Hotel Reservation

Requirements Specification and Analysis

Version1

10/11/2019

Yaren Çoşkun

Ali Haydar Konuk

Oğuzhan Aydın

Fatih Mehmet İdgü

Prepared for

SE301 Software Engineering



Table of Contents

[1. Introduction 1](#_Toc496873294)

[1.1. Purpose of the System 1](#_Toc496873295)

[1.2. Scope of the System 1](#_Toc496873296)

[1.3. Objectives and Success Criteria of the Project 1](#_Toc496873297)

[1.4. Definitions, Acronyms, and Abbreviations 1](#_Toc496873298)

[1.5. Overview 1](#_Toc496873299)

[2. Current System 1](#_Toc496873300)

[3. Proposed System 1](#_Toc496873301)

[3.1. Overview 1](#_Toc496873302)

[3.2. Functional Requirements 2](#_Toc496873303)

[3.3. Nonfunctional Requirements 2](#_Toc496873304)

[Usability 2](#_Toc496873305)

[Reliability 2](#_Toc496873306)

[Performance 2](#_Toc496873307)

[Supportability 2](#_Toc496873308)

[Implementation 2](#_Toc496873309)

[Interface 2](#_Toc496873310)

[Packaging 2](#_Toc496873311)

[Legal 2](#_Toc496873312)

[3.4. System Models 2](#_Toc496873313)

[Scenarios 2](#_Toc496873314)

[Use case model 2](#_Toc496873315)

[Object model 2](#_Toc496873316)

[Dynamic model 2](#_Toc496873317)

[User interface—navigational paths and screen mock-ups 3](#_Toc496873318)

[3.5. Project Schedule 3](#_Toc496873319)

[4. Glossary 3](#_Toc496873320)

[5. References 3](#_Toc496873321)

# Introduction

The purpose of tatiloradaburada.com is to provide easy and understandable reservation system to the users. It is usable in every level of education. The system has filters and search methods to finding suitable hotel for everyone. Project’s success criteria are making reservation easier to find a hotel for vacation.

## Purpose of the System

The main purpose of the tatiloradaburada.com is, providing an easy, understandable and quality reservation service for the users who are looking for a hotel to do vacation. The system is useful for users, search and reserve a hotel room or cancel his/her reservation over the internet at any time. Users can make comment to indicate their satisfaction. By this way, users can find the best hotel for their vacation to make a reservation.

## Scope of the System

tatiloradaburada.com aims to involve the hotel owners and their hotels that are registered to our system and the users who are searching for any hotel. In general, system has 3 different type of user; hotel owners, registered users and admin. Users can register as registered user or hotel owner without any check easily. But if a hotel owner wants to add a hotel to hotel owner’s hotel list admin has to accept it first. It is not necessary to be a registered to search or filter hotels, but only registered users can make a reservation. Registered user can search a hotel, filter hotels, make comment and change their information. Hotel owner can deactivate his/her hotels and activate, edit hotel and change his/her information.

## Objectives and Success Criteria of the Project

To success tatiloradaburada.com should be chosen from people to make easily reservation instead of other reservation sites.

## Definitions, Acronyms, and Abbreviations

tatiloradaburada.com : Our site to making reservation.

## Overview

We mentioned our current system and how we designed our web site, our functional and nonfunctional requirements.

# Current System

In tatiloradaburada.com, a user can search and filter hotels without registered. To making a reservation user has to register to the tatiloradaburada.com. Users can easily register with filling registered user form which includes name, last name, password, e-mail, phone number, gender, birthday. Registered users can see their past and current reservations their information. Also, they can cancel any current reservation. Finally, registered user can comment on hotels.

Everyone can register as hotel owner by filling hotel owner register form which includes name, last name, password, e-mail, phone number, gender, birthday. After registration hotel owner directs to hotel owner page. He/she can add hotel under my hotels tab. After she/he added his/her hotels, admin gets a request to accept it. After hotel accepted hotel owner can edit his/her hotel and also, she/he can deactivate or activate (after deactivated) it. At least, she/he can change his/her own information.

In addition, admin can accept or deny the hotels which are belonged to the hotel owner. Users can send request for deleting their account to the admin. Eventually, admin can delete comment which are not related with hotels.

# Proposed System

The purpose of the system is, provide users make reservation in easy few steps. Also, users can search and filter hotels, read comments without registration. In the tatiloradaburada.com, login is mandatory to make a reservation. Registration is easy and everyone can register without any condition. The system interface designed so simple to use from every education level. We took tatilbudur.com as an example while designing our site and kept it simpler than it.

## Overview

Presents a functional overview of the system.

The system designed to be user friendly. It aims to brings the hotels which are suitable for user choice easily and let the registered user to make a reservation with just a few clicks. Hotel owners can add own hotels and edit hotels easily. Therefore, the system’s basic and user-friendly interactive interface helps both registered user and hotel owner.

## Functional Requirements

Visitor (non-registered user) can view all hotels, search and sort hotels with their properties and keyword. Also, he/she can filter the hotels with their properties. He/she can sign-up the website.

Registered user can view all hotels, search and sort hotels with their properties and keyword. Also, he/she can filter the hotels with their properties. He/she can make a reservation a room from selected hotel with certain days, number of rooms and by paying its price. He/she can comment about the hotels and delete his/her own comment. Also, he/she can log out.

From his/her own profile, he/she can view and change his/her own account information which contains name and last name, mail address, password. Also, he/she can add his/her own balance amount to use in reservation. He/she can see his/her current reservations and can cancel any reservations. He/she can view his/her past reservations.

Hotel owner can see only his/her own profile. This profile contains account information and his/her hotels. In account information, he/she can view and change his/her own account information which contains name and last name, mail address, password. In his/her hotels, he/she can view his/her own hotels with information and empty rooms. Moreover, he/she can see people who are reserved his/her own hotel room. He/she can delete any hotels from his/her own profile and add new hotel with information which has hotel name, hotel location, price, capacity, rooms’ type and pictures. In addition to this, he/she can edit hotel information which are empty hotel rooms, price and type of hotel rooms. Also, he/she can log out.

## Nonfunctional Requirements

### Usability

tatiloradaburada.com can be used by any user without any prior knowledge.

The user is able to reserve a room from a hotel with just 3 clicks.

User cannot login the system without e-mail and password informations.

### Reliability

When a data loss occurs, the website reports to related user.

Before an account is deleted, account owner can't be registered with same e-mail address for same role.

A hotel is added to website if and only if admin is approval.

### Performance

The website can be only closed for monthly maintaining.

Number of online users does not affect website performance negatively.

### Supportability

tatiloradaburada.com has a role for helping it’s users and this role is Admin. Admin has functions to help RegisteredUsers and HotelOwners. Admin can do some adjustment on tatiloradaburada.com if any non-planned accident occurs. tatiloradaburada.com is in English, most common language in the world and use US Dollar for payments. In the future, Turkish language option and payment with Turkish Lira can be added to tatiloradaburada.com.

### Implementation

TatilOradaBurada.com has been developed with Django which is a Pyhton framework used all over the world by developers and companies. For using python, we used PyCharm. Fort he front-end, we use HTML5, CSS3 and JavaScript. For our database, we use MySql.

### Interface

tatiloradaburada.com can only ran in web browser.

tatiloradaburada.com is not compatible with any mobile application.

Web connection is required.

### Packaging

The system must be available on desktop and laptops and it performs properly on every browser. There is no time constraint for opening the website.

### Legal

The system is free for all users to use it. The system is provided All hotels added to the system has been checked and confirmed.

## System Models

### Scenarios

1)

Scenario Name: DisplayWebsite

Participating Actor Instance(s): Julia:Visitor

Flow of Event:

1. Julia opens the website.
2. She scans hotels from hotel list to take a vacation.
3. She selects a hotel, and clicks to the hotel, then she views information of the hotel.
4. She decides to re-scan later.
5. She closes the website.

2)

Scenario Name: BecomeHotelOwner

Participating Actor Instance(s): Mark:Visitor, Ulio:Admin

Flow of Event:

1. Mark wants to use website to offer online service of his hotel, so he opens the website.
2. He clicks to ‘Register’ button.
3. He selects ‘Hotel Owner Registration’ button, and he views a registration form which includes name, last name, password, e-mail, phone number, gender, birthday. Name, last name, password and e-mail are imperative.
4. He fills up the form and clicks to ‘Register’ button.
5. His account is created, and he is directed to main page of hotel owner.
6. He logs out.

3)

Scenario Name: BecomeRegisteredUser

Participating Actor Instance(s): Vlad:Visitor

Flow of Event:

1. Vlad opens the website.
2. He clicks to ‘Register’ button.
3. He selects ‘User Registration’ button, and he views a registration form which includes name, last name, password, e-mail, phone number, gender, birthday. Name, last name, password and e-mail are imperative.
4. He fills up the form and clicks to ‘Register’ button.
5. His account is created, and he is directed to main page of registered user.
6. He logs out.

4)

Scenario Name: ForgetPassword

Participating Actor Instance(s): Mustafa:RegisteredUser/HotelOwner/Admin

Flow of Event:

1. Mustafa logins to website.
2. He clicks to ‘Login’ button, and a new page is opened that he can login.
3. He forgets his own password, then he clicks to ‘Forget Password’, and a new page is opened to change his password.
4. He enters his e-mail address and new password, and he clicks to ‘Save’ button. Then, he updates his password.
5. He is directed to main page of him.
6. He logs out.

