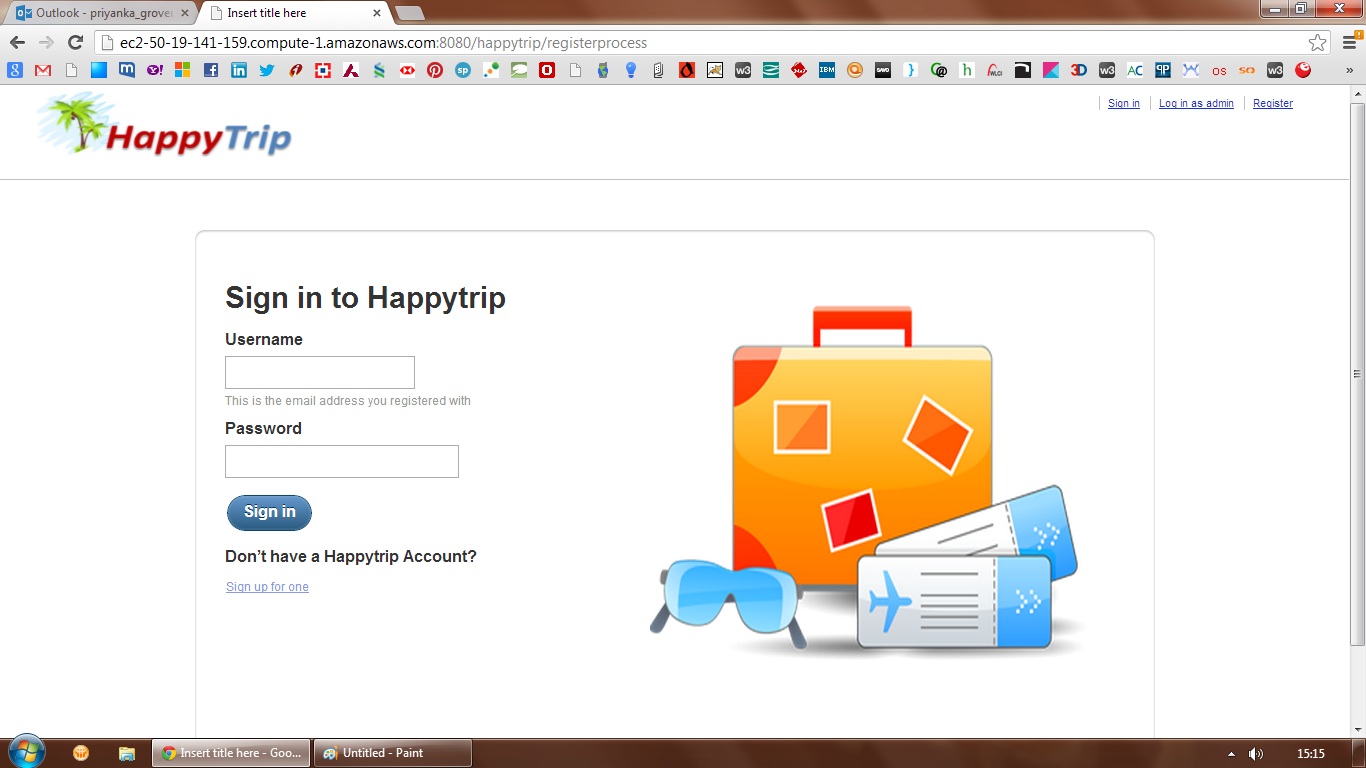
#### Payment confirmation - E-Ticket



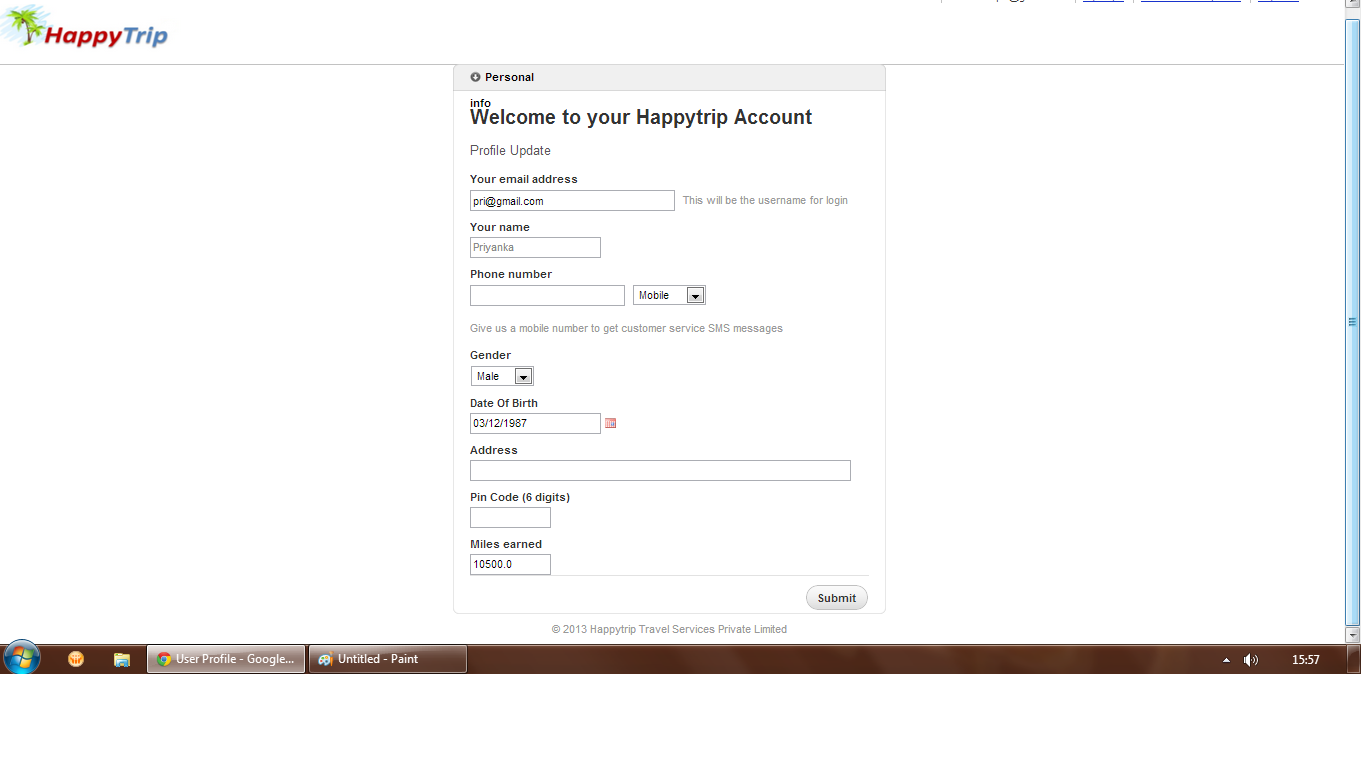
#### Registration



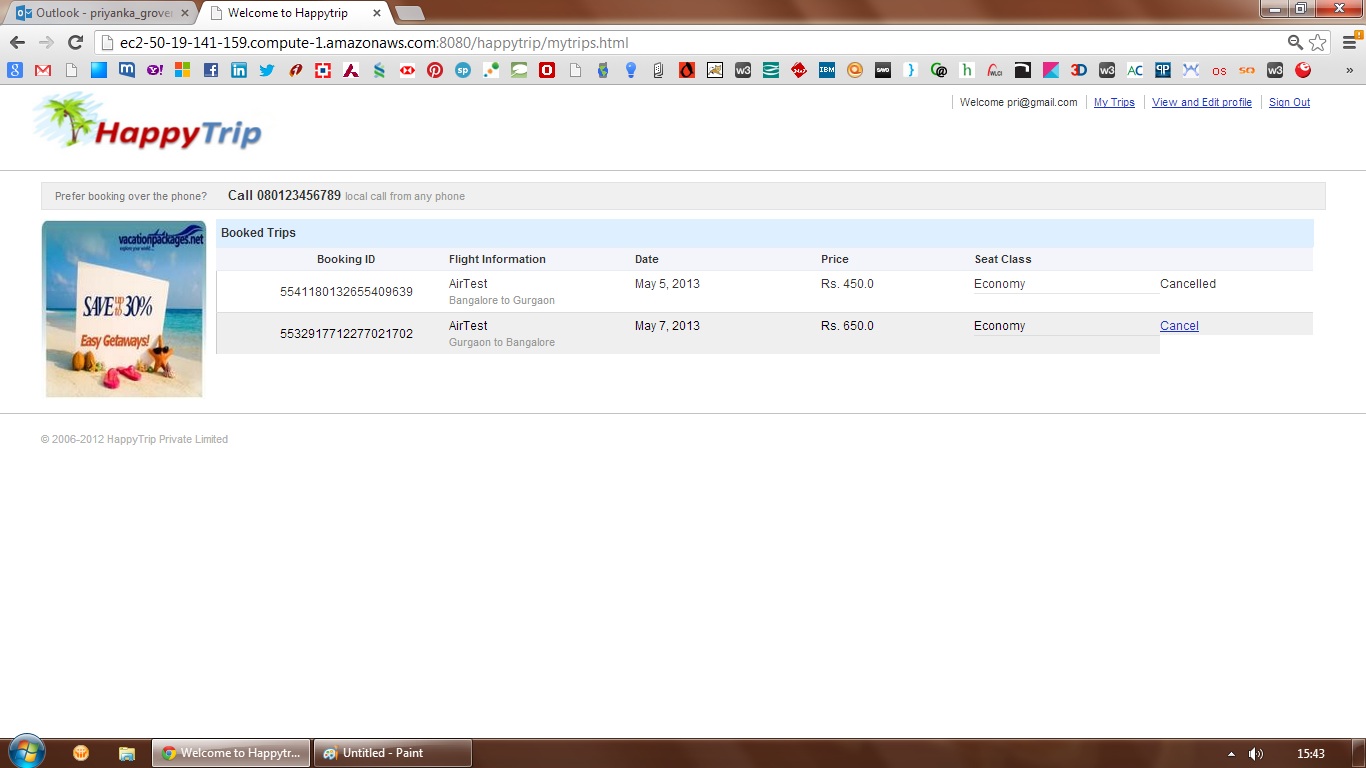
