### **HOTEL RESERVATION**

### **SYSTEM**

**Requirements Specification**

**and**

**Analysis Document**

26.06.2016

P.MELİS DELİPINAR

YAĞIZ ŞENER

ONUR CİMŞİT

ORHUN BÜYÜKAYDIN

**Prepared for**

**SE301 Software Engineering**

****

Table of Contents

[1. Introduction 3](#_Toc453932853)

[1.1. Purpose of the System 3](#_Toc453932854)

[1.2. Scope of the System 3](#_Toc453932855)

[1.3. Objectives and Success Criteria of the Project 3](#_Toc453932856)

[1.4. Definitions, Acronyms, and Abbreviations 4](#_Toc453932857)

[1.5. Overview 4](#_Toc453932858)

[2. Current System 4](#_Toc453932859)

[3. Proposed System 5](#_Toc453932860)

[3.1. Overview 5](#_Toc453932861)

[3.2. Functional Requirements 5](#_Toc453932862)

[3.3. Nonfunctional Requirements 6](#_Toc453932863)

3.3.1 [Usability 7](#_Toc453932864)

3.3.2 [Reliability 7](#_Toc453932865)

3.3.3 [Performance 7](#_Toc453932866)

3.3.4 [Supportability 8](#_Toc453932867)

3.3.5 [Implementation 8](#_Toc453932868)

3.3.6 [Interface 8](#_Toc453932869)

3.3.7 [Packaging 8](#_Toc453932870)

3.3.8 [Legal 8](#_Toc453932871)

[3.4.System Models 9](#_Toc453932872)

3.4.1 [Scenarios 9](#_Toc453932873)

3.4.2 [Use case model 13](#_Toc453932874)

3.4.3 [Object model 13](#_Toc453932875)

3.4.4 [Dynamic model 15](#_Toc453932876)

3.4.5 [User interface—navigational paths and screen mock-ups 20](#_Toc453932877)

[4. Glossary 26](#_Toc453932878)

[5. References 27](#_Toc453932879)

**1. INTRODUCTION**

The following section provides an overview of the derived Requirement Analysis Document for the subject “Hotel Reservation System”. First of all, the aim of the analysis is presented and written as understandable for client, users and professions. Purpose, scope of the system, objectives and success criteria are specified below. This document also includes functional, non-functional requirements, system models, glossary and references.

**1.1 PURPOSE OF THE SYSTEM**

The main purpose of the system is, providing a quality reservation service for the users who are looking for a hotel. The system is beneficial for users, search and reserve a hotel room or cancel his/her reservation over the internet at any time. This system also gives to users a choice to compare hotels with their ratings, which are given by the previous guests. By this way, users can select the best hotels for their budget and their vacation or business purposes.

**1.2 SCOPE OF THE SYSTEM**

The system that we are currently developing is an online hotel reservation system, which is called “Hotel Reservation System”. This system aims to involve the hotels that are registered to our system and the users who are searching for any hotel. In general, system has 3 major participants; hotels, users and an system administrator. For hotels and users, the system usage is free to use but hotels and users have to register on the system. For users, signing up is not mandatory for searching any hotel in the system. A user can search a hotel, look their name, state and city. However, for a reservation attempt, signing up is needed. All the membership control is done by the system administrator.

**1.3 OBJECTIVES AND SUCCESS CRITERIA OF THE PROJECT**

Turkey is a holiday paradise, that’s a fact. Also, Turkey is a common point for a lot of businesses. Every day many people are looking for a hotel for a vacation or business reasons. That’s why a system can be really useful to facilitate booking procedures and we are trying to achieve it with “Hotel Reservation System”. The main objective of our project is to provide an efficient and secure reservation. A visitor to our site must select the best hotel for his/her need with minimal effort and maximum satisfaction. Other than this, the system should ease the communication between user to hotel. Finally, our purpose is to be the first choice that comes to mind in the booking sector. One day, all the hotels in Turkey will be registered to our database and we will provide the best service to our users.

**1.4 OVERVIEW**

First of all, we are planning our system. When the plan is complete, we are going to make some analysis for some situations. After these, we start to design our website and finally, we are going to implement our codes. That’s when our system will be completed, “Hotel Reservation System” will be served.

**2. CURRENT SYSTEM**

Nowadays, “Booking.com” is the Turkey’s most commonly used online hotel reservation system. It provides service to thousands of people with thousands of hotels around the world. “Booking.com” has been designed as user friendly, because a user can make a successful reservation with just a few clicks. Also, it is free to sign up and search hotels. We are able to examine the functions of user side, but we need to remember that, “Booking.com” also has a hotel side with many hotels registered to their system. There are many hotel reservation systems on the internet, which are remarkably similar to the “Booking.com”.

In our system, a user can search for hotels for free without signing in. Signing up is needed only for making a reservation. A user can go to the sign up page, which is located right at the top, to fill his/her information data to be able to make a reservation. Also, users can check their own past/future reservations information and any reservation in the future can be cancelled by the user. Finally, a user can comment on his/her past reservations by an admin approval.

When we look at the hotel side of the system, hotel owner needs to sign up. When the signing operations complete, hotel owner can see their hotel information page and reservations page. He/she can change the information and the pricing of the rooms in the information tab.

In addition, admin can approve or reject the hotel registrations and also an admin can approve or reject the comments of the users about hotels. Users can be deleted by an admin according to the site rules. All in all, an admin can see the whole information about any user or about any hotel user.

**3. PROPOSED SYSTEM**

Our system is designed as a “as-is” system. The purpose of the system is, let users make online reservations in easy few steps. Also users are able to surf on the system and explore new hotels. In the system, login is mandatory to give a reservation, but user can surf on the site without login. Signing up and making reservation is free. Users are also able to view their all past reservations .In addition to this, users could comment and rate the previous reservations soon after the vacation is finished. The system interface will be designed as quite simple, target-driven and easy to use. To user side, we take E-booking as an example, to hotel side we keep the system much more basic than E-booking due to the our limited resources. Hotels are able to control their own reservations.

**3.1. OVERVIEW**

The system will be designed to be user friendly. It aims to browse the hotels which are adaptable for user preferences easily and let the user to make reservation with just a few clicks. Also hotels are able to control their own reservations. Therefore, the system’s basic and user friendly interactive interface helps both user and hotel. The functions are specified by considering these issues.

**3.2. Functional Requirements**

Functional requirements specify the abilities and functions that a system must have. The functional requirements of this online reservation system includes three actors and their functions that are;

\* User Functions

\* Hotel Functions

* System Admin Functions.

**User Functions**

|  |
| --- |
| **Function Description** |
| The user is able to search hotels by their names,states,cities. |
| The user can set check-in/check-out date that s/he select. |
| The user can makes multiple reservations at the same time. |
| The user is able to comment and rate a hotel on their past reservations. |
| The user is able to view the past reservations and cancel new reservation. |
| The user could pay the reservation cost by online payment with credit card. |
| The user must sign in the system with e-mail and password. |
| The user cannot makes reservation before being logged in. |
| The user is able to view promotion hotels. |
| The user can cancel the reservation till one day before that choosen date. |

**Hotel Functions**

|  |
| --- |
| **Function Description** |
| Hotel must inform the system about the reservation has seen. |
| Hotel is able to delete room type option from its own page. |
| Hotel is able to update and add a new information from its own page. |
| Hotel is able to determine if a room is enable or not. |
| Hotel must inform the system if it is opened or closed. |

**System Admin Functions**

|  |
| --- |
| **Function Description** |
| The system admin must add the new hotels to database when a contract is made. |
| The system admin is able to delete a hotel from database when the contract is over. |

**3.3. Nonfunctional Requirements**

A non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific functions.