5)

Scenario Name: AddReservation

Participating Actor Instance(s): Helga:Registered User

Flow of Event:

1. Helga logins to website.
2. She finds a hotel for her vacation and clicks to ‘Make Reservation’ button.
3. She enters beginning and ending date of her holiday and chooses room type, then she sees the cost.
4. She clicks to ‘Pay’ button and makes payment of reservation.
5. She is directed to main page of registered user.
6. She logs out.

6)

Scenario Name: CancelReservations

Participating Actor Instance(s): Emma:Registered User

Flow of Event:

1. Emma logins to website.
2. On the main page, Emma sees and clicks her name, and her profile page is opened.
3. Emma clicks to ‘Reservations’ button and the page is opened.
4. On the opened page, she views list of hotels that she booked.
5. She clicks to ‘Cancel’ button which is next to the reservation information.
6. She sees the notification “Are you sure you want to cancel your reservation?” and clicks to ‘Yes’ button.
7. She is directed to reservations page of registered user.
8. She logs out.

7)

Scenario Name: ShowCurrentReservations

Participating Actor Instance(s): Işıl:Registered User

Flow of Event:

1. Işıl logins to website.
2. On the main page, Işıl sees and clicks her name, and her profile page is opened.
3. She clicks to ‘Reservations’ button and the page is opened.
4. On the opened page, she sees list of currently reservations.
5. She logs out.

8)

Scenario Name: ShowPastReservations

Participating Actor Instance(s): Cansu:Registered User

Flow of Event:

1. Cansu logins to website.
2. On the main page, Cansu sees and clicks her name, and her profile page is opened.
3. She clicks to ‘Past Reservations’ button and the page is opened.
4. On the opened page, she sees list of reservations she has previously booked.
5. She logs out.

9)

Scenario Name: MakeComment

Participating Actor Instance(s): Hilmi:Registered User

Flow of Event:

1. Hilmi logins to website.
2. On the main page, Hilmi saw a Hotel which is he went before.
3. Hilmi clicks to Hotel’s name and the page is opened.
4. On the opened page, he sees a text field to make a comment.
5. He writes his comment and clicks “Save” button.
6. He logs out.

10)

Scenario Name: SearchHotels

Participating Actor Instance(s): Mary:Visitor/RegisteredUser

Flow of Event:

1. Mary opens the website.
2. She writes a ‘Rixos’ on search bar.
3. She views hotel list which names includes Rixos.
4. She checks hotel.
5. She logs out.

11)

Scenario Name: FilterHotels

Participating Actor Instance(s): Nicki:Visitor/RegisteredUser

Flow of Event:

1. Nicki opens the website.
2. She selects price between 1000$ and 1500$ and select location as Marmaris on filter section.
3. She views hotel list which required qualifications.
4. She checks hotel.
5. She logs out.

12)

Scenario Name: DeleteAccountRequest

Participating Actor Instance(s): Cardi:HotelOwner/RegisteredUser, Offset:Admin

Flow of Event:

1. Cardi logins the website.
2. She clicks to ‘Account Information’ tab and clicks to ‘Delete My Account’ button. Then, this request sends to Offset.
3. She logs out.

13)

Scenario Name: ChangeRegisteredUserInformation

Participating Actor Instance(s): Ahmet:Registered User

Flow of Event:

1. Ahmet logins to website.
2. On the main page, Ahmet sees and clicks his name, and his profile page is opened.
3. He clicks to ‘Account Information’ tab and he clicks to ‘Edit’ button, so editing page is opened.
4. On the opened page, he selects the field which he wants to change and enters new information.
5. He clicks to ‘Save’ button, and he is directed to information page of the register user.
6. He logs out.

14)

Scenario Name: AddHotelByHotelOwner

Participating Actor Instance(s): Rıdvan:Hotel Owner, Erol:Admin

Flow of Event:

1. Rıdvan logins to website.
2. Rıdvan clicks to ‘My Hotels’ tab from main page of hotel owner and a new page is opened which has list of all hotel he has.
3. He clicks to ‘Add’ button which is at end of the list of all hotels.
4. He views a blank form and he fills up the form with information of new hotel. These information are hotel name, location, quality, room type and price. All fields are required.
5. He clicks to ‘Save’ button and he is directed to the page that list of all hotels.
6. Request is sent to Erol, and Erol decides to accept and does it.
7. He logs out.

15)

Scenario Name: EditHotelByHotelOwner

Participating Actor Instance(s): Aykut:Hotel Owner

Flow of Event:

1. Aykut logins to website.
2. Aykut clicks to ‘My Hotels’ tab from main page of hotel owner and a new page is opened which has list of all hotel he has.
3. He clicks to ‘Edit’ button that is near the hotel which he wants to edit, and he views information of the hotel.
4. He selects an information, and he rewrite new information for hotel.
5. He clicks to ‘Save’ button and he is directed to the page that list of all hotels.
6. He logs out.

16)

Scenario Name: DeactivateHotelByHotelOwner

Participating Actor Instance(s): Oğuz:Hotel Owner

Flow of Event:

1. Oğuz logins to website.
2. Oğuz clicks to ‘My Hotels’ tab from main page of hotel owner and a new page is opened which has list of all hotel he has.
3. He clicks to ‘Deactivate’ button that is near the hotel which he wants to deactivate.
4. He sees a pop-up which contains a message that ‘Are you sure you want to deactivate?’.
5. He clicks to ‘Yes’ button, and he deactivates hotel and pop-up is closed then he is directed to main page of hotel owner.
6. He logs out.

17)

Scenario Name: ActivateHotelByHotelOwner

Participating Actor Instance(s): Elvir:Hotel Owner

Flow of Event:

1. Elvir logins to website.
2. Elvir clicks to ‘My Hotels’ tab from main page of hotel owner and a new page is opened which has list of all hotel he has.
3. He clicks to ‘Activate’ button that is near the hotel which he wants to activate.
4. He sees a pop-up which contains a message that ‘Are you sure you want to activate?’.
5. He clicks to ‘Yes’ button, and he activates hotel and pop-up is closed then he is directed to main page of hotel owner.
6. He logs out.

18)

Scenario Name: ChangeHotelOwnerInformation

Participating Actor Instance(s): Derin:Hotel Owner

Flow of Event:

1. Derin logins to website.
2. On the main page, Derin sees and clicks his name, and her profile page is opened.
3. She clicks to ‘Account Information’ tab and clicks to ‘Edit’ button, so editing page is opened.
4. On the opened page, she selects the field which she wants to change and enters new information.
5. She clicks to ‘Save’ button, and she is directed to information page of hotel owner.
6. She logs out.

19)

Scenario Name: AcceptHotel

Participating Actor Instance(s): Anıl:HotelOwner, Fatih:Admin

Flow of Event:

1. Anıl sends a request to add a new hotel.
2. When Fatih checks hotel owners, he sees that Anıl sent a request to Fatih about accepting Anıl’s hotel.
3. Fatih knows Anıl bought a new hotel and wants to add it to website. Fatih clicks ‘Accept’ button to add Anıl’s hotel.
4. Hotel is added to Anıl’s hotels.
5. Anıl is received an e-mail that explains about this action.

20)

Scenario Name: DeleteHotelOwner

Participating Actor Instance(s): Anıl:HotelOwner, Fatih:Admin

Flow of Event:

1. Anıl wants to delete his account on website. He send request to Fatih.
2. Fatih decides to delete Anıl’s account and opens ‘Hotel Owners’ tab and finds Anıl’s account.
3. Fatih clicks to ‘Delete’ button in front of HotelOwner.
4. Anıl’s account and all hotels are which is belong to Anıl are updated as deleted.
5. Anıl is received an e-mail that explains about this action.

21)

Scenario Name: DeleteComment

Participating Actor Instance(s): Anıl:HotelOwner, Fatih:Admin

Flow of Event:

1. When Anıl views his hotel’s comments, he sees a comment which is not about his hotel.
2. He reports comment to Fatih and he explains it to Fatih.
3. Fatih opens ‘Comments’ tab and finds the comment which Anıl reports.
4. Fatih reads comment and notices that comment is irrelevant with hotel.
5. Fatih selects the comment with checkbox, selects the action ‘Update’ and clicks ‘Go’ button.
6. Comment is updated as deleted from the “Comments” tab.
7. Anıl is received an e-mail that explains about this action.