#### Sign in



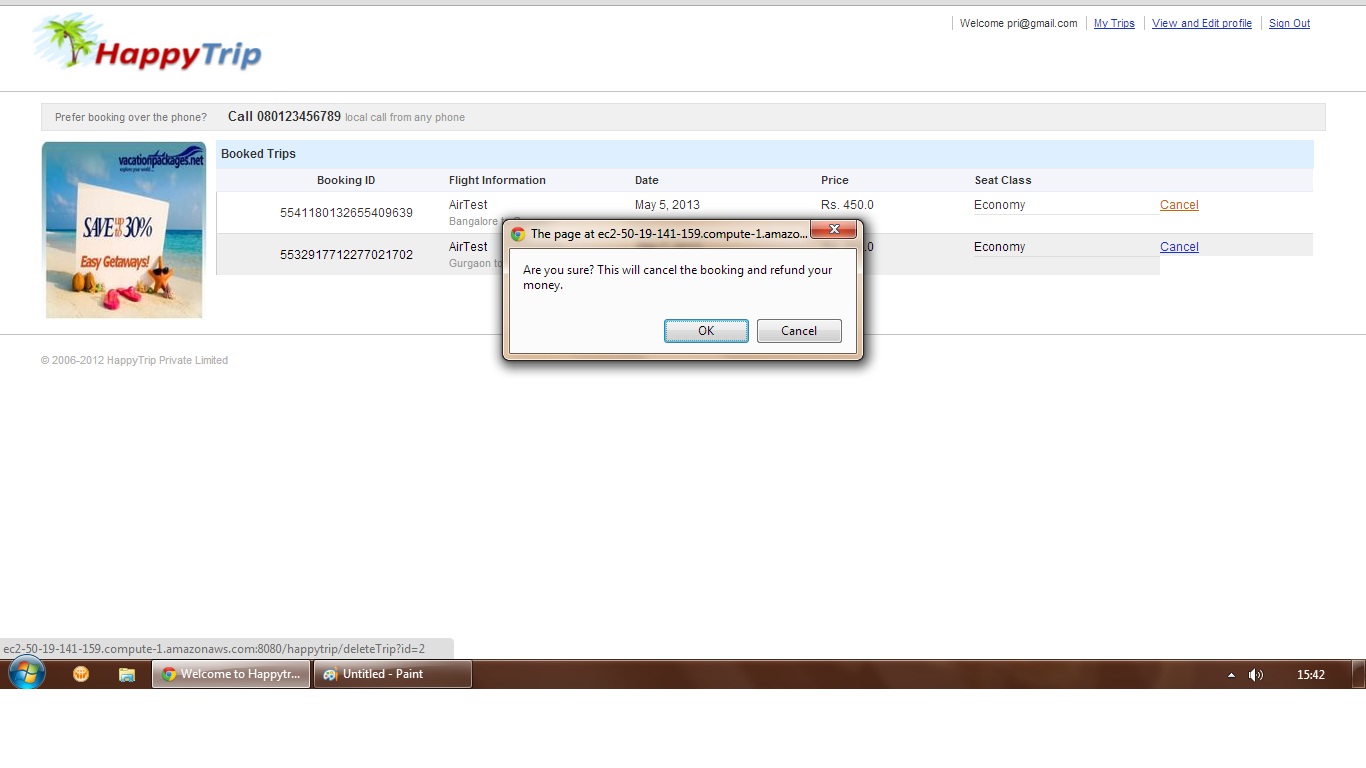
#### View/Edit Profile



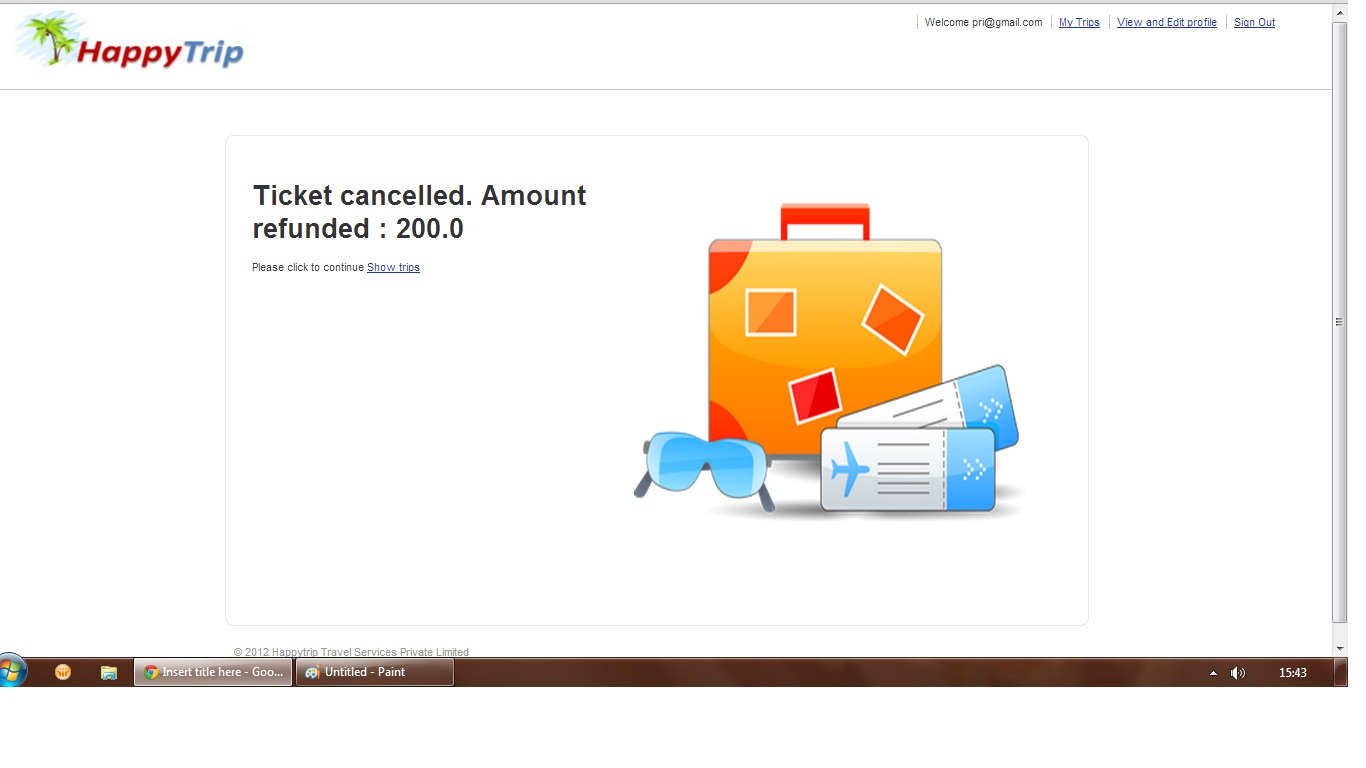
#### My Trips



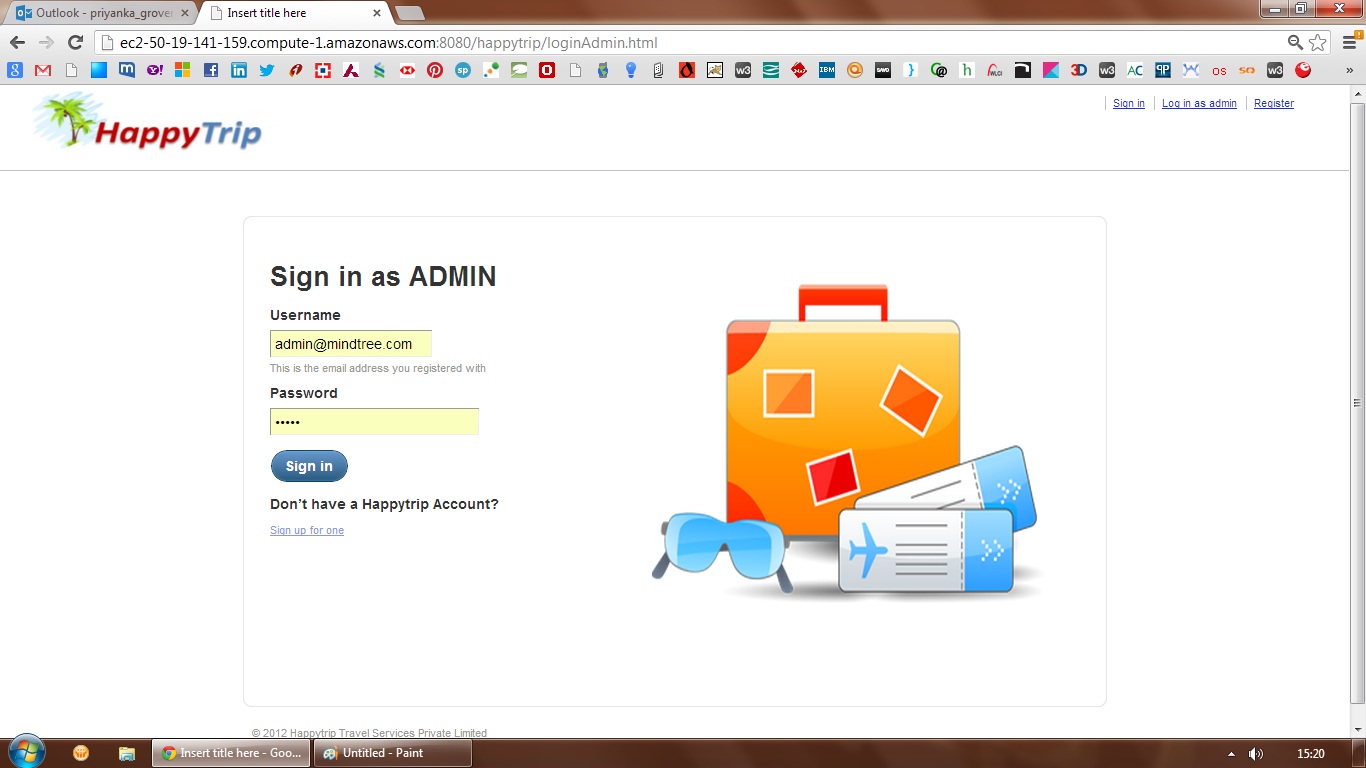