**3.3.1 Usability**

|  |
| --- |
| **Function Description** |
| The user is able to reserve a room from a hotel with just few clicks. |
| When in online payment function, system must identify the credit card type from the first four digits of the credit card. |
| System warns the hotel about the new completed reservation by an acknowledgment by mail. |
| User cannot sign up the system without e-mail, password and phone number informations. |

**3.3.2 Reliability**

|  |
| --- |
| **Function Description** |
| A system crash should not seen with any reservation lost. |
| The user password length must be between 8 and 16 digits with integer and string together. |
| It cannot possible to sign up two times with the same email/username. |

**3.3.3 Performance**

|  |
| --- |
| **Function Description** |
| System allows user to comment a hotel after vacation. |
| System must send reservation information to the hotel . |

**3.3.4 Supportability**

|  |
| --- |
| **Function Description** |
| The reservations that selected by user must be shown in My Reservations Page concurrently. |
| If the hotel not inform the system within the 15 minutes, the system should repeat the acknowledgment voice. |
| If the hotel is not informed by the system about the reservation is done, after one hour later it has seen, the system must warn the hotel again. |
| If the user forget his/her password, system should send an email to user including a temporary password to refresh the password. |
| The system must check the reliability of the user by sending an activation mail just after the user sign up. |

**3.3.5 Implementation**

* The Database is implemented in MySql WorkBench 6.2.
* The Programming is done with Net Beans.

**3.3.6 Interface**

We use interfaces to provide accesses to our system. As we know interface is not just about GUI, it provides us to connect our system and our instances. In our project our interfaces are;

\* Register Page

\* User Home Page

\*Search Hotel Page

\*Reservation Page

\*Hotel Home Page

\*Hotel Registration Form

\*Admin Page

\*Comment Page

\*Hotel Approve-Reject Page

\*UserList

**3.3.7 Packaging**

The system must be available on both of desktop and mobile devices and it performs properly on every browser.

**3.3.8 Legal**

* For users the system is free to use.
* For hotels, the system must be licensed.

**3.4 SYSTEM MODELS**

This section includes scenarios, use cases, sequence diagrams and state machine diagrams of these four functions above mentioned. Also, the whole class diagram and use case diagram is shown below.

In this part, top four crucial functions are also described in detailed way which are;

* **MakeReservation**

In this function, a user searches for any hotel. After compeleting the search, user selects a room type from the list which is listed by the system according to the user’s hotel choice. These steps can be done „**Without“** login to system. After selecting a room type,systems redirects the user to the login page and after logging into system, system shows a form and user fills this form including the payment. However, Finally, making reservation is completed by confirming the reservation by user.

* **HotelMenuOperation**

In this function, hotel user clicks to „Hotel Request Form“. Hotel fills the mandatory fields in this form about the hotel’s details. After filling the fields, hotel user sends the form for admin’s approval by clicking the „Confirm“ button.

* **SeeMyReservation**

A user clicks the „See My Reservations“ to checks about past and future reservations. After clicking the button, system shows a list about user’s all reservations that are stored in the system. Future reservations can be cancelled by the user. However, past reservations can not be changed and cancelled by the user. User clicks any listed reservation and see more detailed information about any reservation.

* **HotelApprove**

Admin clicks for „Waiting Approval“ button and selects any hotel from the list. Admin checks the information about selected hotels and clicks to „Approve“ button and saves the selected hotel to the system.

**3.4.1 SCENARIOS**

**MakeReservation Scenario**

|  |  |
| --- | --- |
| *Scenario Name* | MakeReservation |
| *Participating Actors* | Markus |
| *Flow of Events* | 1. Markus searches hotel without login to system.  2. Skynet shows a list of hotels.  3. Markus selects any hotel from the shown list.  4. Skynet shows the selected hotel’s attributes, room types and prices according to user’s input.  5. Markus selects a room type from the shown list and clicks “Book” button.  6. Skynet redirects the user to login page.  7. Markus inputs his/her login details, username and password, to continue booking.  8. Skynet checks the username, password and usertype matching according to the user’s input. If the inputs are valid, system redirects user to the reservation details.  9. Markus fills the mandatory fields to complete reservation including the payment period.  10. Skynet checks the validation of user’s input. If the inputs are valid, the system shows a reservation information page. |
| *Entry Condition* | Markus search for a hotel. |
| *Exit Condition* | Markus completes the reservation. |
| *Quality Condition* | 1. Markus cannot make multiple reservations at the same time.  2. Markus can cancel a reservation till one day before the reservation date.  3. Markus fills the login details incorrectly. |

**HotelMenuOperation Scenario**

|  |  |
| --- | --- |
| *Scenario name* | HotelMenuOperation |
| *Participating actors* | Isik Hotel |
| *Flow of events* | 1. Isik Hotel opens the login page and fills the login information. Hotel clicks the “Login” button.  2. Skynet checks the information and accepts the login request and shows a form that needs to be filled by the The Hotel.  3. Isik Hotel fills the form with their own information and The Hotel clicks “Send for Approval” button.  4. Skynet receives the form and sends it for The Admin’s approval. |
| *Entry Condition* | Isik Hotel logins to the system. |
| *Exit Condition* | Isik Hotel fills the own information. |
| *Quality Condition* | Isik Hotel fills the login details incorrectly. |