### 

### Use case model

1.

|  |  |
| --- | --- |
| *Use case name* | Login |
| *Participating actor* | Initiated by RegisteredUser/HotelOwner/Admin  Intereacts with tatiloradaburada.com |
| *Flow of events* | 1. RegisteredUser opens the tatiloradaburada.com  2. tatiloradaburada.com responses by presenting homepage  3. RegisteredUser clicks on login button.  4. tatiloradaburada.com presents login page which has email and password fields.  5.RegisteredUser enter his email and password to the empty text fields.  6. tatiloradaburada.com takes fields and checks if the fields are matches with a user information and returnes matched user. |
| *Entry condition* | * RegisteredUser has registered before. |
| *Exit condition* | * RegisteredUser successfully logins to the tatiloradaburada.com. |

*Exceptional Cases*

*[Invalid Email] : If email the RegisteredUser entered is wrong or not registered before.The system gives an warning message and returns to login screen again.*

*[Invalid Password] : If password the RegisteredUser entered is wrong.The system gives an warning and return login screen.*

2.

|  |  |
| --- | --- |
| *Use case name* | Logout |
| *Participating actor* | Initiated by RegisteredUser/HotelOwner/Admin  Intereacts with tatiloradaburada.com |
| *Flow of events* | 1.RegisteredUser opens the tatiloradaburada.com with his/her account information.  1.1 HotelOwner opens the tatiloradaburada.com with his/her account information.  1.2. Admin opens the tatiloradaburada.com with his/her account information.  2. tatiloradaburada.com responses by presenting homepage to the RegisteredUser  2.1 tatiloradaburada.com responses by presenting HotelOwner’s Main Page to the HotelOwner  2.2 tatiloradaburada.com responses by presenting admin panel to the Admin  3. RegisteredUser clicks on logout button.  3.1. HotelOwner clicks on logout button.  3.2. Admin clicks on logout button.  4. tatiloradaburada.com presents home page. |
| *Entry condition* | * RegisteredUser/HotelOwner/Admin has logined before. |
| *Exit condition* | * RegisteredUser/HotelOwner/Admin successfully logout from the tatiloradaburada.com. |

3.

|  |  |
| --- | --- |
| *Use case name* | DisplayWebsite |
| *Participating actor* | Initiated by Visitor |
| *Flow of events* | 1. Visitor opens the tatiloradaburada.com  2. tatiloradaburada.com responses by presenting homepage which includes hotels lists.  3. Visitor clicks on a hotel name.  4. tatiloradaburada.com opens hotel page which has otel information(Hotel name, hotel location, quality and price).  5.After visitor look at hotel information then visitor closes the tatiloradaburada.com. |
| *Entry condition* | * Visitor knows how to use tatiloradaburada.com |
| *Exit condition* | * Visitor gets information about hotels. |

4.

|  |  |
| --- | --- |
| *Use case name* | Register |
| *Participating actor* | Initiated by Visitor  interacts with tatiloradaburada.com |
| *Flow of events* | 1. Visitor opens the tatiloradaburada.com  2. tatiloradaburada.com responses by presenting homepage  3. Visitor clicks on ‘Register’ button.  4. tatiloradaburada.com presents register page which has ‘Hotel Owner Registration’ button and ‘User Registration’ button.  5. Visitor clicks on ‘Hotel Owner Registration’ button.  5.1 Visitor clicks on ‘User Registration’ button.  6. tatiloradaburada.com opens register page thats is specified for Hotel Owners and views a registration form which includes name, last name, password, e-mail, phone number, gender, birthday.  7. Visitor enters at least all of the required informations and clicks to ‘Register’ button  8. tatiloradaburada.com takes fields, creates hotel owner’s account, and directs hotel owner to main page of hotel owner.  8.1 tatiloradaburada.com takes fields and checks if the fields are all acceptable and if they are; create the Registered User account for this user. He/She is directed to his/her main page as Registered User. |
| *Entry condition* | * Visitor opens the tatiloradaburada.com |
| *Exit condition* | * Visitor successfully registers to the tatiloradaburada.com. |

*Exceptional Cases*

*[registered Email] : If email the Hotel owner entered is registered before ,The system gives an warning message and returns to hotel owner registration page again.*

*[null field] : If* Name, last name, password and e-mail *is null. The system gives an warning and return to hotel owner registration page again..*

*[Invalid Email] : If email the RegisteredUser entered is not valid or has registered before.The system gives an warning message and do not complete registration.*

*[Invalid Password] : If password the Visitor entered is not in wanted form.The system gives an warning and do not complete registration.*

*[Invalid Input]: If the input is not at the wanted type, the system gives an warning and do not complete registration.*

5.

|  |  |
| --- | --- |
| *Use case name* | ForgetPassword |
| *Participating actor* | Initiated by RegisteredUser / HotelOwner / Admin  interacts with tatiloradaburada.com |
| *Flow of events* | 1. RegisteredUser/HotelOwner/Admin opens the tatiloradaburada.com  2. tatiloradaburada.com responses by presenting homepage  3. RegisteredUser/HotelOwner/Admin clicks on ‘Login’ button.  4. tatiloradaburada.com presents login page.  5. RegisteredUser/HotelOwner/Admin clicks on ‘Forget Password’ button.  6. tatiloradaburada.com opens new page to change password.  7. RegisteredUser/HotelOwner/Admin enters his/her own e-mail address and new password, and he/she clicks to ‘Save’ button.  8. tatiloradaburada.com directs him/her to main page. |
| *Entry condition* | * RegisteredUser/HotelOwner/Admin registered tatiloradaburada.com before. |
| *Exit condition* | * RegisteredUser/HotelOwner/Admin successfully changes his/her own new password. |

*Exceptional Cases*

*[null field] : If e-mail or password is null. The system gives a warning and keeps to stay in same page.*

*[Invalid Email] : If email is not valid or has not registered before. The system gives an warning message and keeps to stay in same page.*

6.

|  |  |
| --- | --- |
| *Use case name* | SearchHotels |
| *Participating actor* | Initiated by Visitor/RegisteredUser  interacts with tatiloradaburada.com |
| *Flow of events* | 1. Visitor/RegisteredUser opens the tatiloradaburada.com  2. tatiloradaburada.com responses by presenting homepage which includes hotels lists.  3. Visitor/RegisteredUser writes a hotel name and selects dates on search bar.  4. tatiloradaburada.com opens a hotel list which matches with searched features. |
| *Entry condition* | * Visitor/RegisteredUsed knows how to use tatiloradaburada.com |
| *Exit condition* | * Visitor/RegisteredUser gets information about searched hotels. |

7.

|  |  |
| --- | --- |
| *Use case name* | FilterHotels |
| *Participating actor* | Initiated by Visitor/RegisteredUser  interacts with tatiloradaburada.com |
| *Flow of events* | 1. Visitor/RegisteredUser opens the tatiloradaburada.com  2. tatiloradaburada.com responses by presenting homepage which includes hotels lists.  3. Visitor/RegisteredUser selects some hotel features from filter section. Then, Visitor/RegisteredUser clicks to ‘Apply’ button.  4. tatiloradaburada.com opens a hotel list which matches with searched features. |
| *Entry condition* | * Visitor/RegisteredUsed knows how to use tatiloradaburada.com |
| *Exit condition* | * Visitor/RegisteredUser gets information about searched hotels. |

8.

|  |  |
| --- | --- |
| *Use case name* | AddReservation |
| *Participating actor* | Initiated by Registered User  interacts with tatiloradaburada.com |
| *Flow of events* | 1. Registered User views hotels on the homepage.  2. Registered user selects a hotel for vacation and clicks on ‘Make Reservation’ button.  4. tatiloradaburada.com opens reservation page.  5. Registered user enters beginning and ending date of reservation and chooses room type, then views the cost.  6. Registered User clicks to ‘Pay’ button and makes payment of reservation.  7. tatiloradaburada.com gets entered information, create reservation and direct Registered user to homepage. |
| *Entry condition* | * RegisteredUser logins to tatiloradaburada.com |
| *Exit condition* | * RegisteredUser successfully make a reservation. |

*Exceptional Cases*

*[payment fails] : If payment is not successful ,The system gives a warning message and returns to reservation page again.*

*[null field] : If beginning date, ending date, room type is null. The system gives a warning and returns to reservation page again.*

9.

|  |  |
| --- | --- |
| *Use Case Name* | ShowCurrentReservation |
| *Participating Actors* | Initiated by RegisteredUser |
| *Flow of Events* | 1.RegisteredUser clicks to ‘Reservations’ tab from main page of registered user.  2.tatiloradaburada.com opens a new page which has list of planned reservations.  3.RegisteredUser view his/her planned reservation list.  4.RegisteredUser clicks one of the reservations.  5.tatiloradaburada.com opens a new page which has informations of clicked reservation.  6.RegisteredUser views details of clicked reservation.  7.RegisteredUser click to ‘X’ button and exit from ‘My Reservations’ page.  8.tatiloradaburada.com directs RegisteredUser to registered users main page.  9. RegisteredUser logs out. |
| *Entry Condition* | * RegisteredUser logins to tatiloradaburada.com. * There is at least one planned reservation. |
| *Exit Condition* | * RegisteredUser viewed all the information on clicked reservation. |

10.

|  |  |
| --- | --- |
| *Use Case Name* | ShowPastReservation |
| *Participating Actors* | Initiated by RegisteredUser |
| *Flow of Events* | 1.RegisteredUser clicks to ‘Past Reservations’ tab from main page of registered user.  2.tatiloradaburada.com opens a new page which has list of past reservations.  3.RegisteredUser view his/her past reservation list.  4.RegisteredUser clicks one of the reservations.  5.tatiloradaburada.com opens a new page which has informations of clicked reservation.  6.RegisteredUser views details of clicked reservation.  7.RegisteredUser click to ‘X’ button and exit from ‘Past Reservations’ page.  8.tatiloradaburada.com directs RegisteredUser to registered users main page.,  9. RegisteredUser logs out. |
| *Entry Condition* | * RegisteredUser logins to tatiloradaburada.com. * There is at least one past reservation. |
| *Exit Condition* | * RegisteredUser viewed all the information on clicked reservation. |

