MONASH University

FIT5032 Design Report (Major Application Development Credit/Distinction/High Distinction)

BookMyFlights

Chola Naga Sai Raghu Varma Kallepalli:29876478

Contents Page

1. Overview

The name of the application is 'BookMyFlights'. The application allows users to search for flights, book flight tickets and manage their bookings, get flight details like availability, fares, also allow users reschedule or cancel flight booking.

2. User stories and Use case diagrams

In this application there are 2 types of user

a) Admin b) User

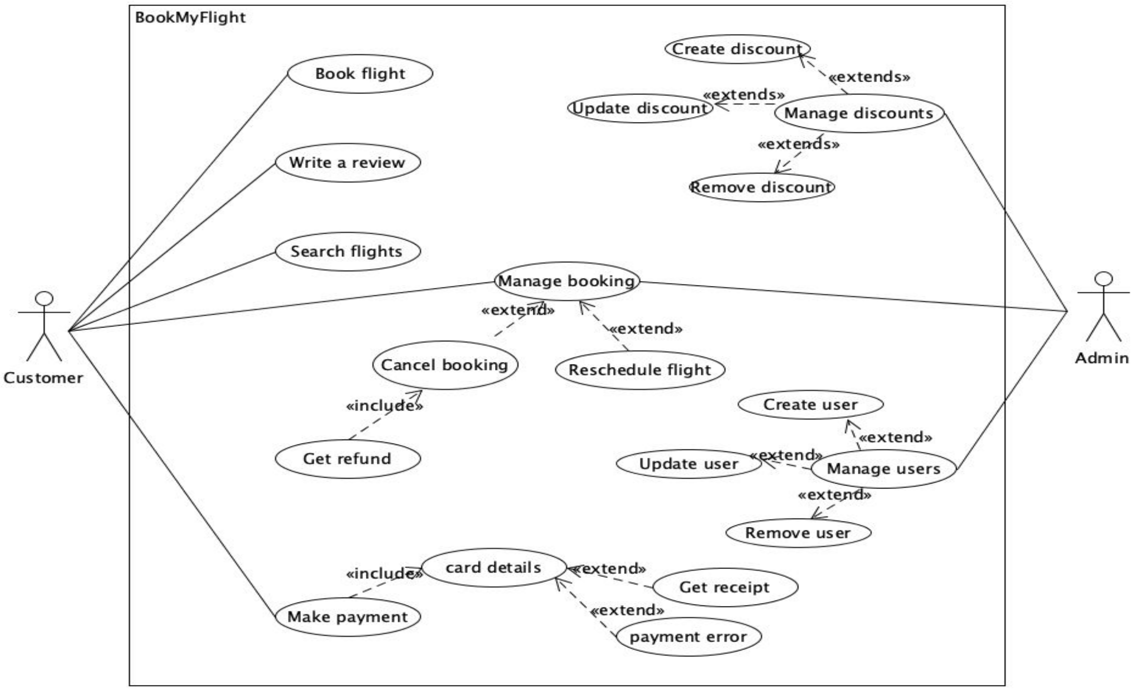
Admin stories:

- As an admin, I want to manage(CRUD) the users, flights, bookings and review.
- As an admin, Iwant to book flight tickets upon request by user.
- As an admin, I want to show the office location on a map to the user. So that they can get some help.
- As an admin, I want users to send an email if they have any simple issues with the site.
- As an admin, I want to show the appropriate menu to reduce the ambiguity. Like, not showing the booking option when user is not logged in.

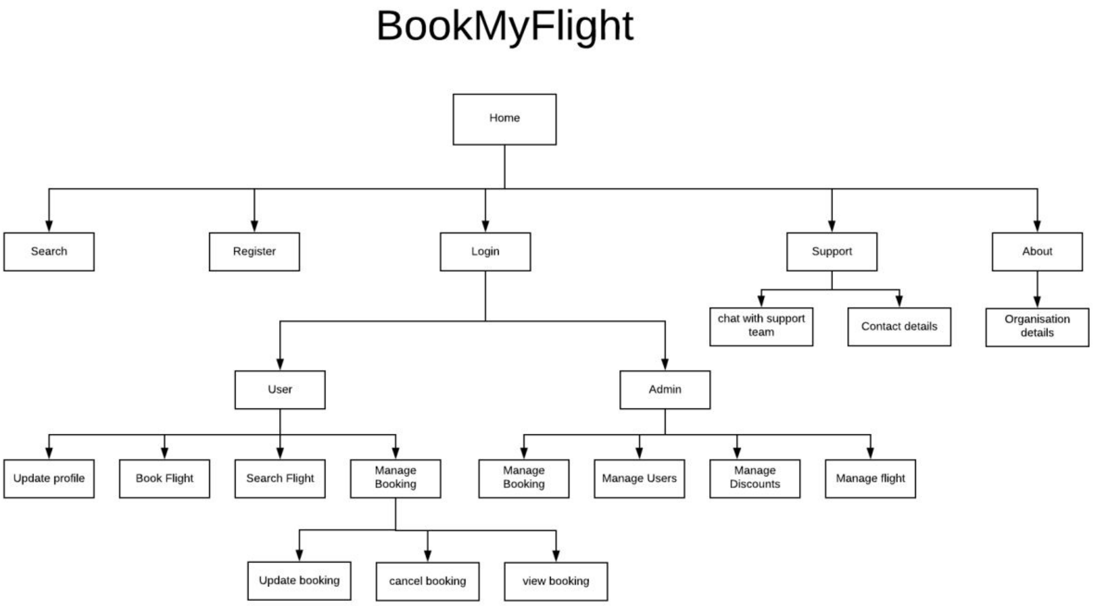
User stories:

