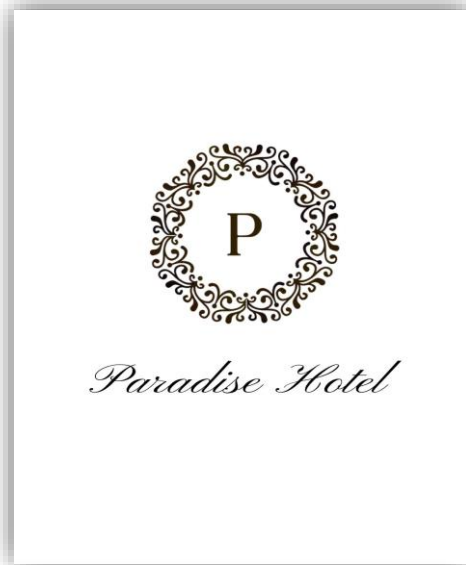


Hotel Reservation System



Uomna Hesham 211001623

Farah Fawki 211001836

Nada Atef 211001921

Mai El-Gazzar 211001757

Dr. Radwa Mohamed Tawfik

Table of Contents

| | |
|---|-----------|
| 1.Introduction..... | 4 |
| 1.1.Purpose..... | 4 |
| 1.2.Scope | 4 |
| 1.2.Technologies Used | 4 |
| 1.2.Intended Audience | 4 |
| 1.3.Overview | 4 |
| 2.Overall Description:..... | 5 |
| 3.Functional Requirements:..... | 5 |
| 4.Non-Functional Requirements: | 6 |
| Interface | 7 |
| 5.1. System Interface..... | 7 |
| 5.2. Software Interface..... | 14 |
| 5.3. Hardware Interface | 14 |
| 6. Diagrams..... | 15 |
| 6.1. Use case Diagram | 15 |
| 6.1.1. Use case scenarios..... | 16 |
| 6.2. Sequence Diagram | 28 |
| 6.2.1 Book Room | 28 |
| 6.2.2 Check Availability..... | 29 |
| 6.3. Class Diagram | 30 |
| 7. Future Work..... | 31 |
| 8. Test cases..... | 32 |

Tables of Figures

| | | |
|-------------------------------------|------------------------------------|-----------------|
| Figure 1: Paradise Hotel | Figure 2: Home Page | 7 |
| Figure 3: Guest Rooms | | 8 |
| Figure 4: Nile-View Room | Figure 5: Pool-View Room | Figure 6: |
| Garden View Room | | 9 |
| Figure 7: Nile-View Single Room | Figure 8: Nile View Double Room | Figure 9: Pool- |
| View Single Room..... | | 10 |
| Figure 10: Pool-View Double Room | Figure 11: Garden-View Single Room | Figure 12: |
| Garden View Double Room..... | | 10 |
| Figure 13: Check Availability | | 11 |
| Figure 14: Room unavailable..... | | 11 |
| Figure 15: Log in | Figure 16: Sign up..... | 12 |
| Figure 17: Payment method | | 12 |
| . Figure 18: Reservation Confirmed | Figure 19: Modify or Cancel | |
| Reservation | | 13 |
| Figure 20: Profile | | 13 |

1.Introduction

1.1.Purpose

A hotel reservation system allows to give guests a fast and convenient way to reserve a room directly from the hotel's website. In addition, allowing guests to amend their booking, recording payment and payment type and accepting reservations for the room service.

1.2.Scope

Hotel reservation system is website that makes booking hotel rooms easier. It allow guests to search for available rooms based on their preferences, such as date, location, price, and room type, also it allow guests to book rooms online and pay securely using various payment methods, such as credit card or bank transfer. It enables to order room service online and track the status of their orders.

1.2.Technologies Used

HTML

CSS

Javascript

MY SQL

1.2.Intended Audience

Guests: who want to book hotel rooms online with ease and convenience. They can also modify or cancel their bookings, order room service, and pay securely using the system.