11.

|  |  |
| --- | --- |
| *Use Case Name* | CancelReservation |
| *Participating Actors* | Initiated by RegisteredUser |
| *Flow of Events* | 1.RegisteredUser clicks to ‘Reservations’ tab from main page of registered user.  2.tatiloradaburada.com opens a new page which has list of past reservations.  3.RegisteredUser view his/her past reservation list.  4.RegisteredUser clicks ‘Cancel’ button next to the reservation which will be cancelled.  5.tatiloradaburada.com opens a warning pop-up which says ‘Are you sure you want to cancel this reservation?’ with two buttons ‘Yes’ for continue the operation and ‘No’ for cancelling the operation.  6.1.RegisteredUser clicks to ‘Yes’ button.  6.2.RegisteredUser clicks to ‘No’ button.  7.1.tatiloradaburada.com deletes the chosen reservation and shows a pop-up which says ‘Operation is Succesful’.  7.2.tatiloradaburada.com cancels the operation.  8.tatiloradaburada.com sends RegisteredUser to register user main page.  9. RegisteredUser logs out. |
| *Entry Condition* | * RegisteredUser logins to tatiloradaburada.com. * There is at least one planned reservation. |
| *Exit Condition* | * Reservation is deleted on all pages including HotelOwners. |

12.

|  |  |
| --- | --- |
| *Use case name* | ChangeInformation |
| *Participating actor* | Initiated by RegisteredUser/HotelOwner  Intereacts with tatiloradaburada.com |
| *Flow of events* | 1.RegisteredUser clicks her/his name.  1.1 HotelOwner clicks her/his name.  2. tatiloradaburada.com responses by presenting RegisterUser profile to the RegisteredUser  2.1 tatiloradaburada.com responses by presenting HotelOwner’s profile to the HotelOwner  3. RegisteredUser clicks on Account Information.  3.1. HotelOwner clicks on Account Information.  4. tatiloradaburada.com responses by presenting AccountInformation profile to the RegisteredUser  4.1. tatiloradaburada.com responses by presenting AccountInformation profile to the HotelOwner.  5. RegisteredUser clicks Edit button.  5.1. HotelOwner clicks Edit button.  6. tatiloradaburada.com brings RegisteredUsers information with editible form, form includes name, last name, password, e-mail, phone number, gender, birthday.  6.1. tatiloradaburada.com brings HotelOwner information with editible form, form includes name, last name, password, e-mail, phone number, gender, birthday.  7.RegisteredUser changes whatever he/she want on the form and clicks save button.  7.1. HotelOwner changes whatever he/she want on the form and clicks save button.  8. tatiloradaburada.com takes updated form and updates informations. |
| *Entry condition* | * RegisteredUser/HotelOwner/Admin has logined before. |
| *Exit condition* | * RegisteredUser/HotelOwner/Admin successfully changed her/his account information. |

*Exceptional Cases*

*[Invalid Email] : If email the RegisteredUser/HotelOwner entered is wrong or registered before.The system gives an warning message and returns to EditingAcount screen again.*

*[Invalid Password] : If password the RegisteredUser/Owner entered is wrong.The system gives an warning and return login screen.*

13.

|  |  |
| --- | --- |
| *Use case name* | AddHotelByHotelOwner |
| *Participating actor* | Initiated by HotelOwner  communicates with Admin |
| *Flow of events* | 1. Hotel owner clicks to ‘My Hotels’ tab from main page of hotel owner.  2. tatiloradaburada.com opens a new page which has list of all hotel that hotel owner has.  3. Hotel owner clicks to ‘Add’ button which is at end of the list of all hotels.  4. tatiloradaburada.com views a blank form requesting hotel information. These information are hotel name, location, quality, room type, vacant room and price.  5. Hotel owner fills up the form with information of new hotel.  6. Hotel Owner clicks to ‘Save’ button.  7. tatiloradaburada.com directs hotel owner to the page that has list of all hotels and sends new hotel information as request to admin.  8.Admin accepts the request. |
| *Entry condition* | * Hotelowner logins to tatiloradaburada.com |
| *Exit condition* | * Hotelowner successfully sends request to admin for adding hotel. |

*Exceptional Cases*

*[null field] : If hotel name, location, quality, room type and price is null. The system gives a warning and returns to reservation page again.*

14.

*Use Case Name* EditHotelByHotelOwner

*Participating Actors* Initiated by HotelOwner

*Flow of Events*

1. HotelOwner clicks to ‘My Hotels’ tab from main page of hotel owner.
   * 1. tatiloradaburada.com opens a new page which has list of all hotel HotelOwner has.
2. HotelOwner clicks to ‘Edit’ button that is near the hotel which HotelOwner wants to edit.

tatiloradaburada.com opens information of the hotel.

1. HotelOwner selects an information, and HotelOwner rewrites new information for hotel.
2. HotelOwner clicks to ‘Save’ button.

tatiloradaburada.com stores new information and directs HotelOwner to the page that list of all hotels.

*Entry Condition* HotelOwner logins to tatiloradaburada.com

*Exit Condition* HotelOwner edits information of the hotel which HotelOwner wants

to edit.

15.

*Use Case Name* DeactivateHotelByHotelOwner

*Participating Actors* Initiated by HotelOwner

*Flow of Events*

1. HotelOwner clicks to ‘My Hotels’ tab from main page of hotel owner.
2. tatiloradaburada.com opens a new page which has list of all hotel HotelOwner has.
3. HotelOwner clicks to ‘Deactivate’ button that is near the hotel which HotelOwner wants to deactivate.
4. tatiloradaburada.com opens a pop-up which contains a message that ‘Are you sure you want to deactivate?’.

5.1 HotelOwner clicks to ‘Yes’ button.

1. HotelOwner clicks to ‘No’ button.
2. tatiloradaburada.com closes pop-up, then tatiloradaburada.com directs HotelOwner to main page of hotel owner.
3. tatiloradaburada.com opens a new page which has list of all hotel HotelOwner has.

*Entry Condition* HotelOwner logins to tatiloradaburada.com

*Exit Condition* HotelOwner deactivates his/her own hotel.

16.

*Use Case Name* ActivateHotelByHotelOwner

*Participating Actors* Initiated by HotelOwner

*Flow of Events*

1. HotelOwner clicks to ‘My Hotels’ tab from main page of hotel owner.
2. tatiloradaburada.com opens a new page which has list of all hotel HotelOwner has.
3. HotelOwner clicks to ‘Activate’ button that is near the hotel which HotelOwner wants to activate.
4. tatiloradaburada.com opens a pop-up which contains a message that ‘Are you sure you want to activate?’.
5. HotelOwner clicks to ‘Yes’ button.
6. HotelOwner clicks to ‘No’ button.
7. tatiloradaburada.com closes pop-up, then tatiloradaburada.com directs HotelOwner to main page of hotel owner.
8. tatiloradaburada.com opens a new page which has list of all hotel HotelOwner has.

*Entry Condition* HotelOwner logins to tatiloradaburada.com

*Exit Condition* HotelOwner activates his/her own hotel.

17.

|  |  |
| --- | --- |
| *Use case name* | AcceptHotel |
| *Participating actor* | Initiated by Admin  communicates with Hotelowner |
| *Flow of events* | 1. Hotel owner sends a request to add a new hotel.  2. tatiloradaburada.com sends new hotel request to admin.  3. Admin checks hotel owners and sees that hotel owner’s request about accepting new hotel under the ‘Accept Hotel’ tab.  4. Admin clicks ‘Accept’ button to add hotel owner’s hotel.  5. tatiloradaburada.com adds new hotel to hotel owner’s hotels and sent an e-mail to hotel owner about his/her hotel is added. |
| *Entry condition* | * Admin logins to tatiloradaburada.com * Hotel owner sends a request for adding a new hotel. |
| *Exit condition* | * Admin accept or decline request from hotel owner for adding hotel. |

18.

|  |  |
| --- | --- |
| *Use case name* | DeleteComment |
| *Participating actor* | Initiated by Admin  Interacts with tatiloradaburada.com |
| *Flow of events* | 1. Admin opens ‘Delete Comment’ tab and finds the comment which he/she wants to delete.  2. Admin clicks to ‘Delete’ button in front of the comment.  3. tatiloradaburada.com updates the comment as deleted from the “Delete Comment” tab and sent an e-mail to hotel owner about one of the comments written to his/her hotel was deleted. |
| *Entry condition* | * Admin logins to tatiloradaburada.com |
| *Exit condition* | * Admin successfully deletes the comment. |

19.

|  |  |
| --- | --- |
| *Use case name* | AcceptHotel |
| *Participating actor* | Initiated by Admin  communicates with Hotelowner |
| *Flow of events* | 1. Hotel owner sends a request to add a new hotel.  2. tatiloradaburada.com sends new hotel request to admin.  3. Admin checks hotel owners and sees that hotel owner’s request about accepting new hotel under the ‘Accept Hotel’ tab.  4. Admin clicks ‘Accept’ button to add hotel owner’s hotel.  5. tatiloradaburada.com adds new hotel to hotel owner’s hotels and sent an e-mail to hotel owner about his/her hotel is added. |
| *Entry condition* | * Admin logins to tatiloradaburada.com * Hotel owner sends a request for adding a new hotel. |
| *Exit condition* | * Admin accept or decline request from hotel owner for adding hotel. |