- As a user, I want to book flight tickets using the web application
- As a user, I want to update or cancel the bookings I made
- As a user, I want to give feedback of my bookings so I can provide review about the airline
- As a user, I want the interface to be simple so I can navigate the application with no complications

Use case diagram:



3. **Functional diagram**



4. **Usability Design Review**

Navigation: All the main functionalities are embedded in the navigation bar so that the user can go wherever he wants with ease.

● Example: The navigation bar is available throughout the application no matter which page the user access.

Familiarity: The application displays the message in concepts familiar to user instead of using system/technical terms

● Example: Instead of showing Error:404 it should exactly tell the user what the error was and possible solution

Consistency: Making sure that the user is not wondering for actions that does the same thing.

● Example: Buy vs Purchase, here both terms mean the same action but, user has to thick for a second whether they do the same thing. So, using the same word throughout the application for user comfort

Error Prevention: This application tries to prevent errors rather than showing a good error message in the first page.

● Example: While entering registration form, if a user forgets to enter a field, the application prompts the user to fill the entry. Also, asks the user for confirmation if they are performing any sensitive tasks.

Feedback: The user gets response from the application if they perform any tasks showing the status or progress of task

● Example: It the user registration was successful, the application responds to the user by showing a successful message.

Visual Clarity: The application presents the data in natural and logical way.

- Example: Like showing the error message in red colour, success message in green colour.
- Registration and login items are placed away from normal items

Flexibility: The application is designed in a way that a new user can also navigate through different sections and find the information he needs.

5. Checklist of site functionality

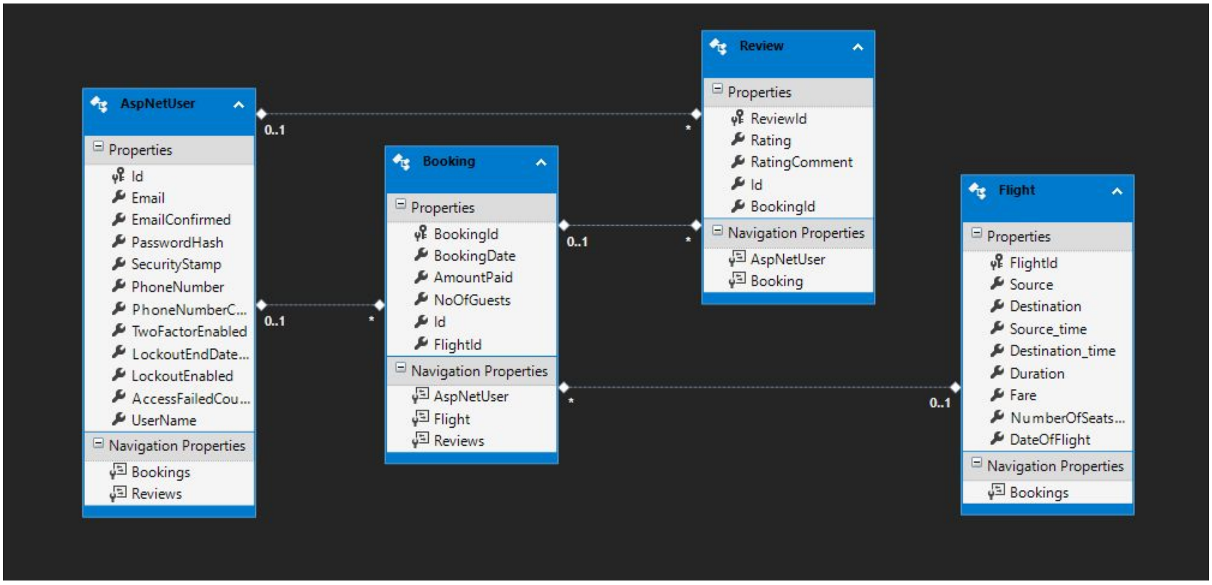
1. (Layout Page)	**TICK if complete**
Good Design	Complete
Stylesheet	Complete
JavaScript	Complete
Menu	Complete
2. (Home page)	
Design and content	Complete
Banner Image	Complete
3. (User Log in)	
Web form and validation controls	Complete
Formatted data entry display	Complete
Overall page design	Complete
4. (Customised Views and Controllers)	
Customised Views	Complete
Customised Controllers	Complete
Other customisations	Complete
5. (Documentation)	
Code Comments	
Attribution of Source of any code used	
6 Business Requirements	
BR(A1): for C to C+	Complete
BR(A2): for C to C+	Complete
BR(B1): for C to C+	Complete
BR(B2): for C to C+	Complete
BR(C1): for C+ to C++	Complete