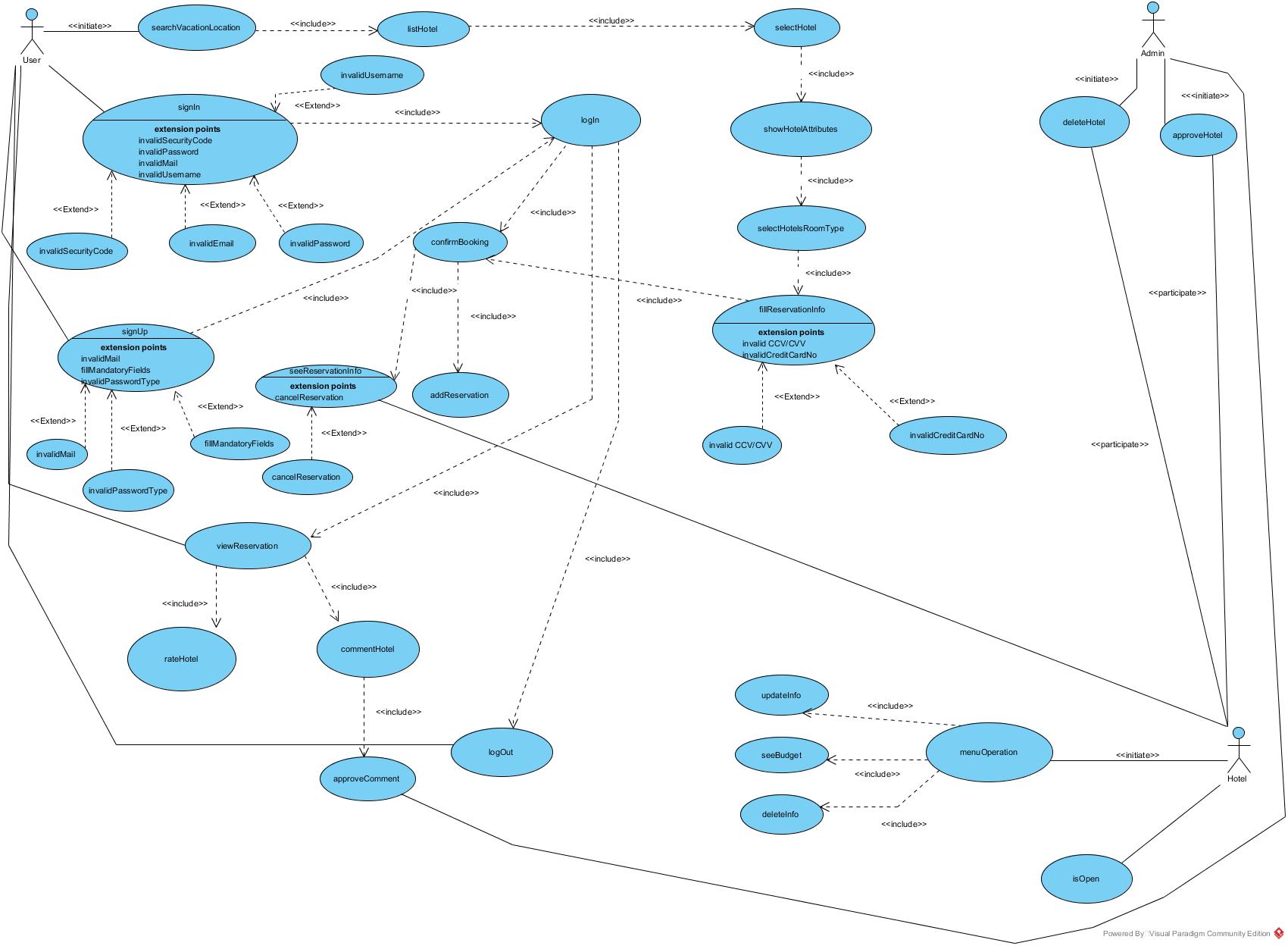
**SeeMyReservation Scenario**

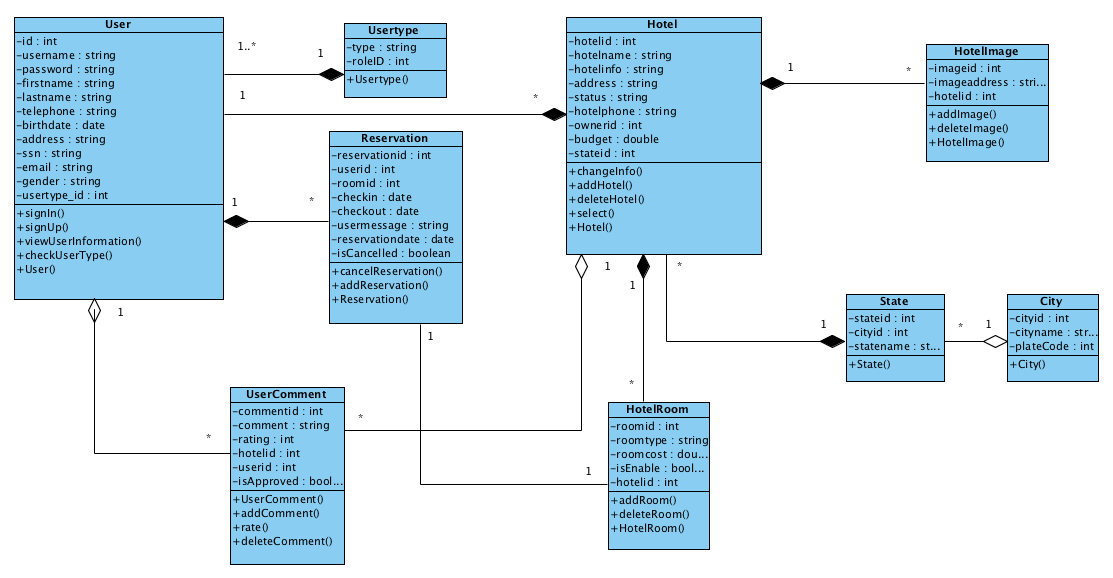
|  |  |
| --- | --- |
| *Scenario name* | SeeMyReservation |
| *Participating actors* | Hodor |
| *Flow of events* | 1.Hodor logins to the website by filling his/her password, e-mail address or username.  2. Skynet checks the user password, e-mail address or username and then, redirects the user to the homepage.  3. Hodor clicks to the “See My Reservation” link on the top of the right side of the homepage.  4. Skynet brings the “See My Reservation” page to the user.  5. Hodor checks his/her reservations. |
| *Entry Condition* | Hodor logins to the website. |
| *Exit Condition* | Hodor see his/her own reservation. |
| *Quality Condition* | 1. Skynet shows past and future reservations in different ways.  2. Hodor fills the login details incorrectly. |

**HotelApprove Scenario**

|  |  |
| --- | --- |
| *Scenario name* | HotelApprove |
| *Participating actors* | John |
| *Flow of events* | 1. John enters the web-site.  2. Skynet brings the required username/email and password fields to the screen.  3. John fills the fields of the form and clicks the login button.  4. Skynet checks the user information and redirects the admin to the admin’s homepage.  5. John clicks to the “Waiting for Approve Hotels” link.  6. Skynet brings the waiting for approval hotels to the screen.  7. John checks the hotel information and clicks the “Approve” button.  8. Skynet approves the selected hotels. |
| *Entry Condition* | John logins to the admin page. |
| *Exit Condition* | John completes the hotel approval. |
| *Quality Condition* | John fills the login details incorrectly. |