Hoteliers: who want to increase their online visibility and revenue by offering their rooms on the platform. They can also manage reservations, inventory, pricing, and guest information using the system.

Software Tester: To design and execute all the tests that verify the system's functionality and quality according to the client's specifications.

Project Manager /SCRUM Master: To draw the roadmap for the project and make sure that the objectives and activities are well-defined and that the team collaborates effectively.

1.3.Overview

Hotel reservation website focusing on its features and functionalities. The website serves as a platform for guests to conveniently make room reservations directly from the hotel's site. It emphasizes a user-friendly interface that caters to users of all technical abilities, ensuring a seamless booking experience. The paper highlights key features such as real-time availability updates, 24/7 bookings, and the ability to handle reservations, pricing, and guest.

2.Overall Description:

The hotel reservation website is a user-friendly platform that lets guests easily book rooms directly from the hotel's site. It aims to provide a hassle-free and secure booking experience. With a strong focus on functionality and user-friendliness, it offers real-time room availability updates, flexible booking options, secure payment processing, and the ability to manage reservations, all while adhering to high standards of performance, security, scalability, reliability, and usability.

3.Functional Requirements:

The website boasts a range of functional features to facilitate an efficient booking process:

User Registration and Login:

The website enables users to create personalized accounts by providing their personal information. This feature ensures that registered users have a secure and personalized experience. It simplifies the booking process by storing their details for future reservations.

Search and Book Rooms:

Guests have the ability to search for available rooms effortlessly. They can refine their search based on specific criteria such as dates, location, room type, and price range. Detailed room information, including images, amenities, and prices, is readily accessible. Once guests have found the perfect room, they can book it with ease and receive an instant confirmation, minimizing any uncertainty.

Availability Management:

The website maintains real-time updates on room availability, ensuring that guests can trust the accuracy of the information provided. This system's reliability prevents double bookings and optimizes the overall user experience.

Payment Processing:

Security is paramount. The website provides secure payment processing, accepting various payment methods such as credit cards. This ensures the confidentiality of user data and payment information, enhancing guest trust in the platform.

User Profiles:

Guests can access their user profiles, which allow them to view and update their contact information and preferences. This personalization feature contributes to a more tailored and enjoyable experience.

Cancellation and Modification:

The platform accommodates changes to reservations, allowing guests to cancel or modify bookings within a specified timeframe and adhering to appropriate cancellation policies. This flexibility adds convenience and peace of mind for users.

Booking and Service:

The website is available 24/7, allowing guests to make reservations at any time, catering to various time zones and schedules. Additionally, the platform accepts reservations for room services, enhancing the overall convenience of a guest's stay.

4.Non-Functional Requirements:

The system places significant emphasis on non-functional requirements to ensure a seamless and secure user experience. These include:

Performance: The website responds quickly to user requests, ensuring minimal latency even during peak booking times.

Security: User data and payment information are encrypted and securely stored, and access control mechanisms protect administrative functionalities.

Scalability: The platform is designed to handle an increasing number of rooms as the business grows.

Reliability: The website operates 24/7 with minimal downtime for maintenance, ensuring guests can make reservations at their convenience.

Usability: The user interface is designed to be user-friendly and accessible to a wide range of users, regardless of their technical abilities.

Interface

5.1. System Interface

When the user opens the website, they are going to see home page as shown in Figure 2. This home page extends a warm welcome, offering users a range of options to explore, including Accommodations, Profile, Reservations, and Login functionalities.