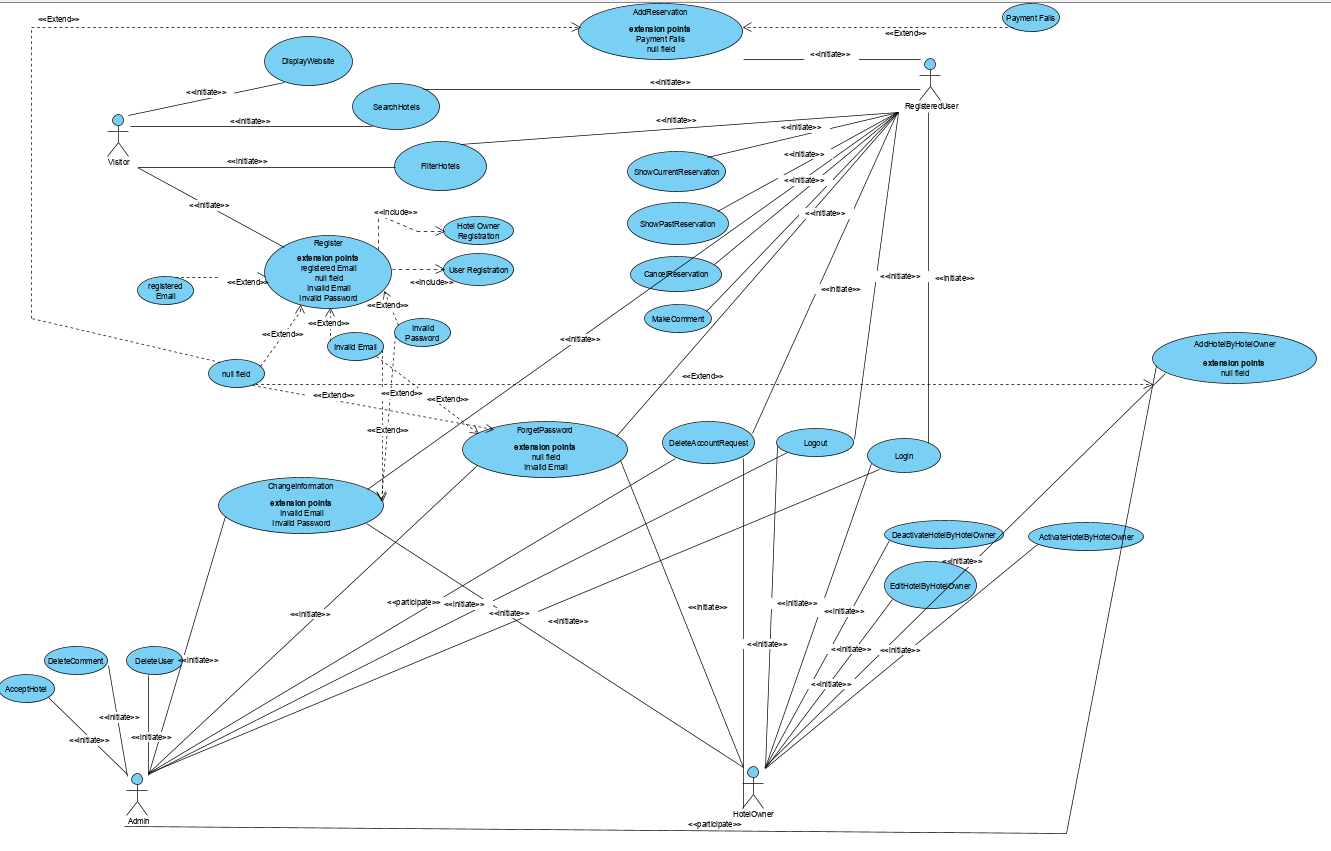
20.

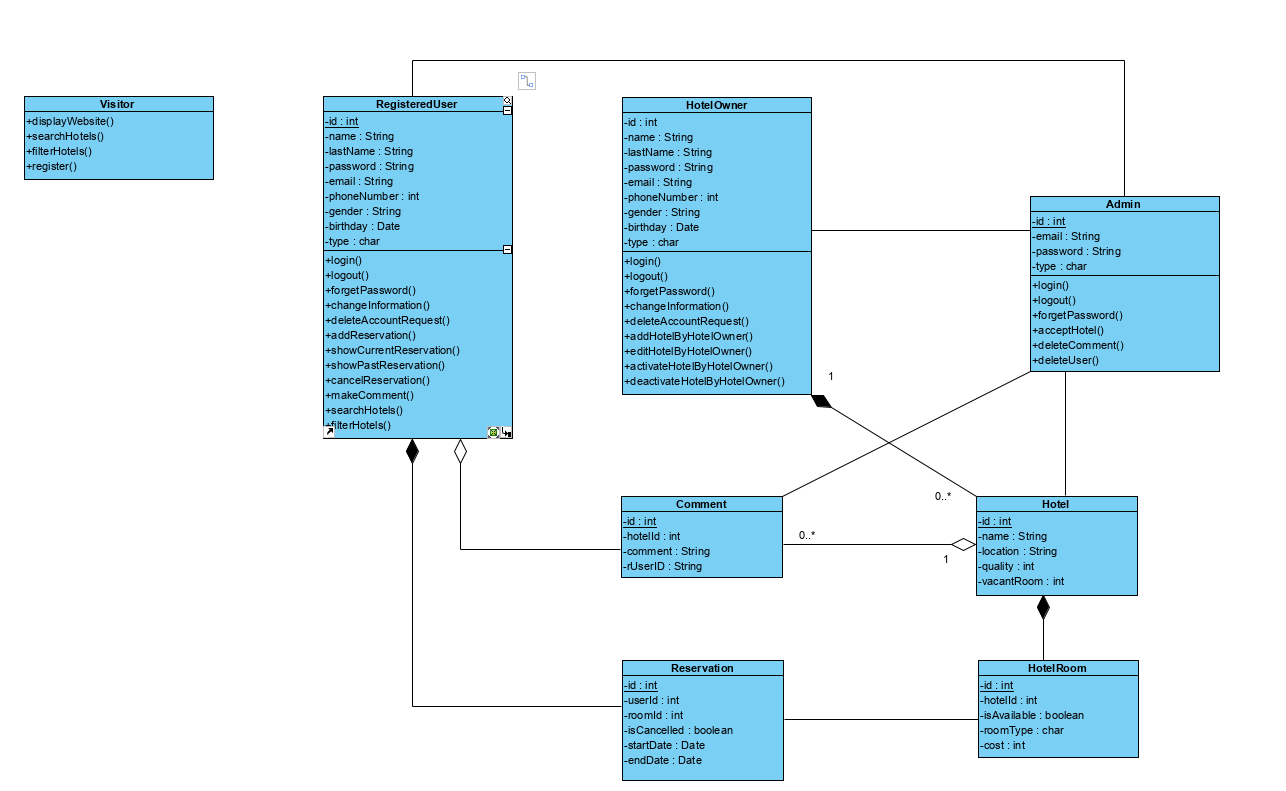
|  |  |
| --- | --- |
| *Use case name* | MakeComment |
| *Participating actor* | Initiated by RegisteredUser  Intereacts with tatiloradaburada.com |
| *Flow of events* | 1. RegisteredUser opens the tatiloradaburada.com  2. tatiloradaburada.com responses by presenting homepage  3. RegisteredUser clicks on Hotel’s name.  4. tatiloradaburada.com presents Hotel’s page.  5.RegisteredUser enters his comment and clicks “Save” button.  6. tatiloradaburada.com takes comment and show it on under the Hotel information. |
| *Entry condition* | * RegisteredUser has logined before. |
| *Exit condition* | * RegisteredUser makes comment successfully. |

21.

|  |  |
| --- | --- |
| *Use case name* | DeleteAccountRequest |
| *Participating actor* | Initiated by HotelOwner/RegisteredUser  interacts with Admin |
| *Flow of events* | 1. HotelOwner/RegisteredUser clicks to ‘Account Information’ tab.  2. tatiloradaburada.com responses by presenting Account Information.  3. HotelOwner/RegisteredUser clicks to ‘Delete My Account’ button.  4. tatiloradaburada.com sends request to Admin. |
| *Entry condition* | * HotelOwner/RegisteredUsed has logined tatiloradaburada.com before |
| *Exit condition* | * Admin receives a request about deleting an account. |