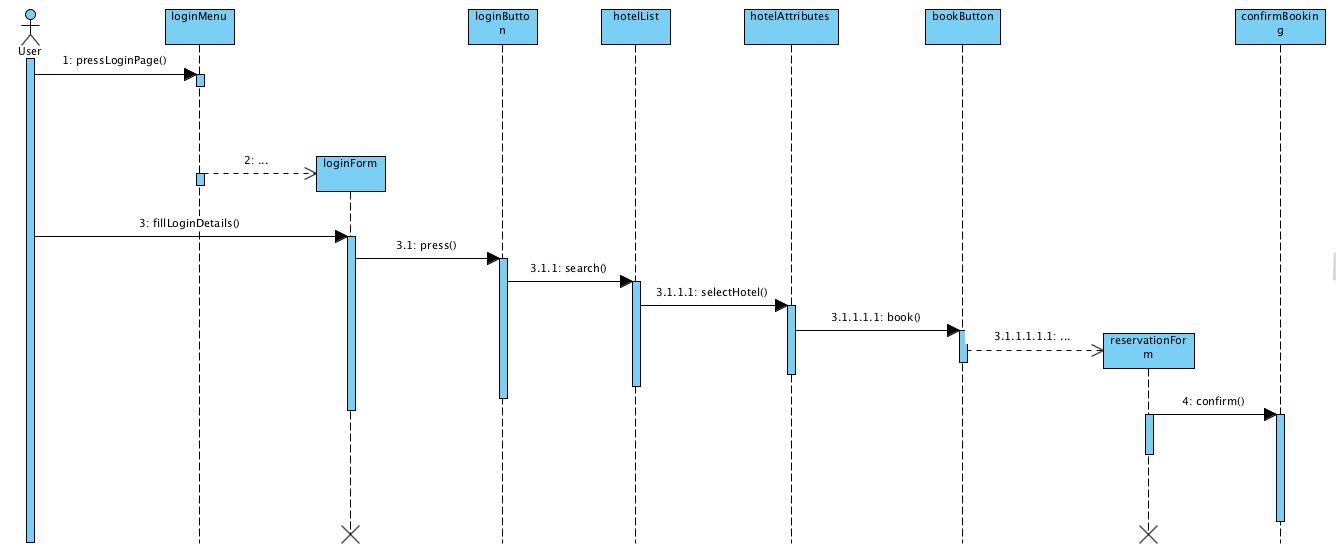
**3.4.2 USE CASE MODEL**

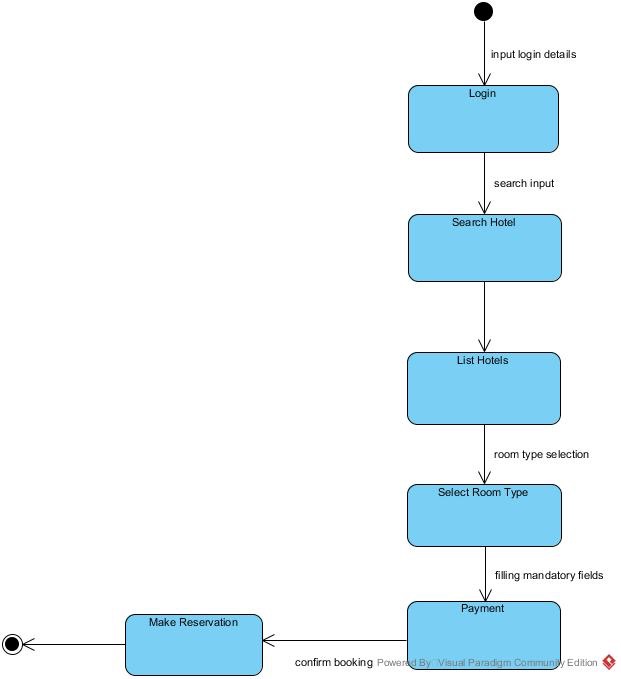
****

**3.4.3 OBJECT MODEL**

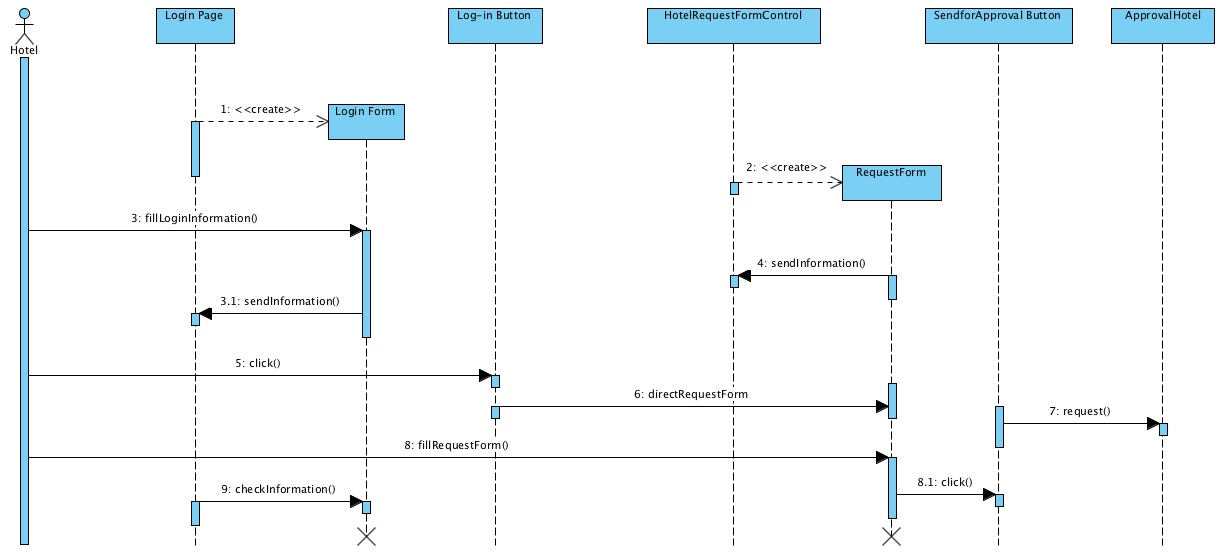
**3.4.4 DYNAMIC MODEL**

**MakeReservation Sequence Diagram**

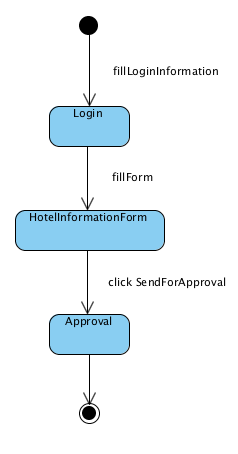