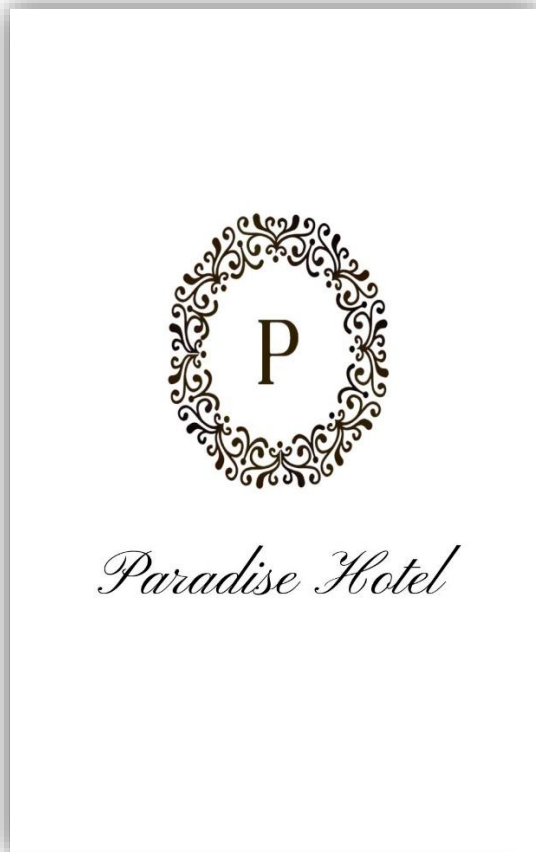


Figure 1: Paradise Hotel

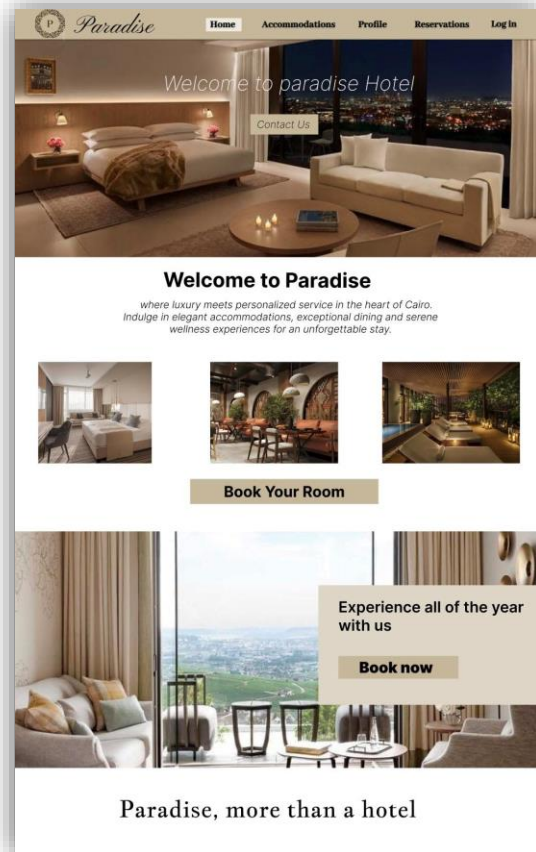


Figure 2: Home Page

When the user picks the accommodations option from the tool bar, it will show the user the room options available, as shown in Figure 3, which are (Nile-View, Pool-View, and Garden View), a brief description of each room, and an option to view details of each room.