### Object model



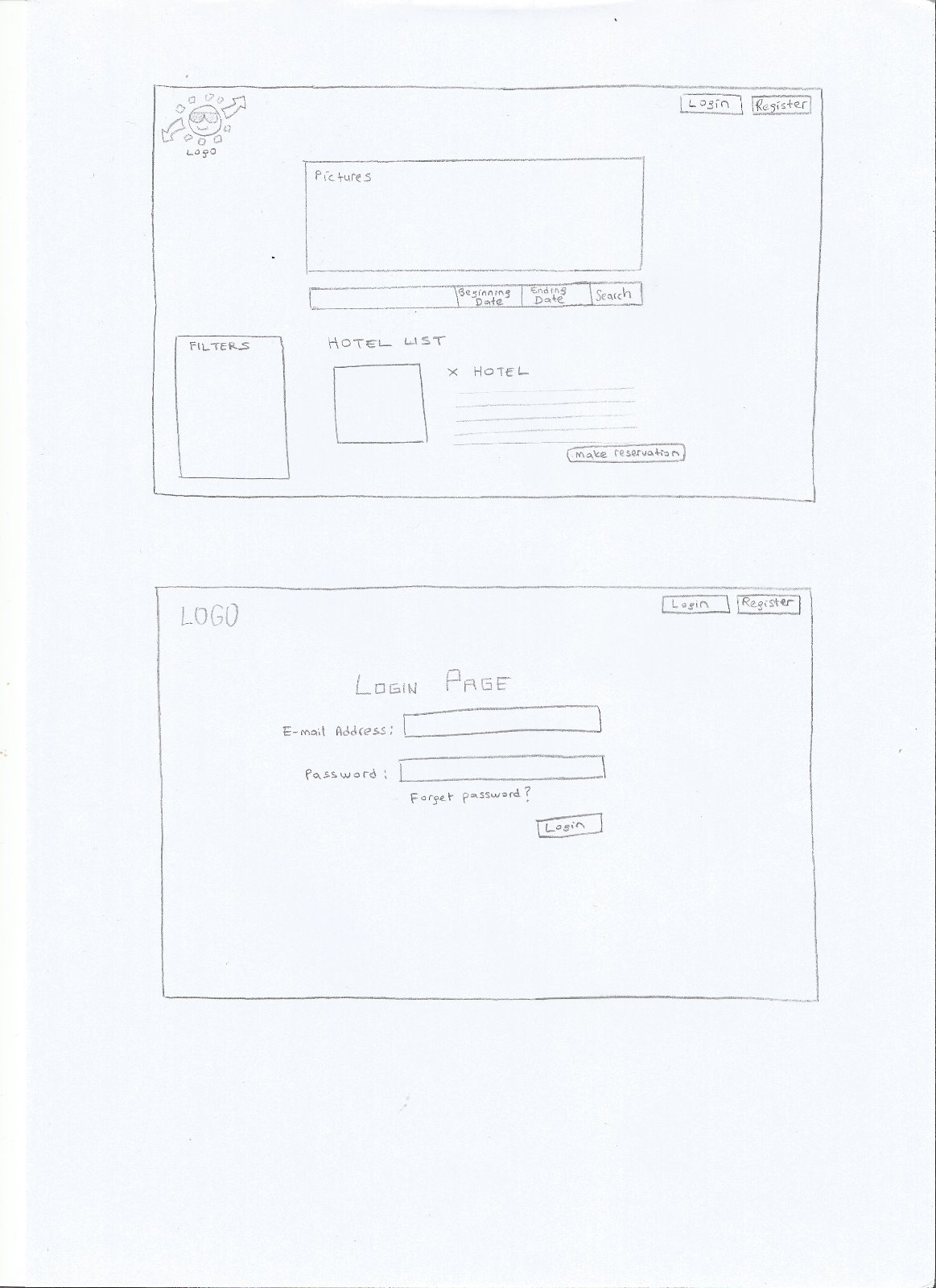


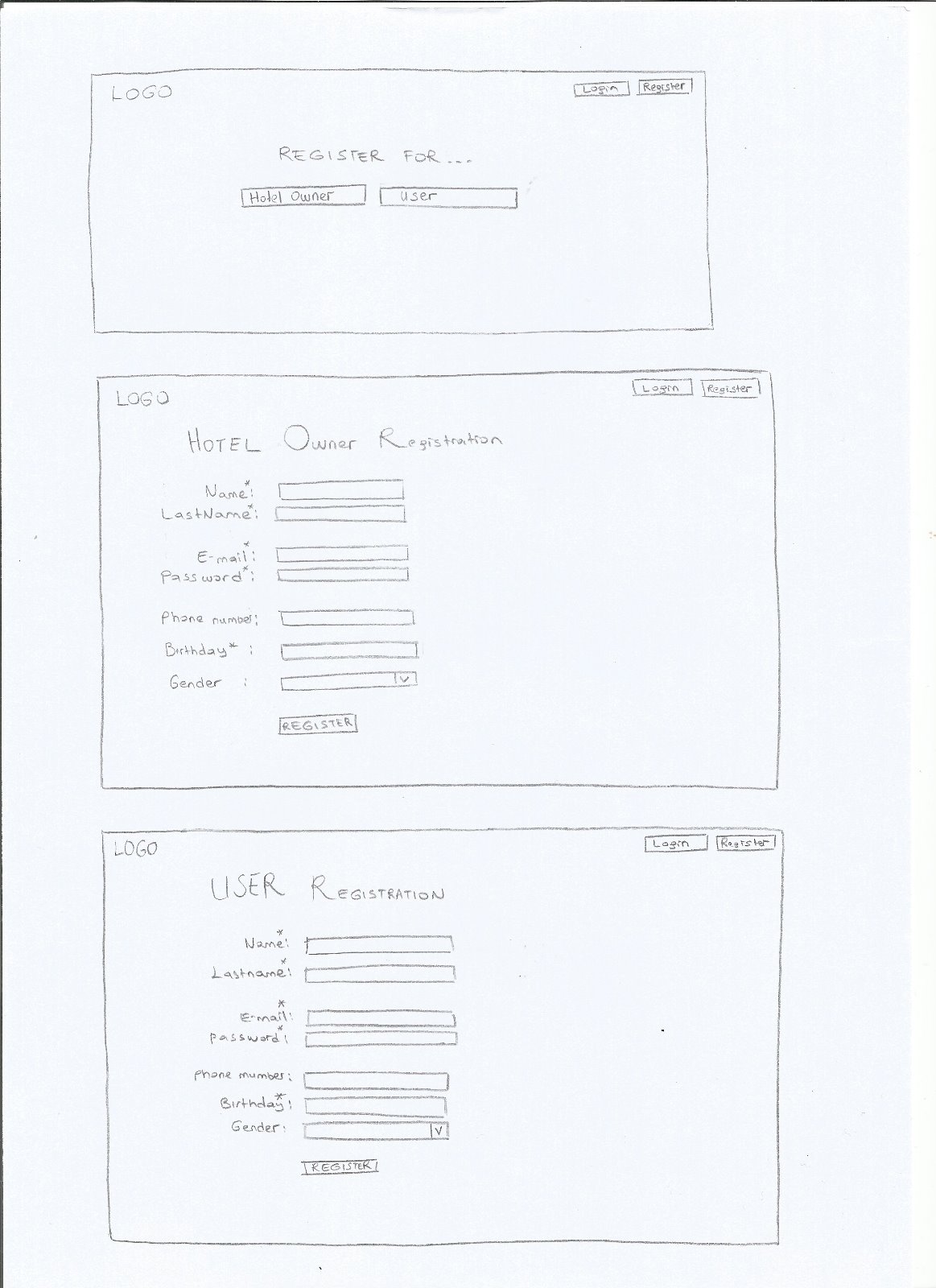
### Dynamic model

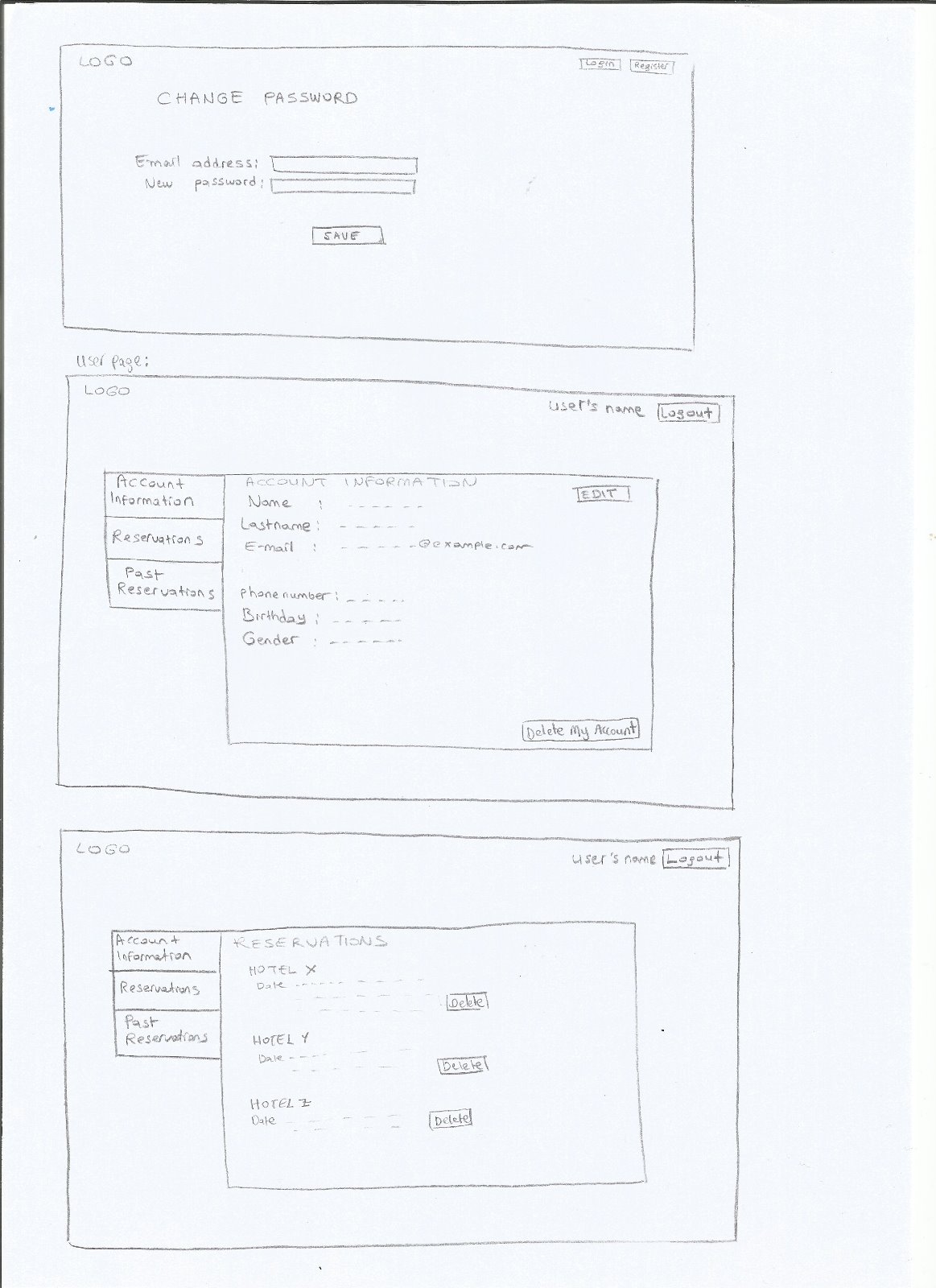
The dynamic model is depicted with sequence diagrams and with state machines. Sequence diagrams represent the interactions among a set of objects during a single use case. State machines represent the behavior of a single object (or a group of very tightly coupled objects). The dynamic model serves to assign responsibilities to individual classes and, in the process, to identify new classes, associations, and attributes to be added to the analysis object model.