#### Cancel – Confirm Box



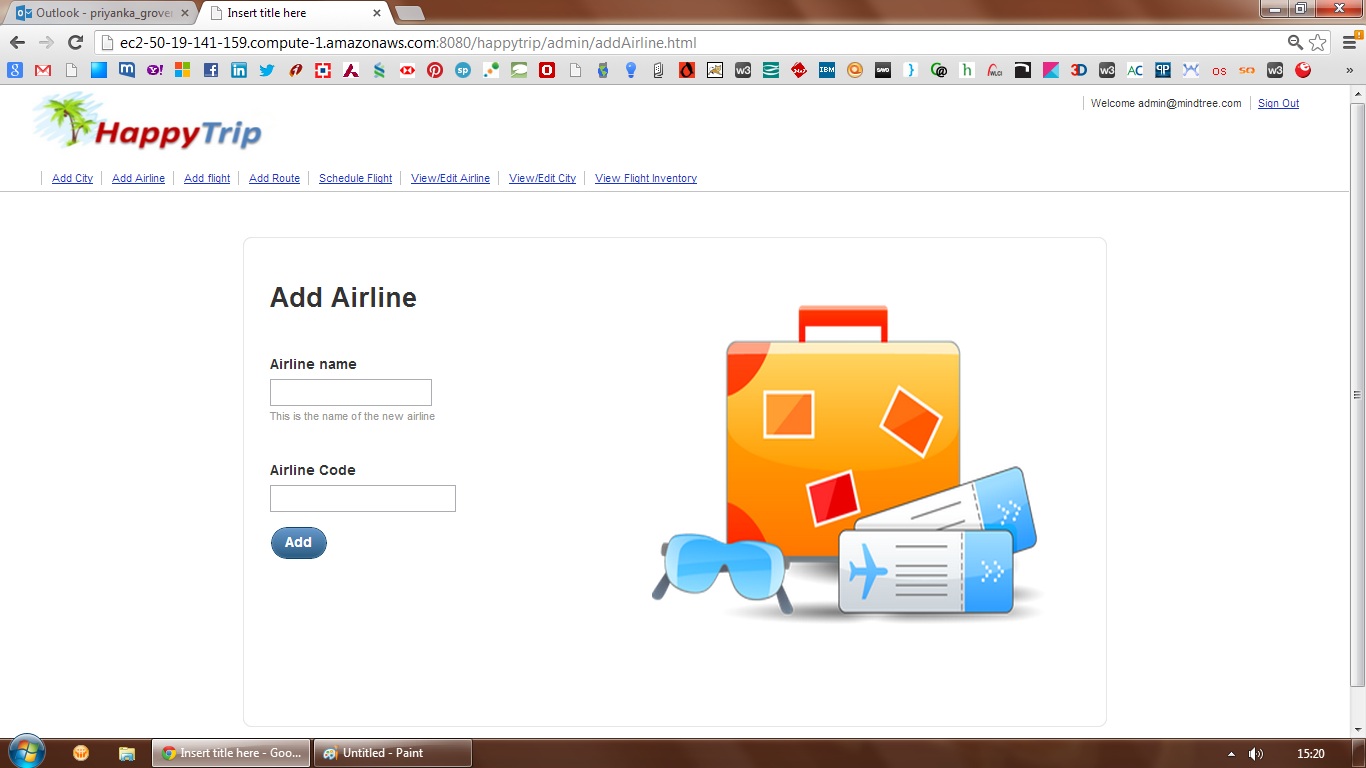
#### Cancel Success Page



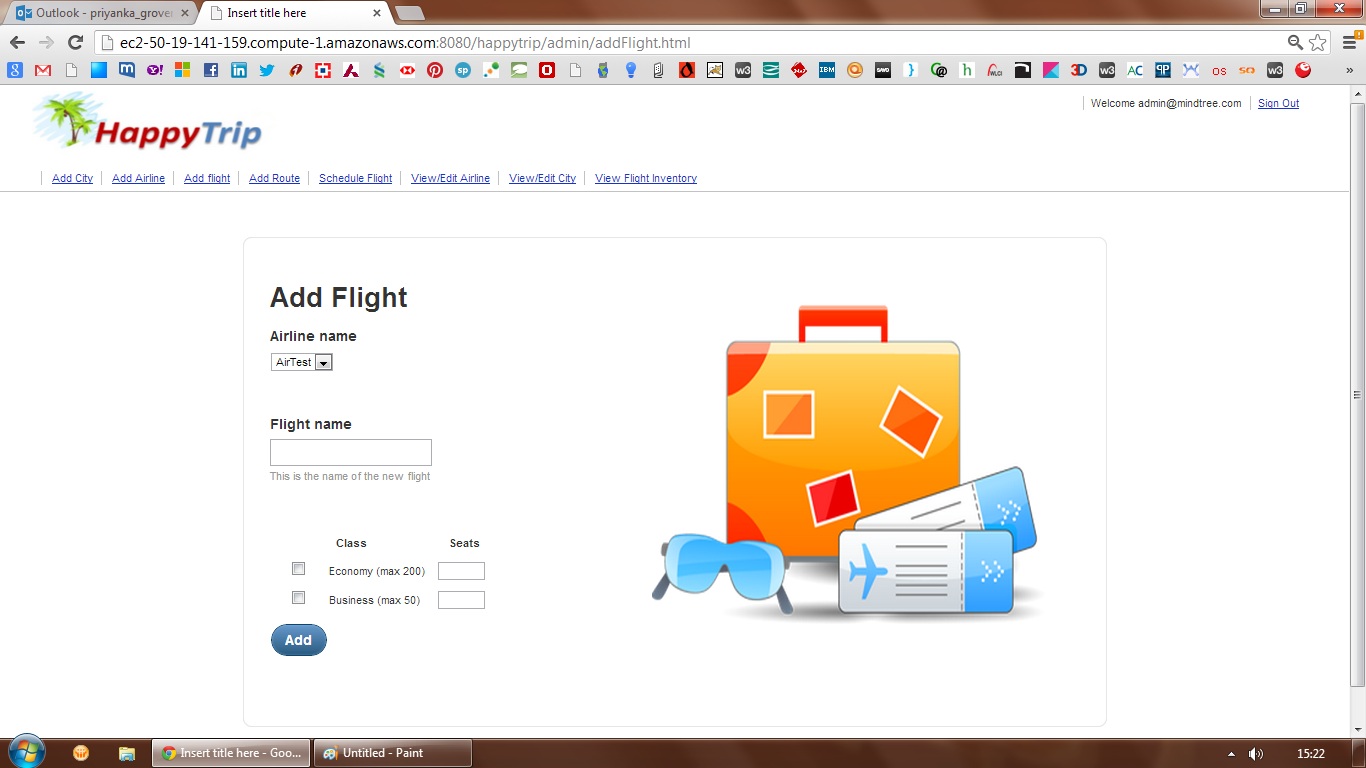
#### Admin – Login



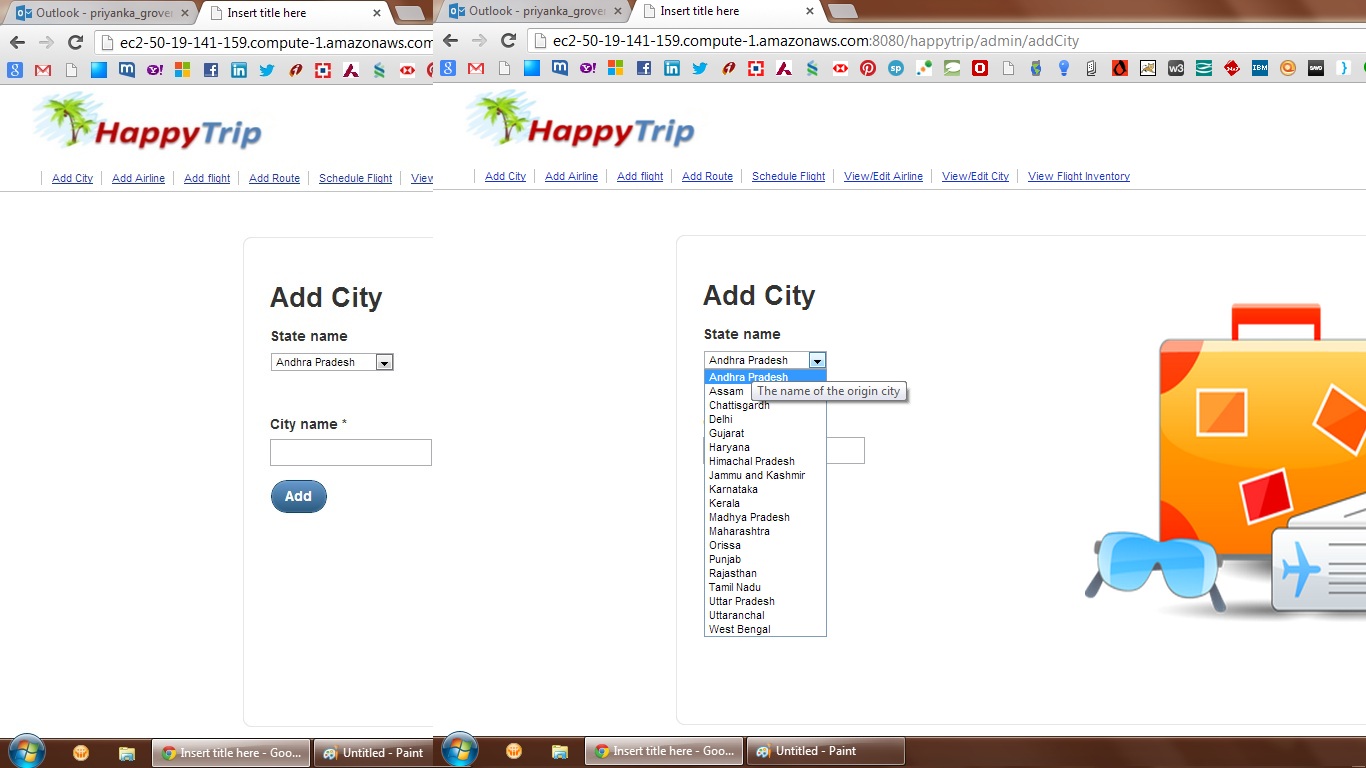