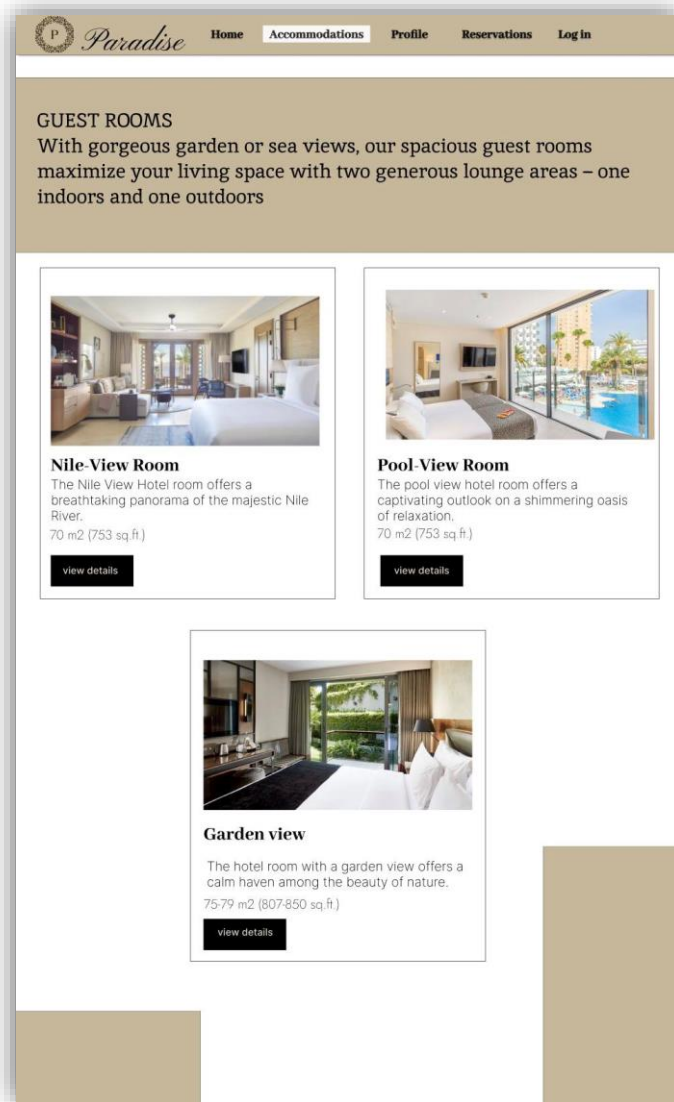


Figure 3: Guest Rooms

When the user presses on the view details options of a specific room, it will take the user to one of the following pages: Figure 4, Figure 5, and Figure 6. Each page has a brief description of the room along with two options to choose from: a single room or a double room. A single room has only one bed and accommodates one person. A double room has two separate beds and can accommodate two people.

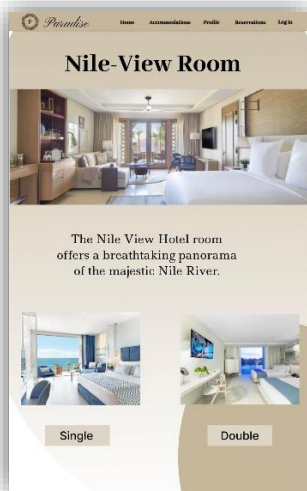


Figure 4: Nile-View Room
Room



Figure 5: Pool-View Room



Figure 6: Garden View

Once the user has chosen their preferred room type, based on what they choose, they will move to one page accordingly, as shown in Figure 7, Figure 8, Figure 9, Figure 10, Figure 11, and Figure 12. Each page shows the details of the room, such as the room size, which varies between two sizes, the amenities provided in the room, and the price of the room per night. The page also provides an option to book the room after viewing its details.