****

**MakeReservation State Machine**

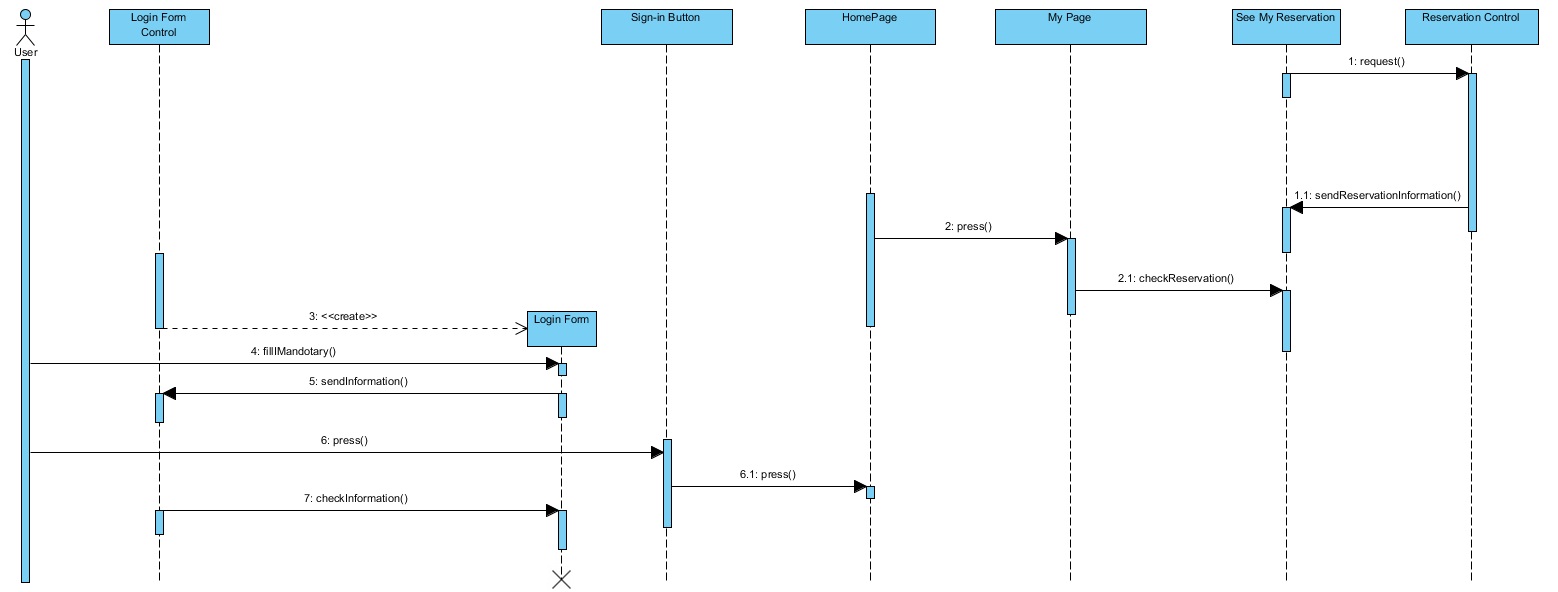
**HotelMenuOperation Sequence Diagram**

****

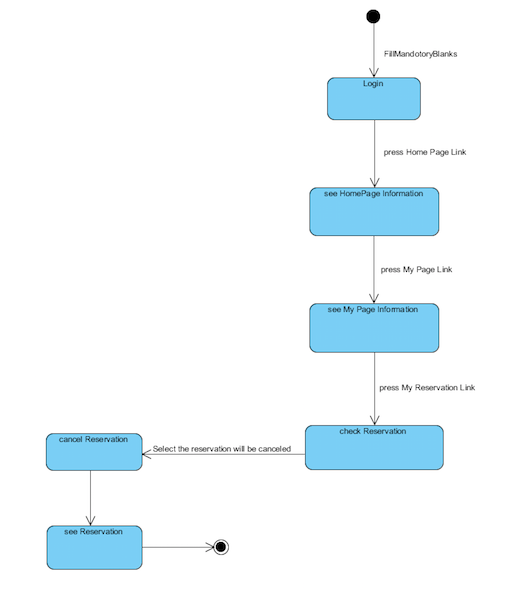
**HotelMenuOperation State Machine**

****

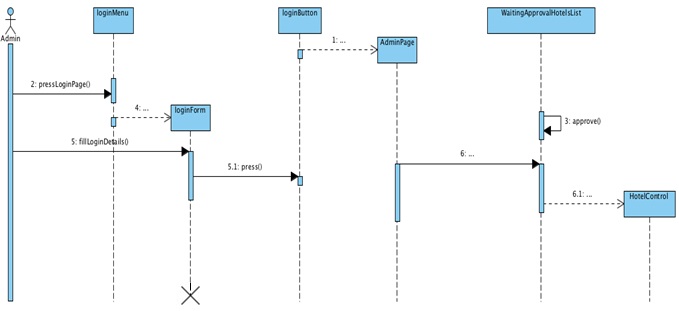
**SeeMyReservation Sequence Diagram**

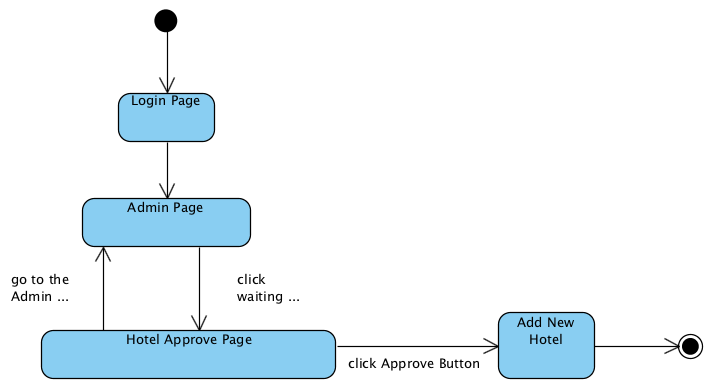