#### Admin – Add Airline



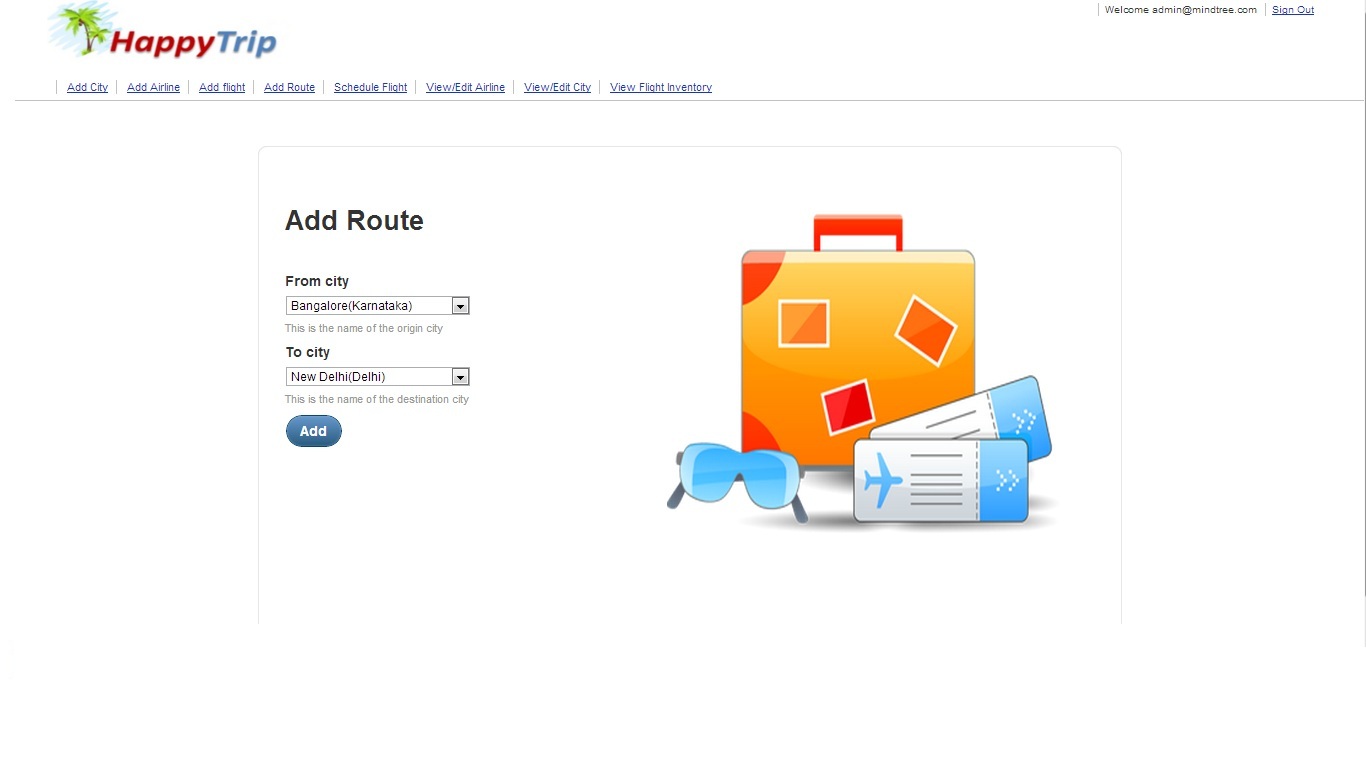
#### Admin – Add Flight



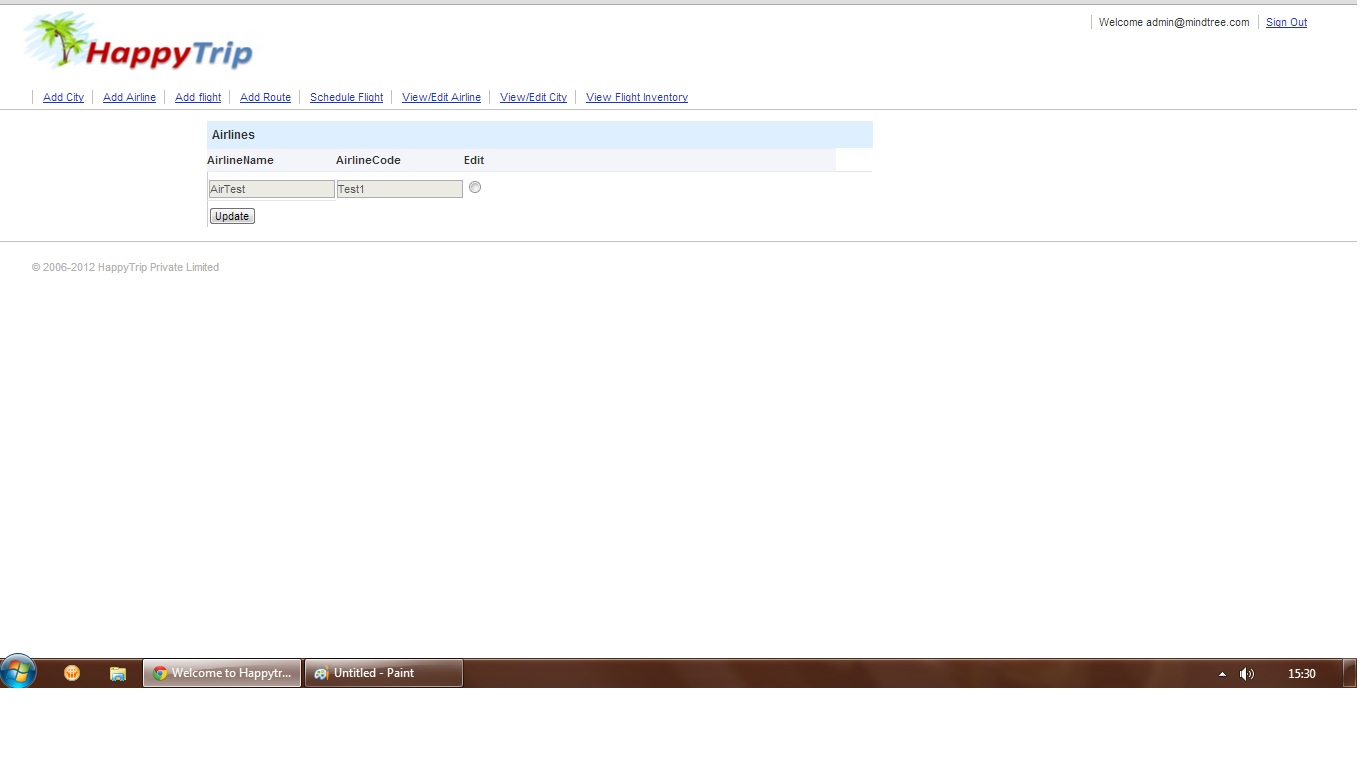