Figure 7: Nile-View Single Room

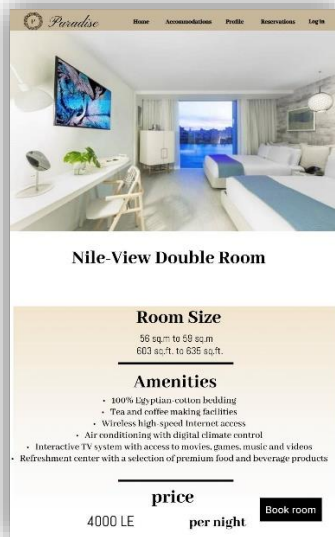


Figure 8: Nile View Double Room

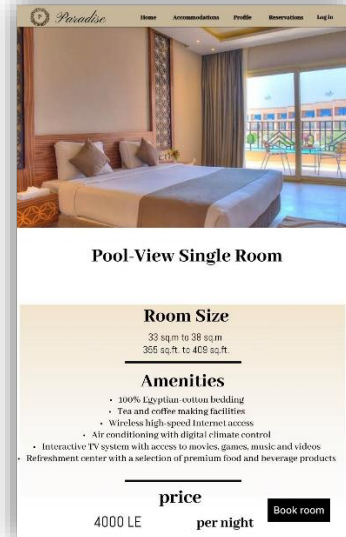


Figure 9: Pool- View Single Room

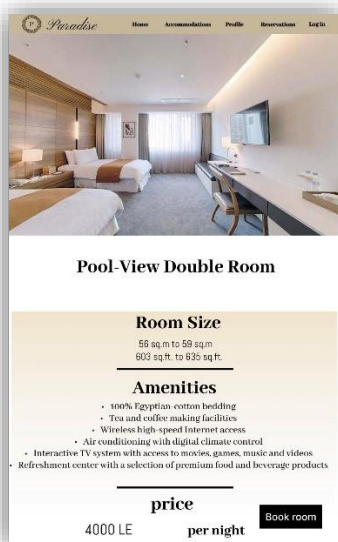


Figure 10: Pool-View Double Room

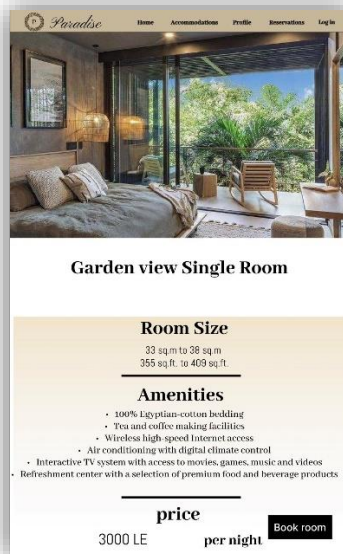


Figure 11: Garden-View Single Room

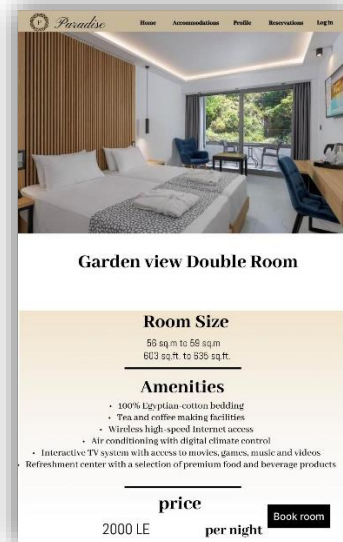
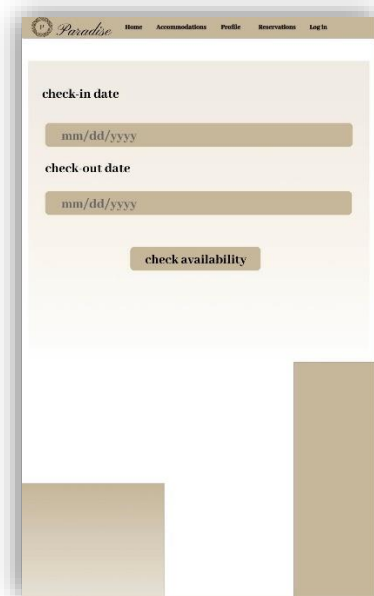


Figure 12: Garden View Double Room

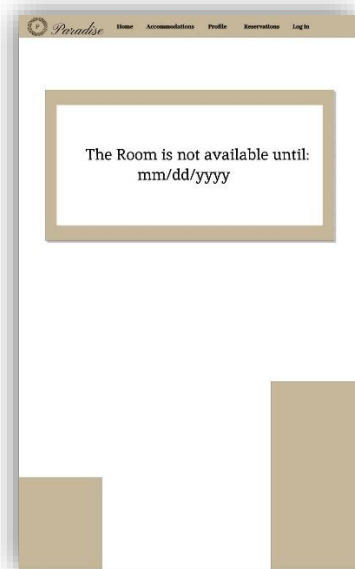
When the user picks the option to book a room, the website asks the user to input the check-in and check-out dates. Then, the user can press "Check Availability" to determine if the room will be occupied within the selected dates or not.



The image shows a mobile application interface for a website named "Paradise". At the top, there is a navigation bar with links: Home, Accommodations, Profile, Reservations, and Login. Below the navigation bar, the interface is divided into two main sections. The first section is titled "check-in date" and contains a text input field with the placeholder "mm/dd/yyyy". Below this, there is a second section titled "check-out date" with another text input field, also with the placeholder "mm/dd/yyyy". At the bottom of this section, there is a button labeled "check availability". The interface is designed with a clean, modern look using a light beige color scheme.