****

**SeeMyReservation State Machine**

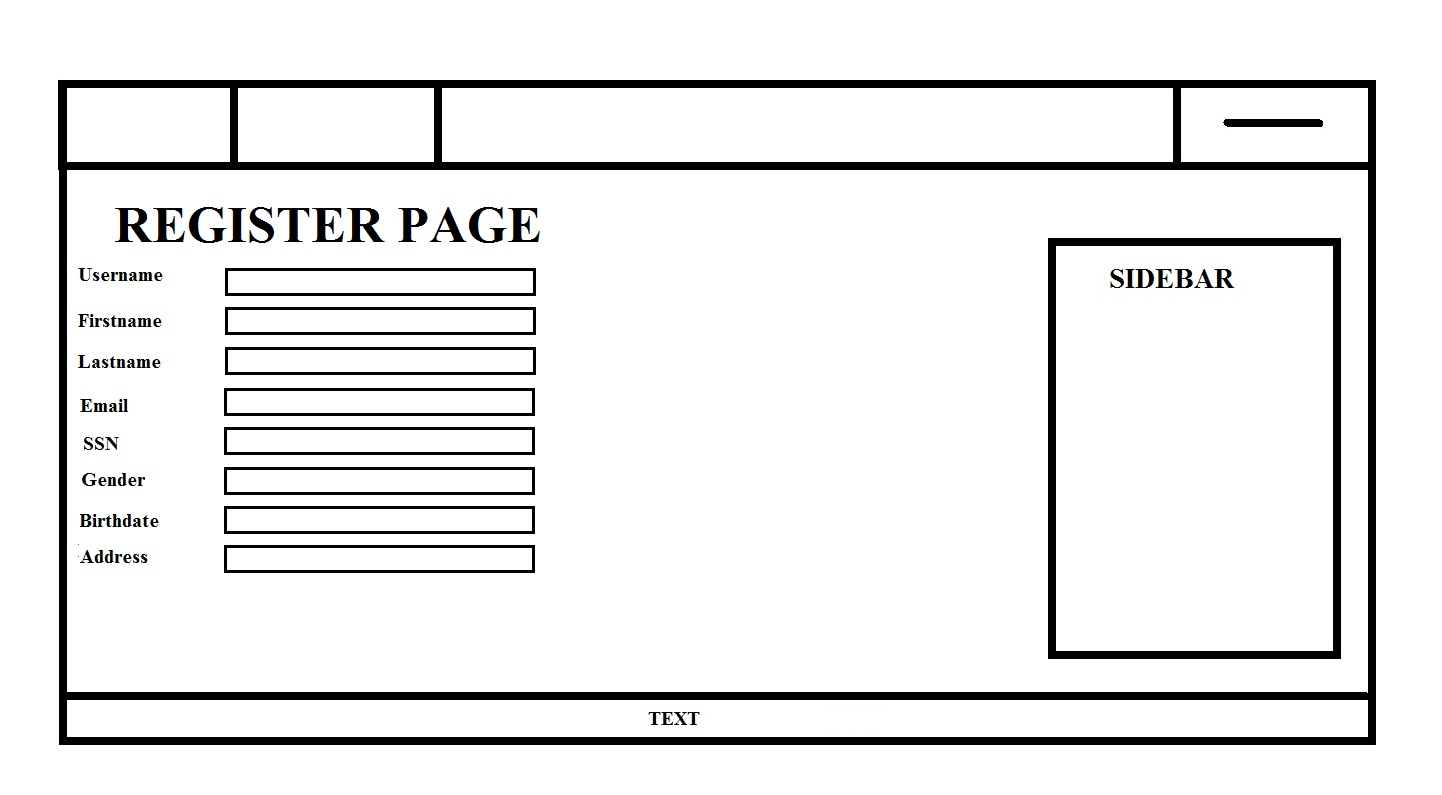
****

**HotelApprove Sequence Diagram**

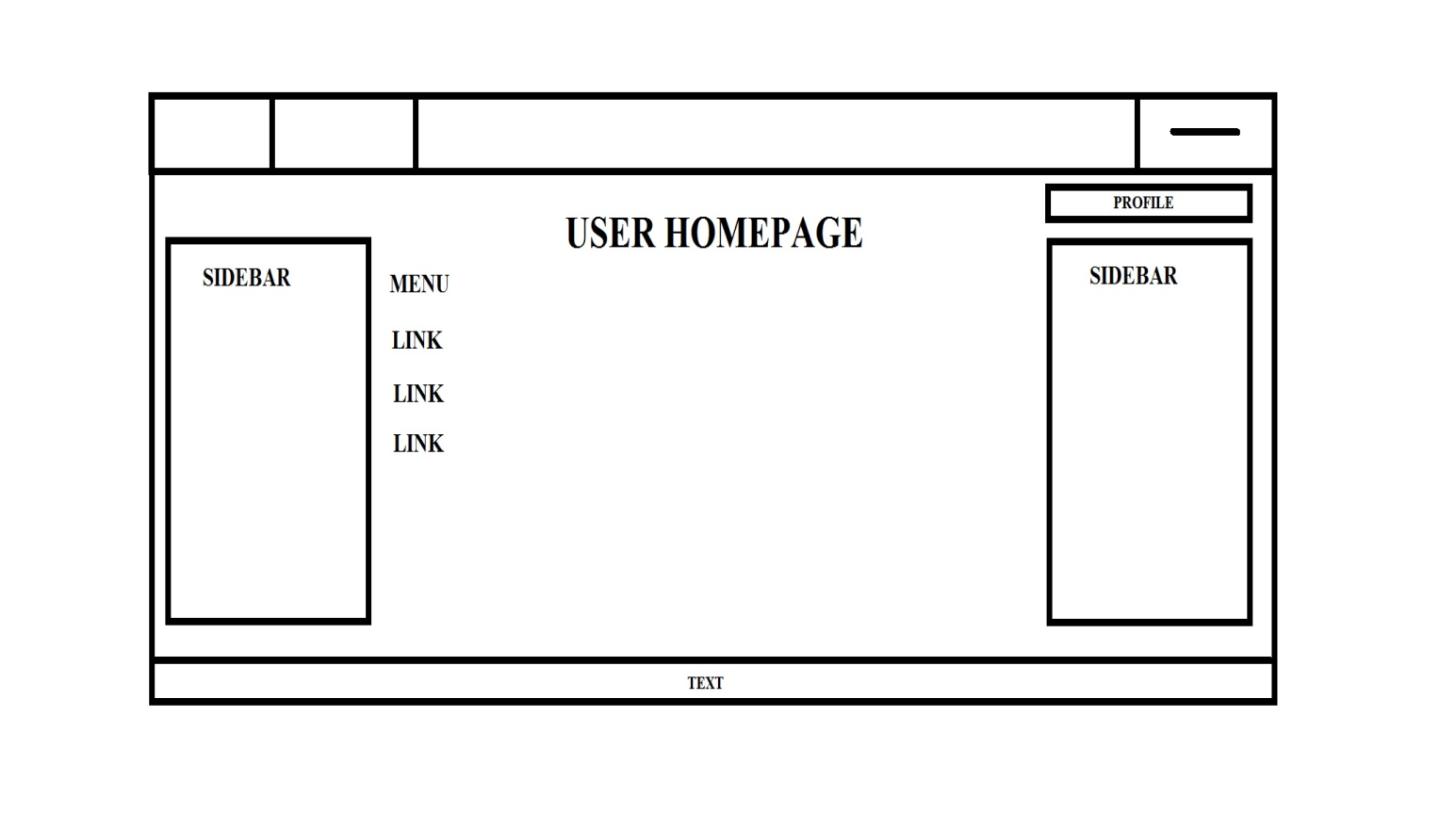
****

**HotelApprove State Machine**

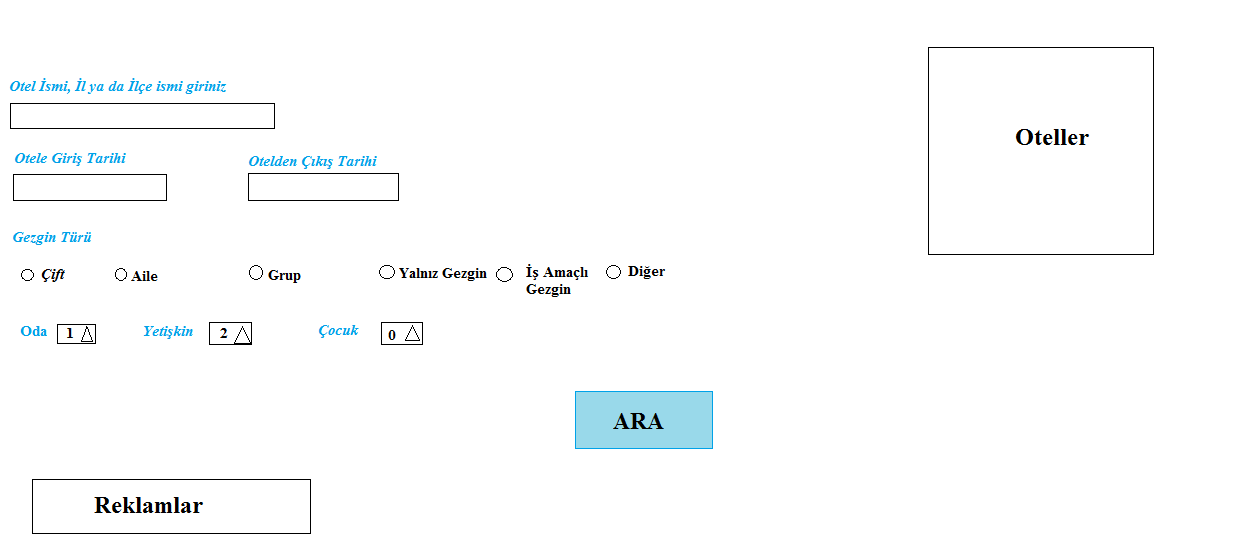
**3.5.5 User interface—navigational paths and screen mock-ups**