#### Admin – Add City



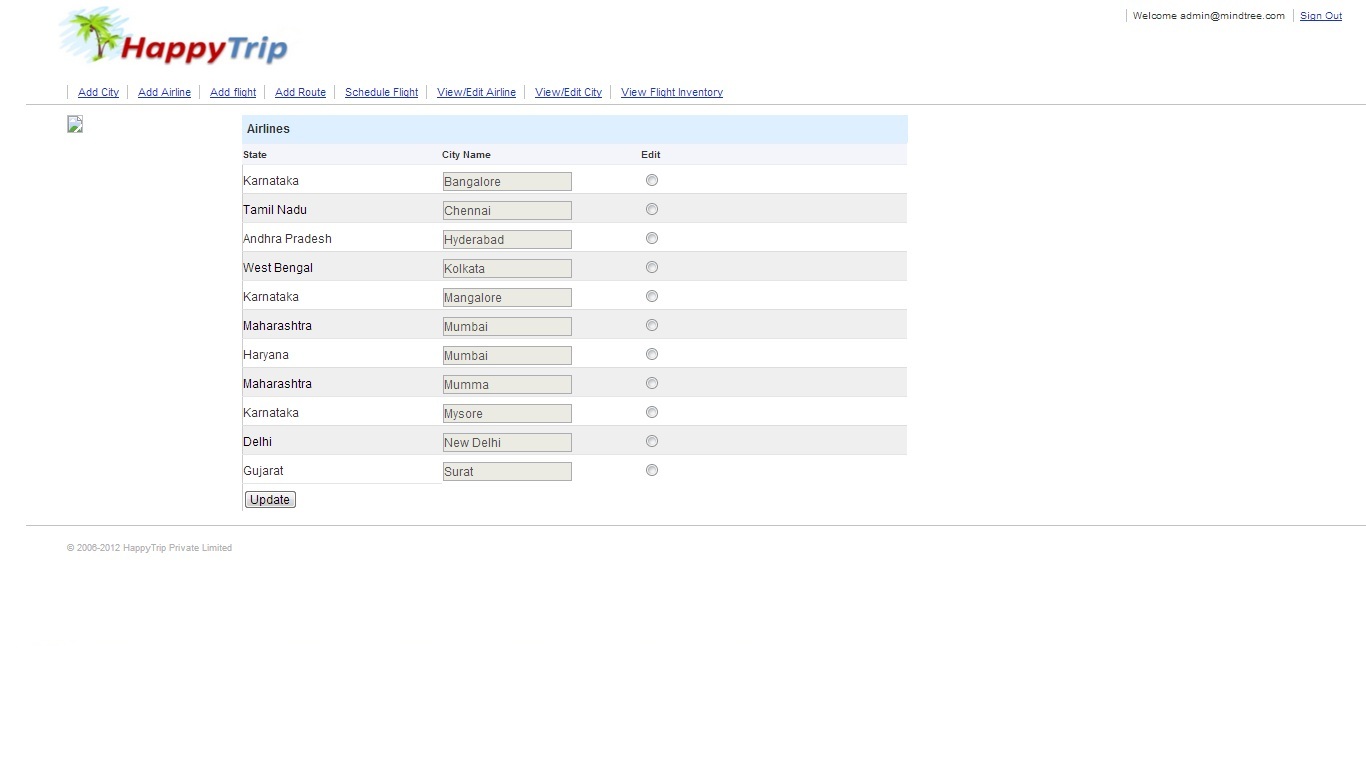
#### Admin – Add Route



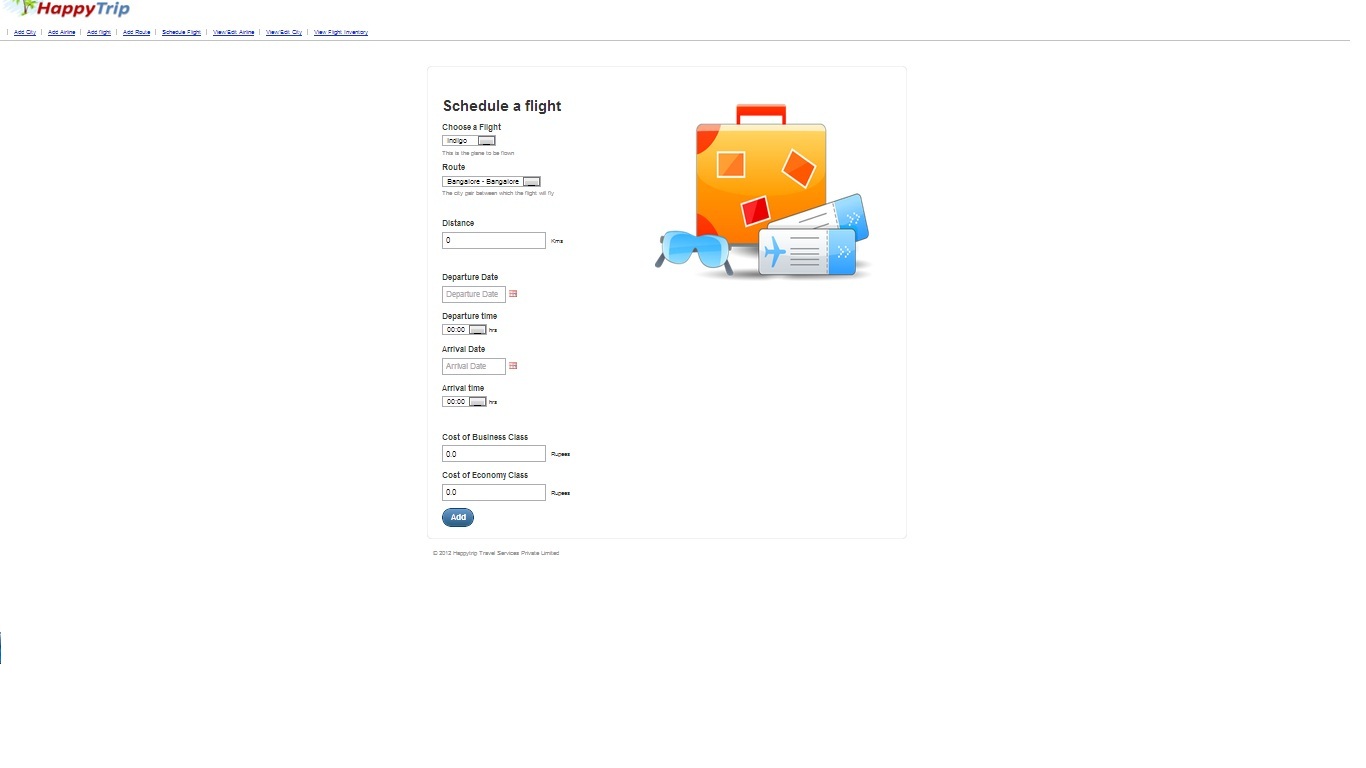
#### Admin – View/Edit Airline



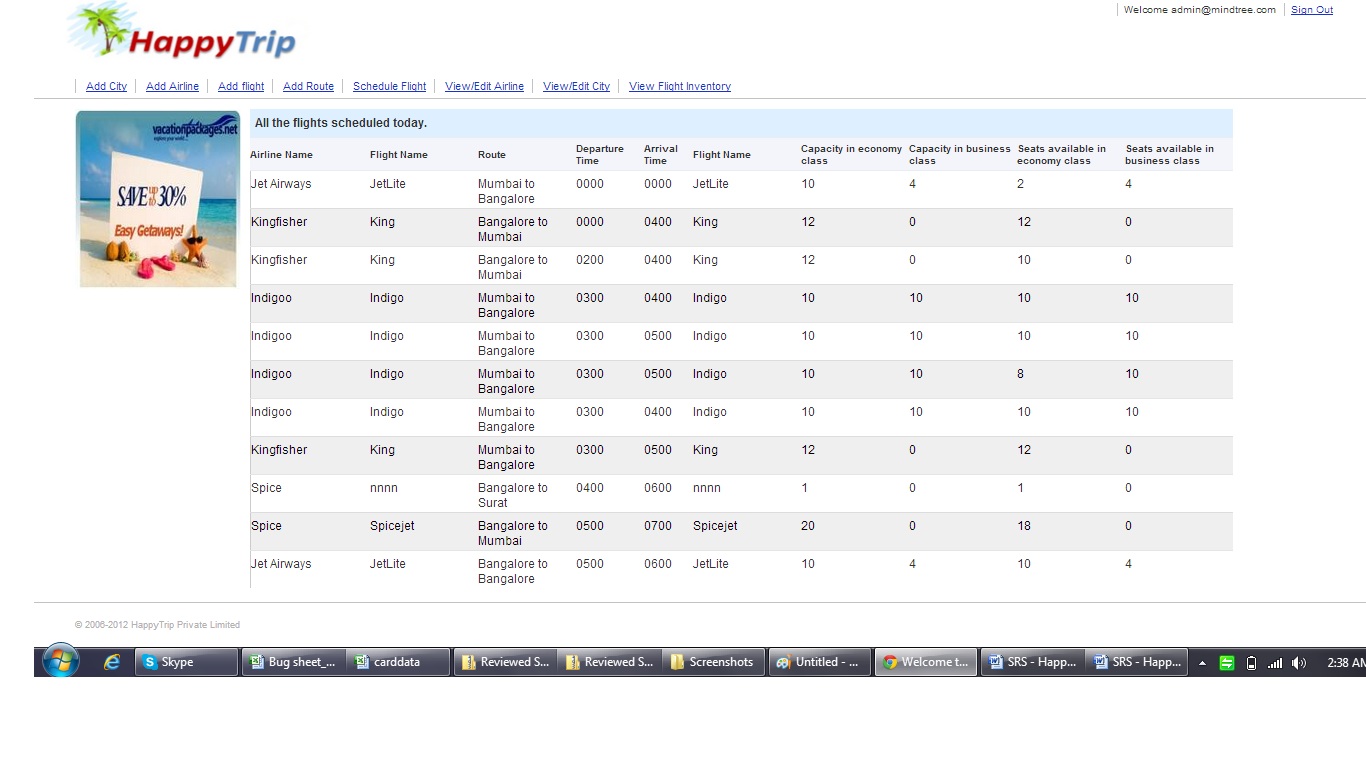