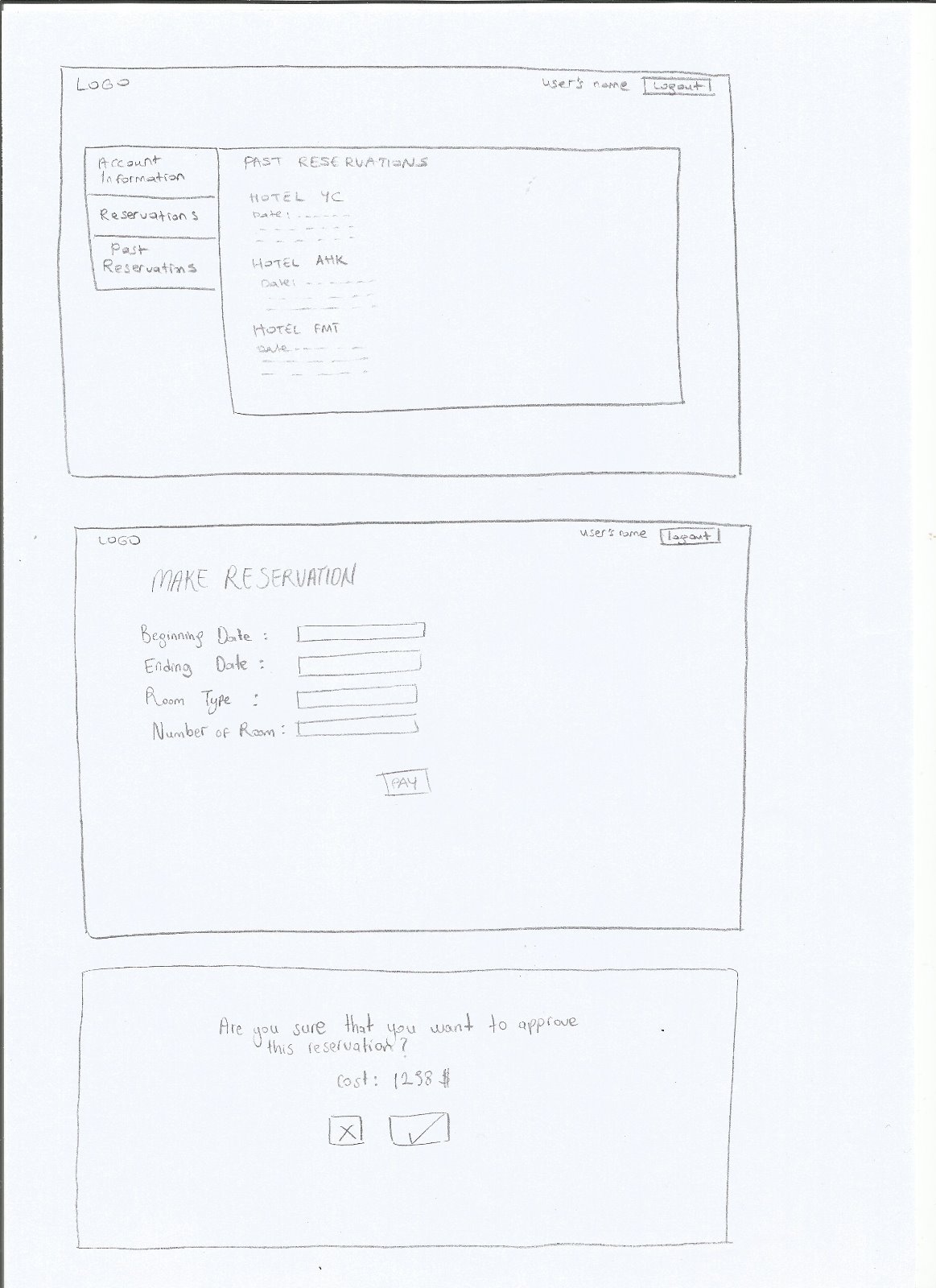
When working with either the analysis object model or the dynamic model, it is essential to remember that these models **represent user-level concepts, not actual software classes or components.**

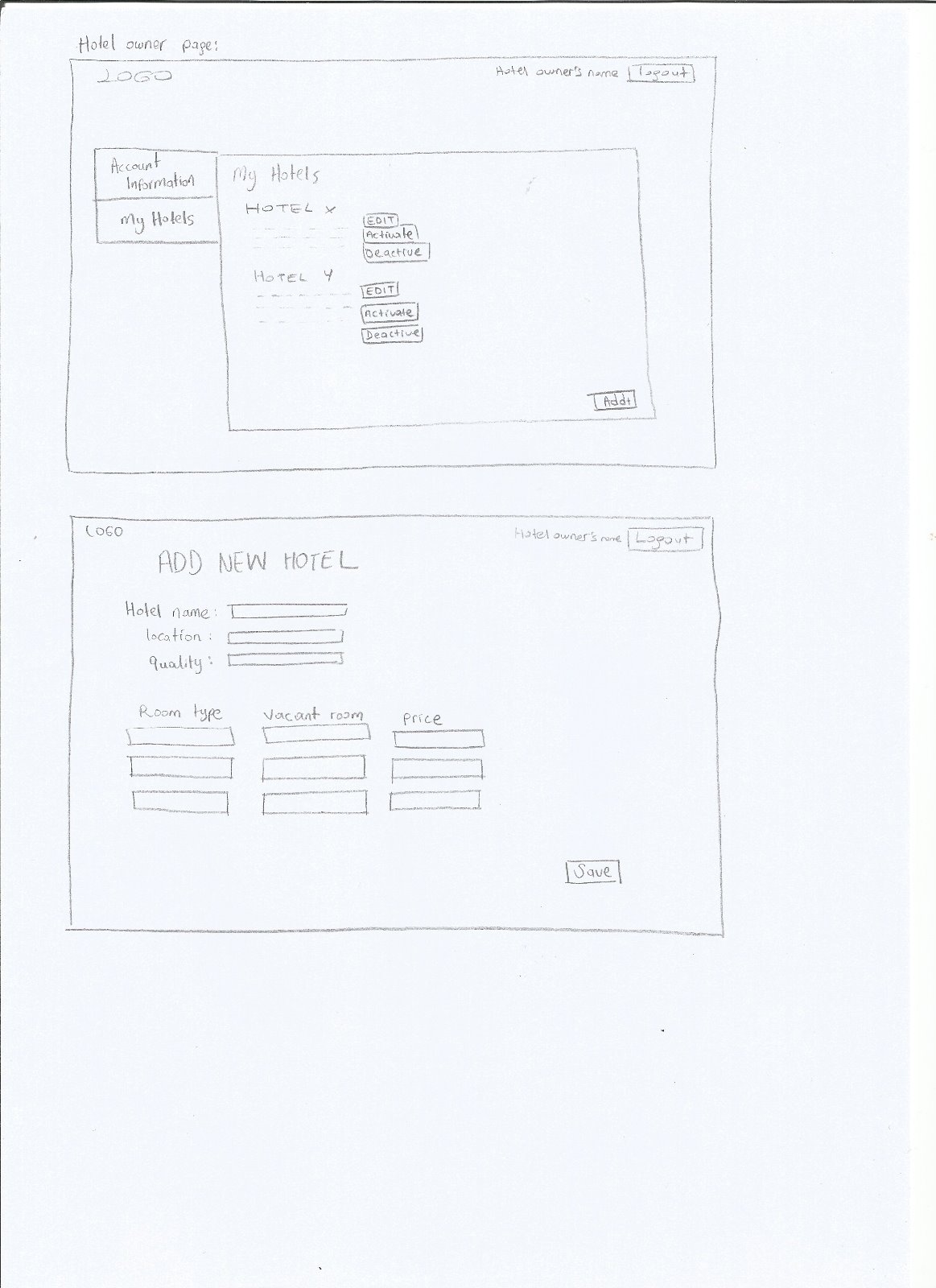
### User interface—navigational paths and screen mock-ups