**Register Page**

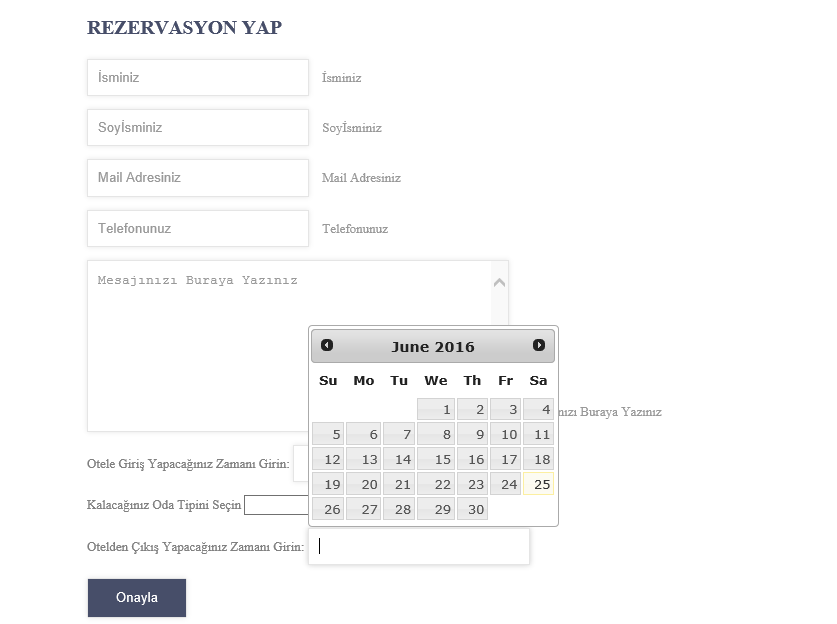
**User Homepage**

****

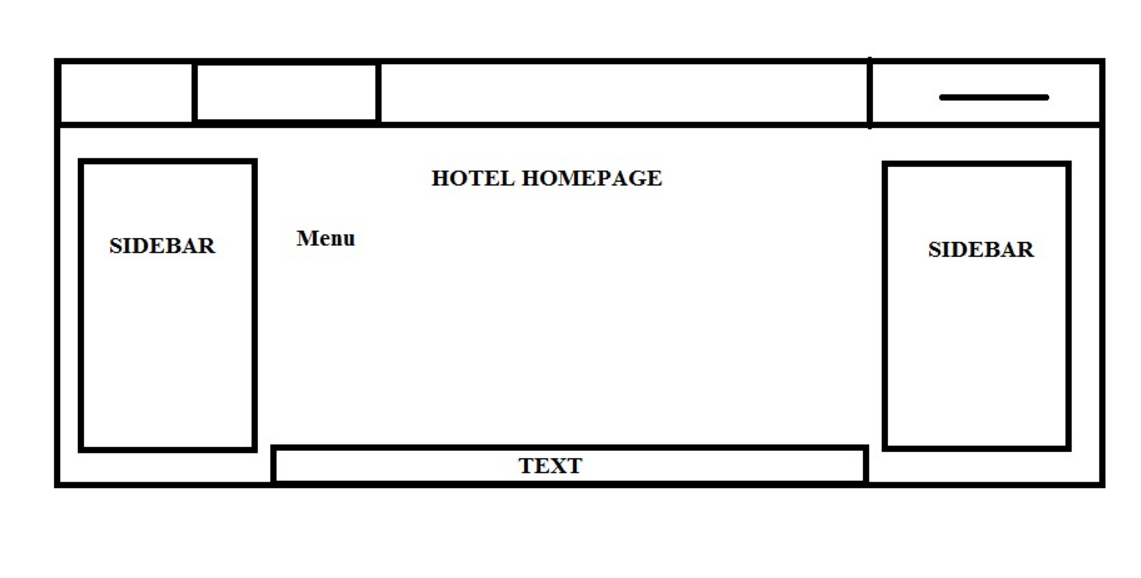
**Search Hotel Page**

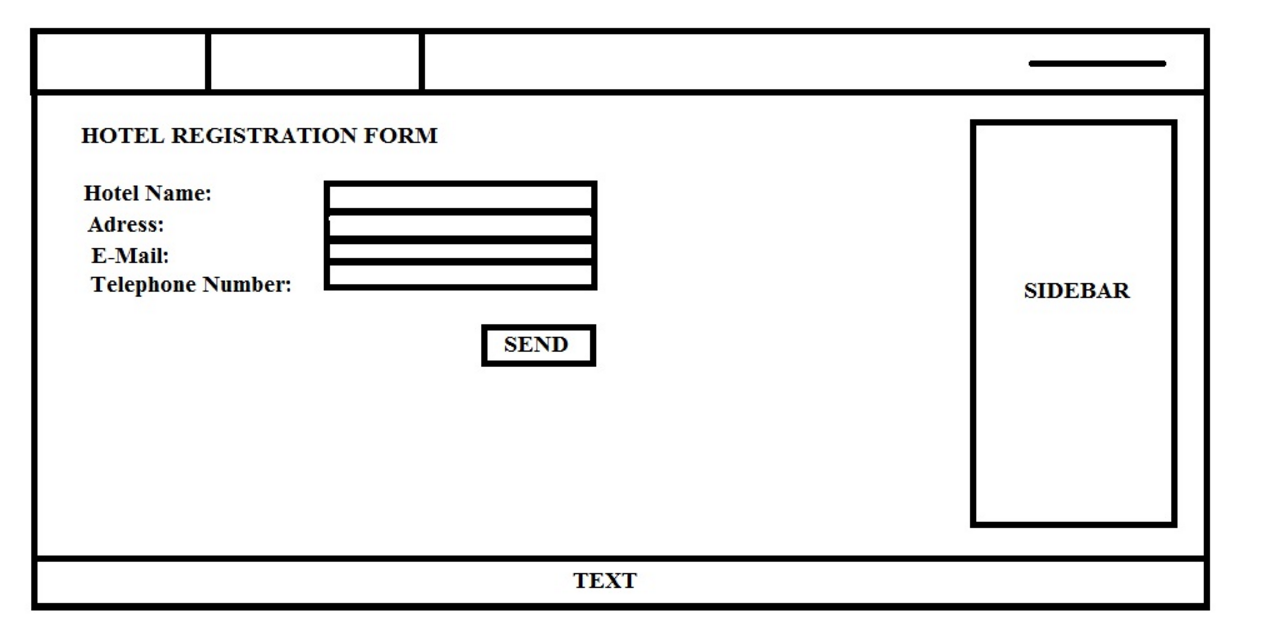
****

**Reservation Page**

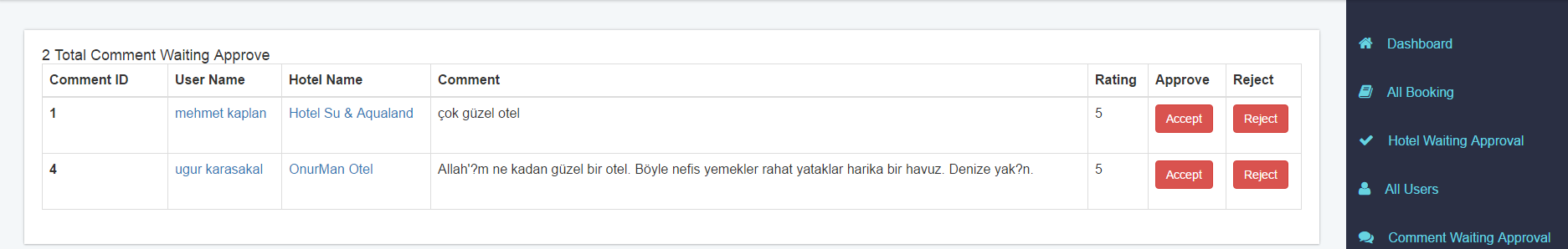
****

**Hotel Home Page**

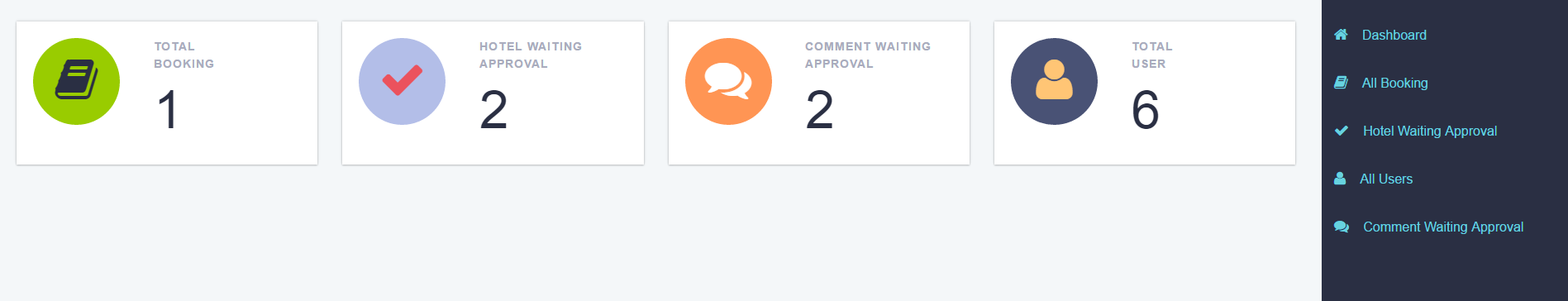
**Hotel Registration Form**

****

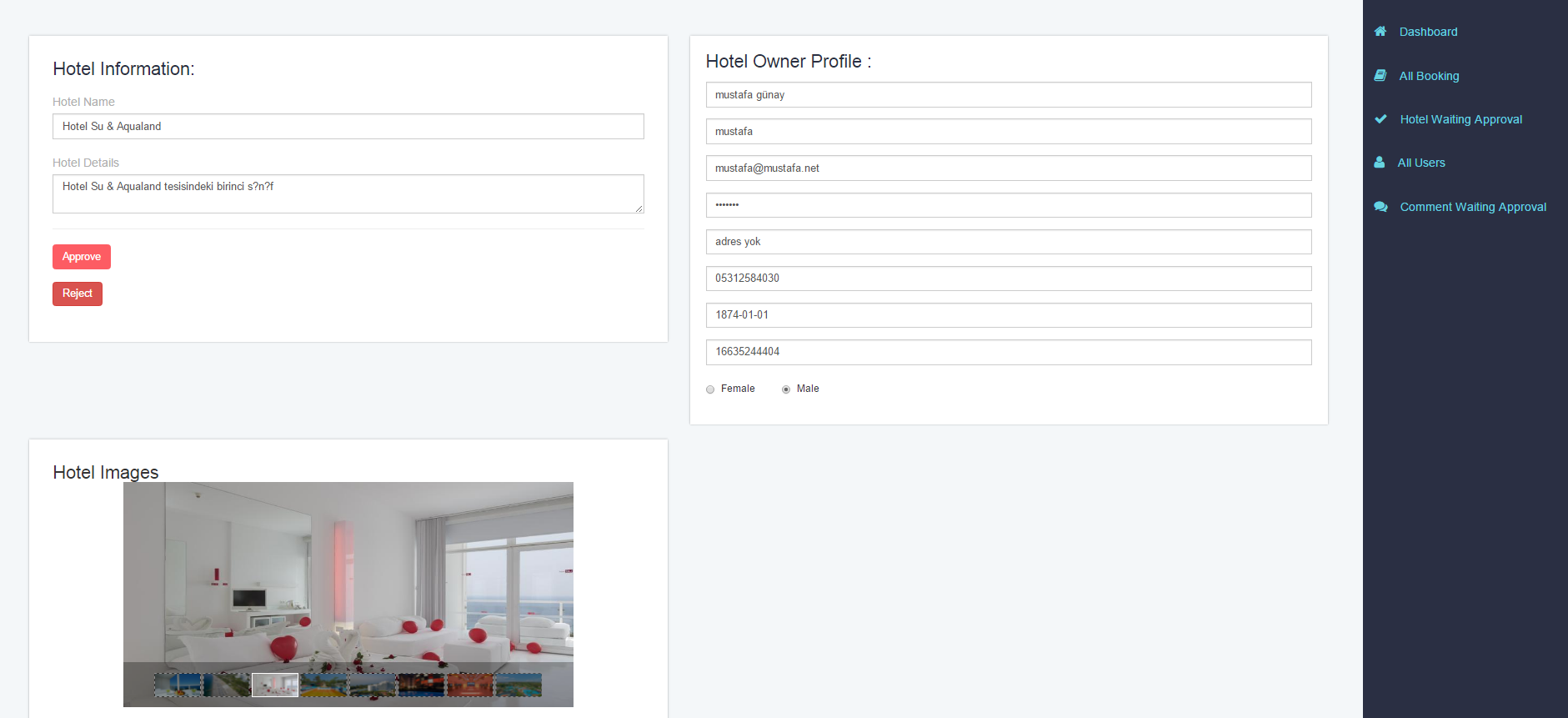
**Comment Page**

****

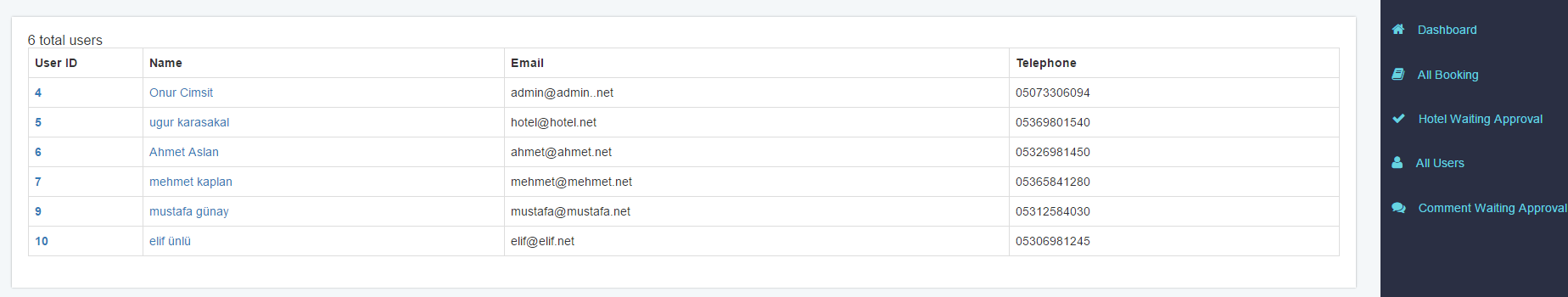
**Admin Page**



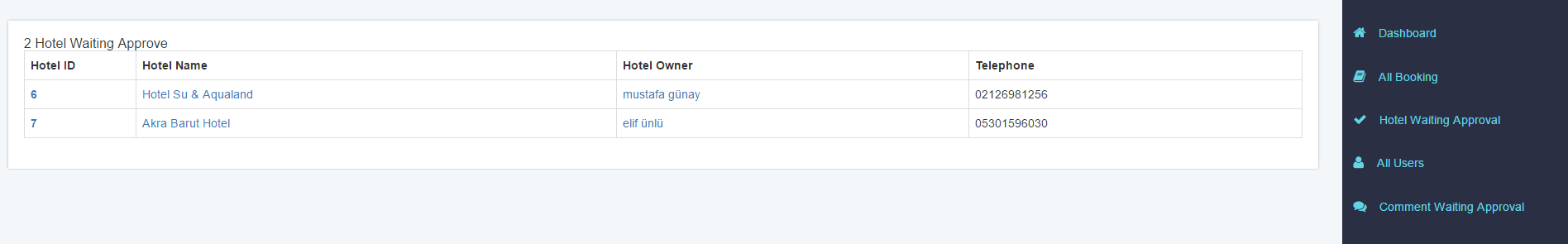
**Hotel Approve-Reject Page**



**User List**



**Hotel Waiting Approval**



**4. GLOSSARY**

**As-Is :** A model that represents the current system as it is without any changes. Here, the current system is Hotel Reservation System.

**Comment** : A comment is a way to inform an user. Because, thank comments gives information about hotels to other users positively/negatively.

**Booking :** An Online Hotel Reservation System. Users can search for any hotel according to their names or locations and user can reserves suitable rooms for them.

**Rating** : Rating is a point scoring system. Each hotel has a rating. Like comments, the user could have information abouthotel or by their ratings. With higher ratings, the hotel is preferred by more users.

**Hotel** : In the system, a hotel is a facility which represents suitable rooms as online.Every hotel has price choices for rooms.