BR(C2): for C+ to C++	Complete
BR(C3): for C+ to C++	Complete
BR(D1): for D to D++	Complete
BR(D2): for D to D++	Complete
BR(D3): for D to D++	Complete
BR(D4): for D to D++	Complete
BR(E1): for HD to HD+	Complete
BR(E2): for HD to HD+	Complete
BR(E3): for HD to HD+	Complete
BR(E4): for HD to HD+	Complete
BR(E5): for HD to HD+	Complete
Audit	
No breaking of copyright	

6. Selected approach when constructing the application

I’ve developed this application using Database First approach

7. Entity Relation Diagram



8. Data dictionary

Flight

Attribute	Datatype	Example	Description
FlightId	nvarchar	D7212	This is used as a primary key to identify or retrieve the data from the Flight table. Since, the flight id is a combination of both number and string I used nvarchar
Source	nvarchar	Melbourne	The source name of a flight is a string. So, I used nvarchar as the data type for Source.
Destination	nvarchar	Sydney	The destination name of a flight is a string. So, I used nvarchar as the data type for Destination.
Source_time	time	10:00	It is used to store the time of the flight. I used time datatype to store the time and also it can be useful to make queries relating to date in future.
Destination_time	time	11:00	Since, it is used to store the time of the flight. I used time datatype to store the time and also it can be useful to make queries relating to date in future.
Duration	time	01:00	I used time datatype to store the duration of the flight.
Fare	float	$70.45	The fare for the flight can be a decimal as well. So I used float to store the Fare.

NumberOfSeats Left	int	150	The number of seats cannot have any decimal number init. So I used int to store the value. Also, when a booking is made the number of seats are modified accordingly and checks for seats available in future for other bookings
DateOfFlight	date	10/10/2019	This attribute stores the date of the flight.

Booking

Attribute	Datatype	Example	Description
BookingId	nvarchar		This is used as a primary key to identify or retrieve the data from the Flight table. Since, the flight id is a combination of both number and string I used nvarchar
BookingDate	date	10/10/2019	This attribute stores the date of the booking made.
AmountPaid	float	$70.45	The fare for the flight can be a decimal as well. So I used float to store the Fare.
NoOfGuests	int	150	The number of guests cannot have any decimal number init. So I used int to store the value.

Review

Attribute	Datatype	Example	Description
ReviewId	nvarchar	r1	This attribute is used as a

(Primary key)			primary key for this entity. It is used as a unique key to get the row information. The data type used is nvarchar because if we use Int(number) type it would not be suitable if there are many reviews.
Rating	Int	1,2,3,4,5	Since we are storing a number for rating that is in range 1-5. Int would be suitable for holding the data
Comment	nvarchar	Nice, Excellent, Awesome, etc	It is used to store the comments made by users about the airline. So for storing words it is best suitable to use nvarchar datatype.

9. Development Methodology

For the development of BookMyFlights, agile methodology is being used for continuous integration of new features. Such as business requirements and features requested by user and admin. So, using agile methodology, number of features are being added using different sprints. Also used Test-driven development for process that relies on the repetition of a very short development cycle. I have divided all the business requirements to a very specific and short test cases to make the development easier. The new features are added upon successful of the tests pass.

10. Versioning

For version control I've used Git to keep track of all the changes and big modifications. Whenever there is a new feature added, I've pushed the changes to the Git so that if something goes wrong in the future while adding a new feature, I can get back to the old version at anytime as I wanted.

Repository link: https://github.com/srvarma7/IADwithDB