Figure 13: Check Availability

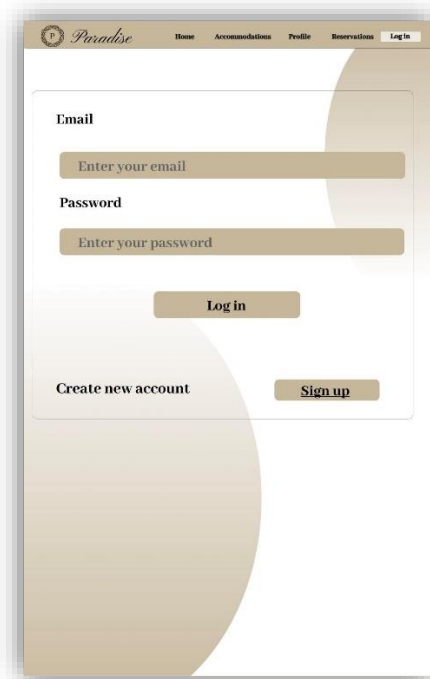
Once a user selects a room for booking, the system displays its availability status. If the room is currently unavailable, a message informs the user that the room is unavailable on this date and specifies the date when the room is expected to become available again as shown in Figure 14. However, if the room is available, the user starts the process to complete the reservation.



The image shows a mobile application interface for a website named "Paradise". At the top, there is a navigation bar with links: Home, Accommodations, Profile, Reservations, and Login. Below the navigation bar, the interface is divided into two main sections. The first section is a message box with a white background and a thin black border, containing the text "The Room is not available until: mm/dd/yyyy". Below this message box, there is a large, empty rectangular area. At the bottom of the interface, there is a button labeled "check availability". The interface is designed with a clean, modern look using a light beige color scheme.

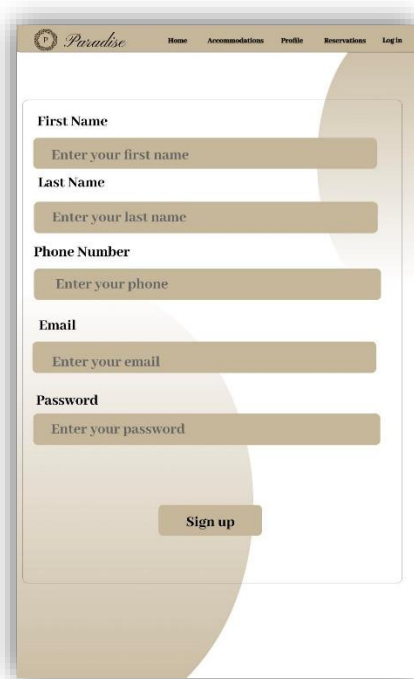
Figure 14: Room unavailable

As shown in Figure 15 and 16, if the room is available, users are required to either log in if they have an existing account or sign up if they don't.



The login form is titled "Paradise" and includes a navigation bar with "Home", "Accommodations", "Profile", "Reservations", and "Log in". The form contains two input fields: "Email" with the placeholder "Enter your email" and "Password" with the placeholder "Enter your password". Below these fields is a "Log in" button. At the bottom of the form, there is a link "Create new account" and a "Sign up" button.