#### Admin – View/Edit City



#### Admin – Add Schedule

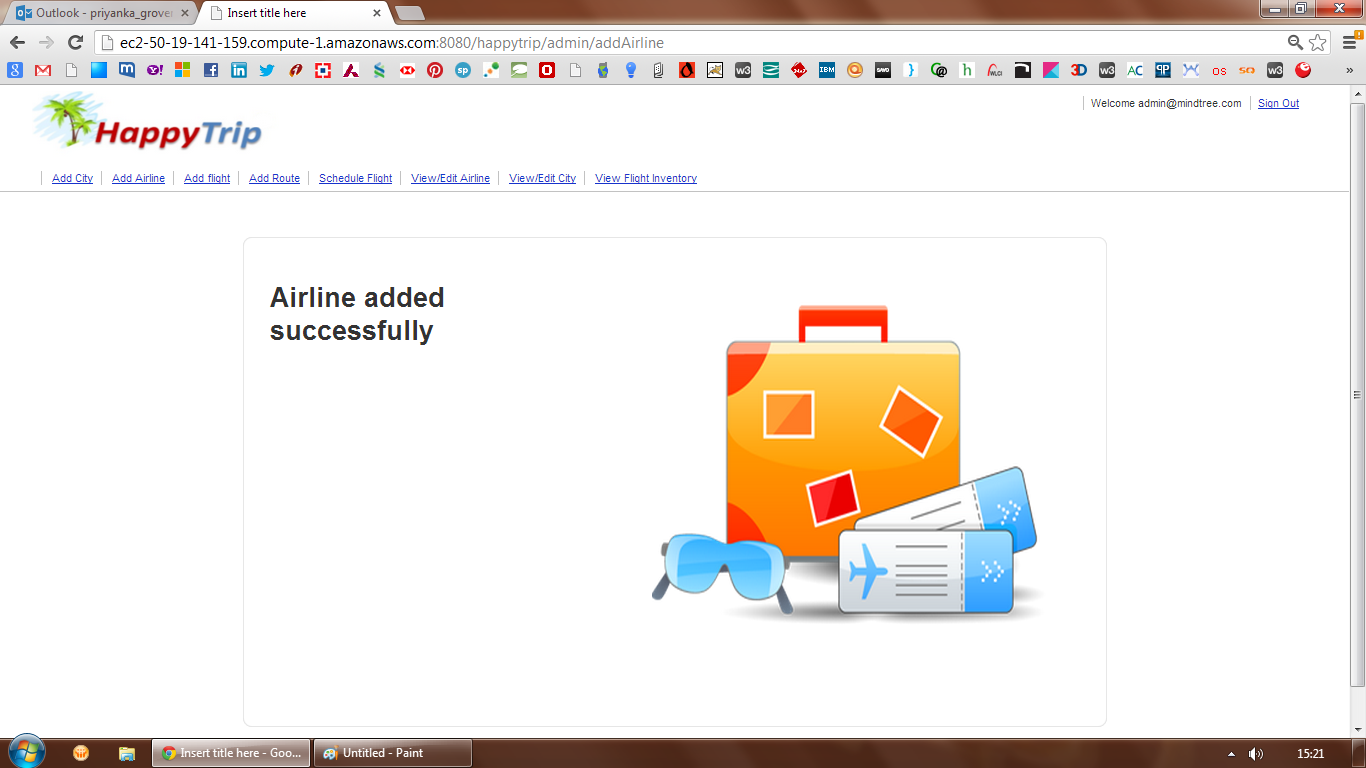


#### Admin – View Inventory



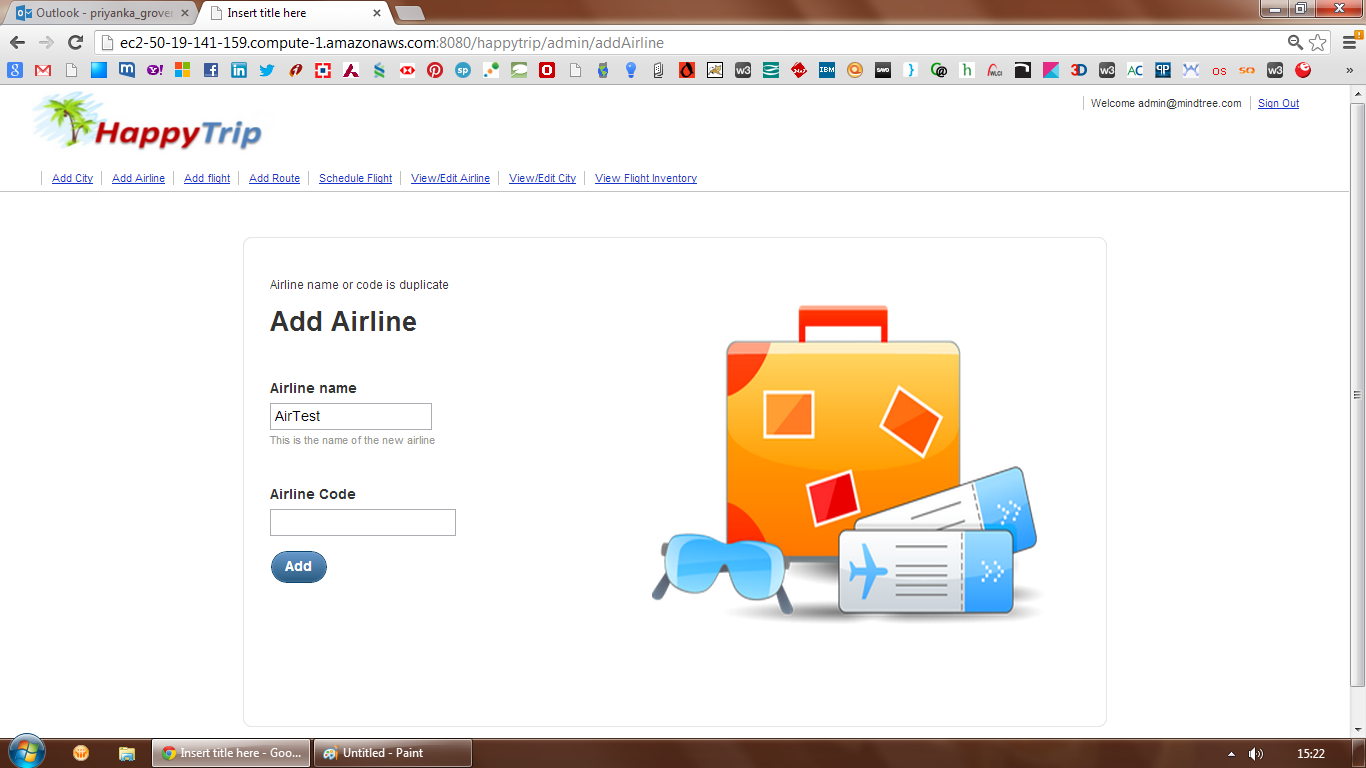
#### Admin – Successful Add

The following type of page should be displayed to the user, when any of the admin modules executes successfully & add of a new item is successful.