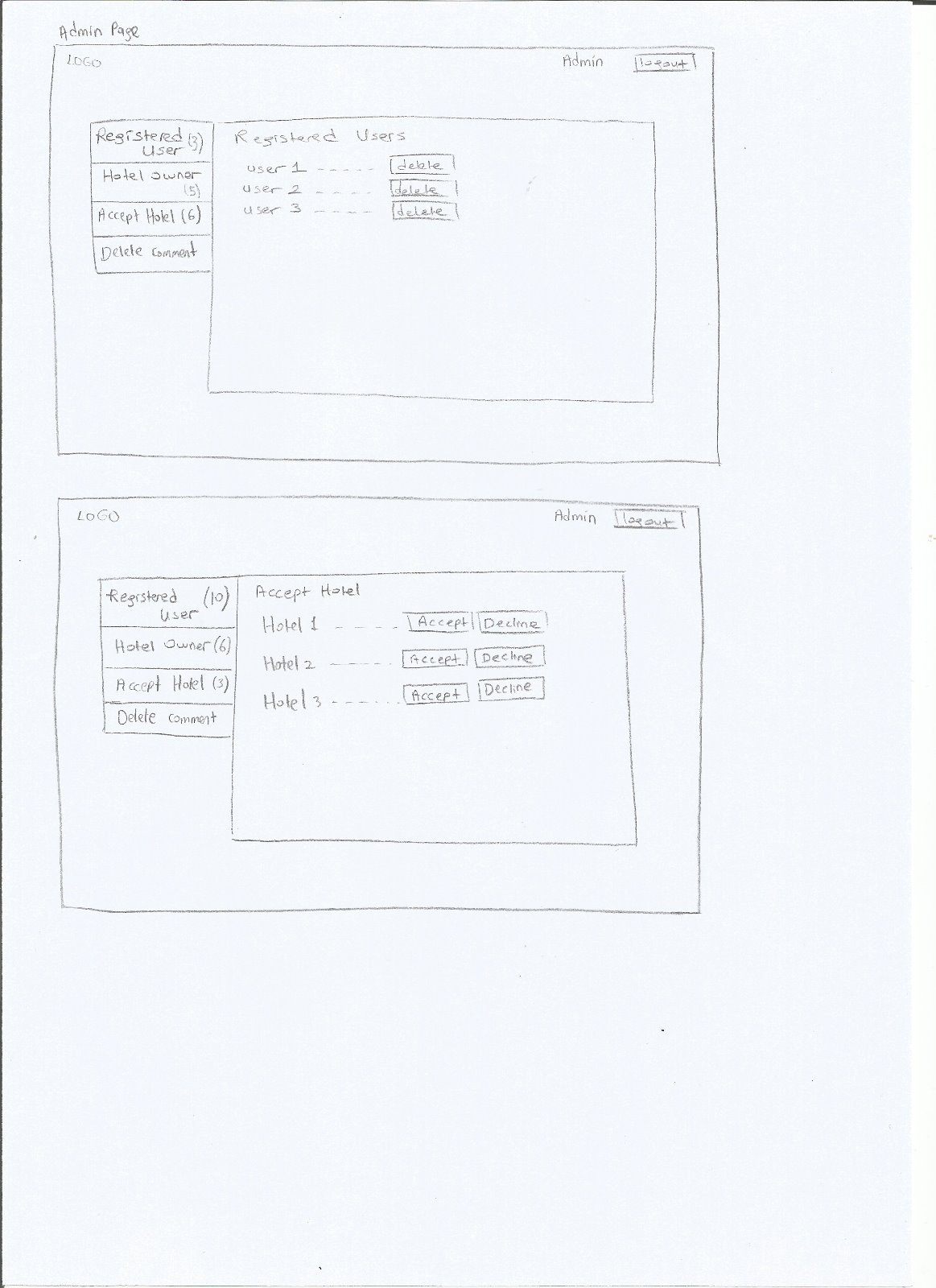


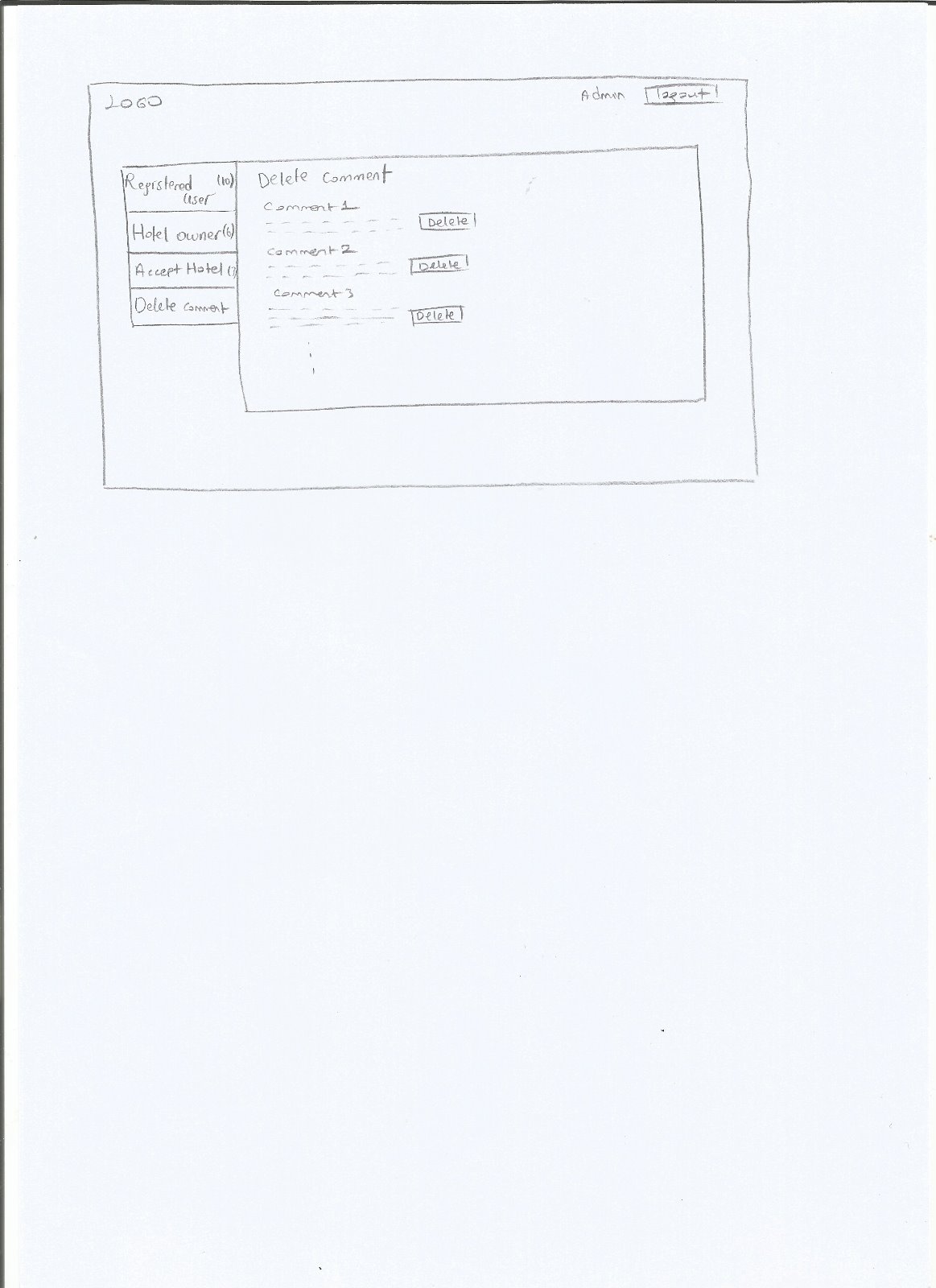












## Project Schedule

Prepare Gannt Chart, and add it to this section.

# Glossary

**A**ccount: When u register with your personal info you will get an account.

**A**dmin: Type of account. He/she can do some certain functions.

**H**otel owner: Type of account, he/she can add hotels to renting room.

**I**nput: Blank fields such as text filed.

**I**nvalid: not true or acceptable, or not correctly thought out.

**L**ogin: Entering tatiloradaburada.com with email and password.

**N**ull field: Empty field.

**P**op-up: Another screen when you click on button it suddenly appears.

**R**egistered user: Type of account, he/she can make reservation for vacation.

**R**egister: You should register to tatiloradaburada.com to do some certain functions.

**R**eservation: 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.

**T**ab: a tag or label.

**V**isitor: While you are using tatiloradaburada.com without login.

# References

1. Bruegge B. & Dutoit A.H.. (2010). *Object-Oriented Software Engineering Using UML, Patterns, and Java*, Prentice Hall, 3rd ed.