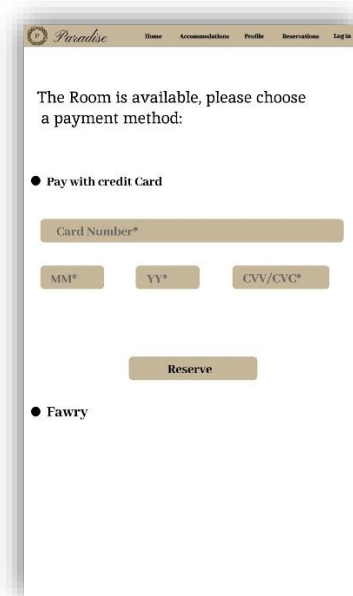
Figure 15: Log in



The sign-up form is titled "Paradise" and includes a navigation bar with "Home", "Accommodations", "Profile", "Reservations", and "Log in". The form contains five input fields: "First Name" with the placeholder "Enter your first name", "Last Name" with the placeholder "Enter your last name", "Phone Number" with the placeholder "Enter your phone", "Email" with the placeholder "Enter your email", and "Password" with the placeholder "Enter your password". Below these fields is a "Sign up" button.

Figure 16: Sign up

In Figure 17, the users choose a payment method: credit card or fawry.



The payment method form is titled "Paradise" and includes a navigation bar with "Home", "Accommodations", "Profile", "Reservations", and "Log in". The form displays the message "The Room is available, please choose a payment method:". Below this message, there are two radio button options: "Pay with credit Card" and "Fawry". The "Pay with credit Card" option is selected. Below this option, there are three input fields: "Card Number*", "MM*", and "YY*", and a "CVV/CVC*" field. Below these fields is a "Reserve" button.

Figure 17: Payment method

In Figure 18, the confirmation of the user's reservation is displayed, including the check-in and check-out dates. If users wish to make any changes to their reservation, they can click on the "Reservations" button and will be directed to Reservations page as shown in Figure 19. In Figure 19, users have the option to modify or cancel their reservation. However, if the users cancel the reservation after the specified date, they will be required to pay a fine.



Figure 18: Reservation Confirmed

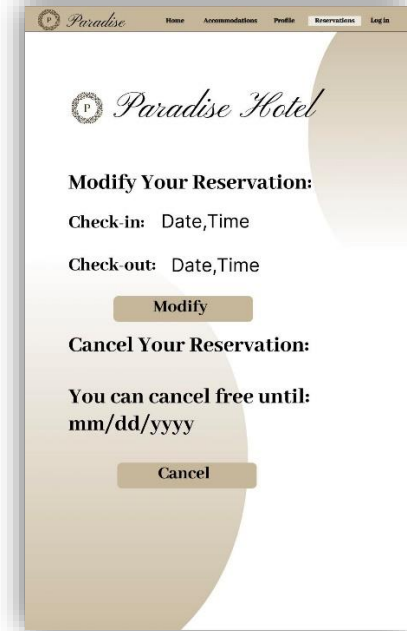


Figure 19: Modify or Cancel Reservation

In Figure 20, it is a profile page and includes the user's information. In addition to the "Log Out" button.