**Log In :** To be recorded to the system with an e-mail and password. To give order, log in is an obligation.

**Sign Up :** To agree to be a participant or recipient of the system by entering personal details.

**System Administrator** : A system admin is a person like user but he/she manages the hotels in the system. He/she should add ordelete the hotels from the system.

**User** : An user is a person who interacts the system directly. He/she uses the specific identity that is different as far as others. He/shecan signs in/up , checks reservations or rate &comment on their previous reservations and makes new reservations.

**Reservation :**  An [arrangement](http://dictionary.cambridge.org/dictionary/english/arrangement) in which something such as a hotel room in a [hotel](http://dictionary.cambridge.org/dictionary/english/aircraft) is [kept](http://dictionary.cambridge.org/dictionary/english/kept) for users.

**Rule** : The system administrator determines some specific principles to fit users’ actions on the system. E.g. Improper comments.

**5.REFERENCES**

**[ 1]** http://www.academia.edu/2112330/A\_SAMPLE\_HOTEL\_MANAGEMENT\_SYSTEM\_PROJECT\_DOCUMENTATION

**[ 2]**

http://webcache.googleusercontent.com/search?q=cache:PQB4QvlrkeUJ:www.site.uottawa.ca/~nelkadri/SEG%25203202/Project-Hotel%2520Reservation%2520Software%2520Product%2520Line.doc+&cd=1&hl=tr&ct=clnk&gl=tr&client=opera