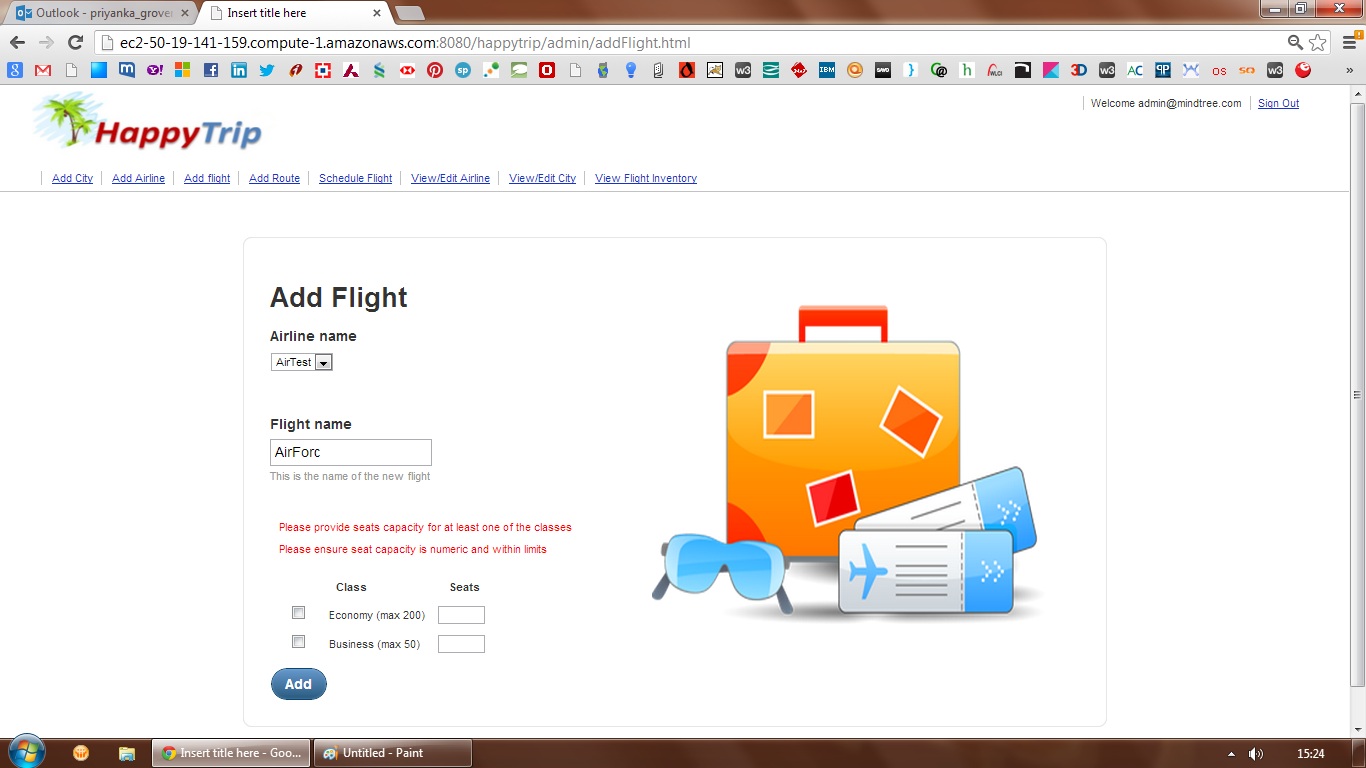
#### Admin – Failure in Add

The following type of page should be displayed to the user, when any of the admin modules executes with failure & add of a new item has failed.



#### Page Validations

Any page validations at field level, should be displayed around, the particular field, and displayed in a similar way as below:



### Non Functional Requirements

#### Browser compatibility

The application will be best viewed in Google Chrome browser

#### Performance Requirements

The performance requirements for this application are currently limited to benchmarking the response time of the application for mainly two pages, as follows:

1. Home Page: The maximum time taken for home page to load, with 50 concurrent users trying to access this page, should not exceed 10 seconds.
2. Search Results Page: For any search criteria specified, which has at max 10 matching schedules, the time taken for search results page to load, with 50 concurrent users trying to access this page, should not exceed 30 seconds

#### Session Management & Security

The requirements for session management & security in this application are related to roles defined in the application. As detailed above, the three actors, Guest User, Registered User & Admin User, should have session management with respect to the user role. At no time, one user should be able to view the other user’s session related data, features & access privileges.

#### Usability

This subsection specifies the following requirements concerning the ease with which the system can be used.

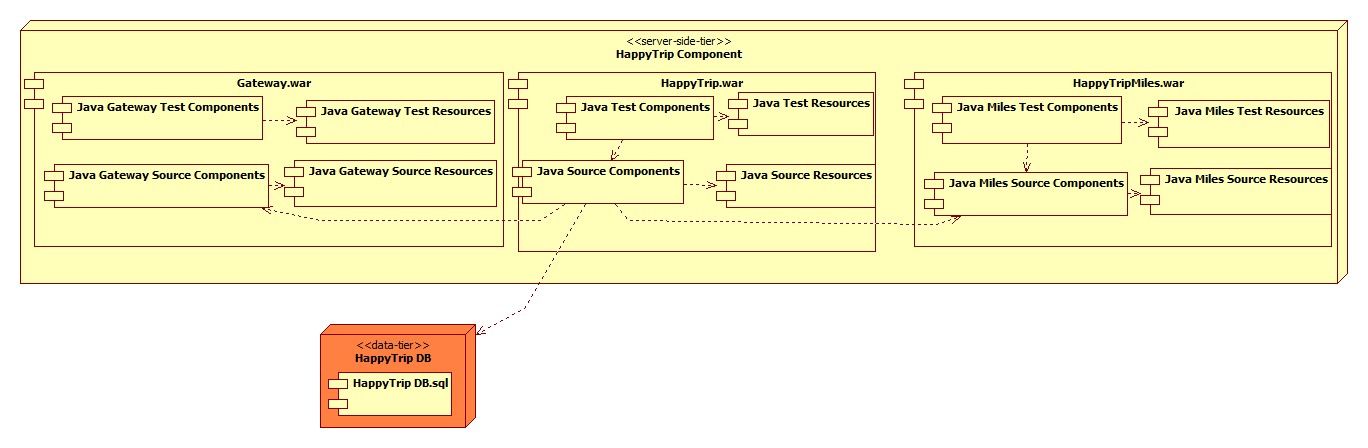
* 1. Consistent Web based UI
  2. Easy to use and navigate
  3. Intuitive – With Tool tips
  4. Colors should be pleasant to the eye of the user
  5. Provide easy access by implementing Ajax and avoid page refresh wherever possible

# Architecture and Design

The section documents the major architecture and design of the system.







# Database Design

# C:\Users\SONY\Downloads\HappyTrip DB.png

# Known Issues

The section documents the following known issues existing in the system:

|  |  |
| --- | --- |
| S.No. | Description |
| 1 | An admin can sign in via registered user sign page as well,and vice-a-versa is also true |
| 2 | No checks done to see business class ticket is higher priced than economy |
| 3 | No checks on passenger details page to see if title is "mr" then gender is mandated as "Male" |
| 4 | A flight can be scheduled for same time, on a different route |
| 5 | There are no business rules defined on the validations for password format, min,max length, etc. |
| 6 | If back button of browser is clicked from the confirmation page, application loses all session data and cannot display the data. We need to click on the home page logo. |
| 7 | There are no business rules defined for field validations for limit on the number of characters, however it is ensured that field lengths & data types are matching their DB lengths. And there is no restriction on entering special characters and they are handled in the application to ensure no breaks in application |

# Future Enhancements

The section documents the following envisioned future enhancements:

* The HappyTrip in future will support IE
* There is plan to create a mobile version of the application, but a feasibility study needs to be done. Also we need to analyze the usage patterns

# References

The section documents the following references useful to understand this document:

1. “HappyTrip.com Project Requirement.pdf”
2. “HappyTrip-Architecture.uml”
3. “HappyTrip-Booking Activity.uml”
4. “HappyTrip-UseCase.uml”
5. “data model.png”