Figure 20: Profile

5.2. Software Interface

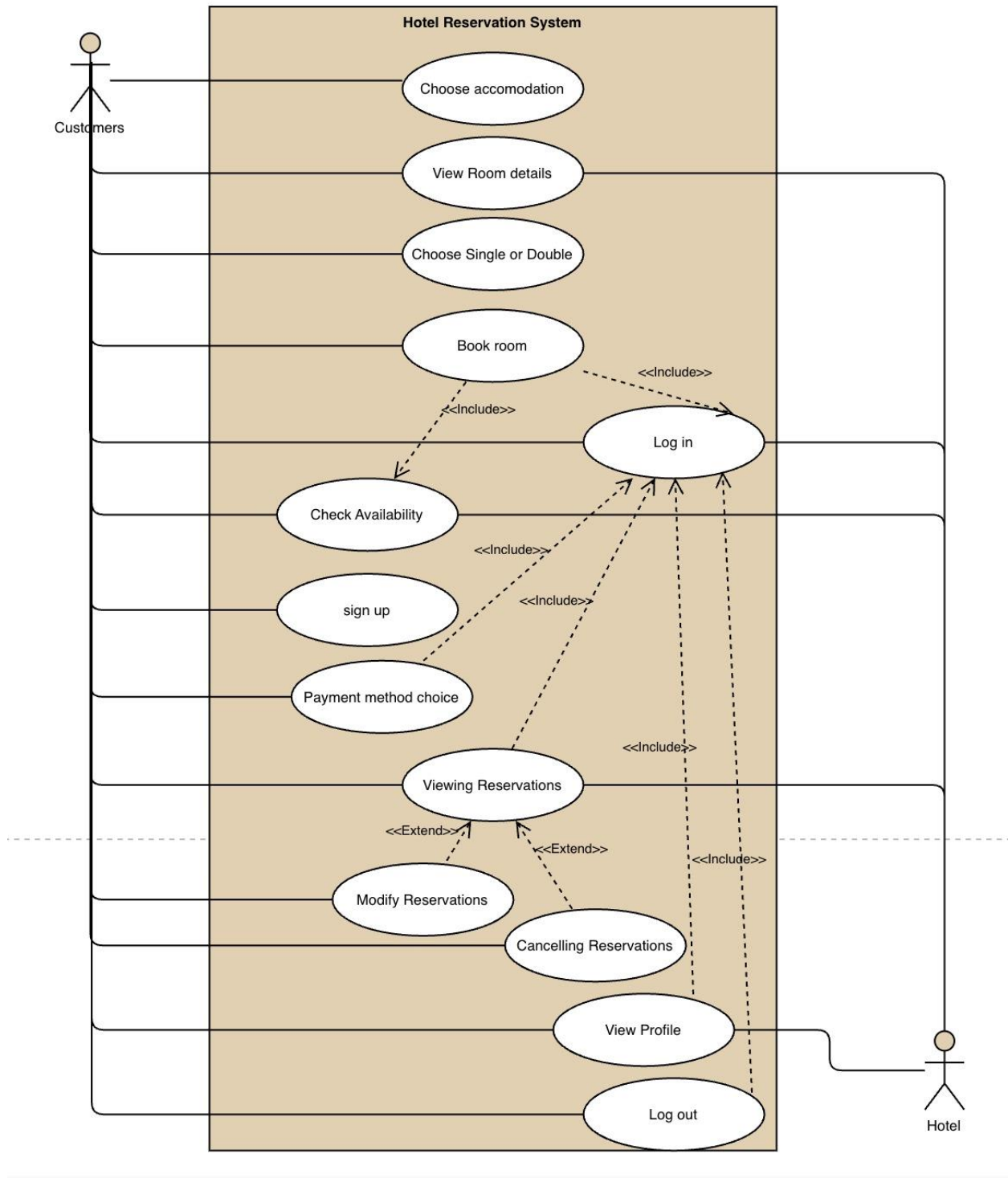
Paradise hotel reservation system is developed using HTML, CSS and java script. It ensures that the website is accessible and usable on various devices, such as smartphones and tablets.

5.3. Hardware Interface

The website requires a stable internet connection to enable users to access the system from anywhere. The system also needs a database interface, as it interacts with a backend database to store and retrieve data related to room availability, reservations, guest information, billing records, and other relevant data.

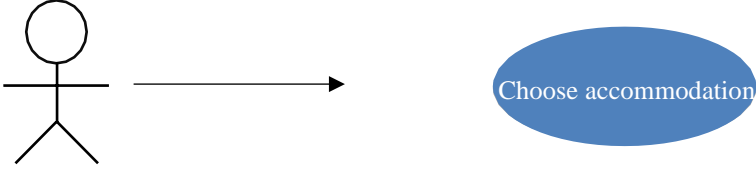
6. Diagrams

6.1. Use case Diagram

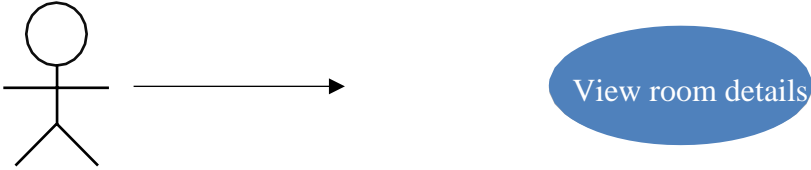


6.1.1. Use case scenarios

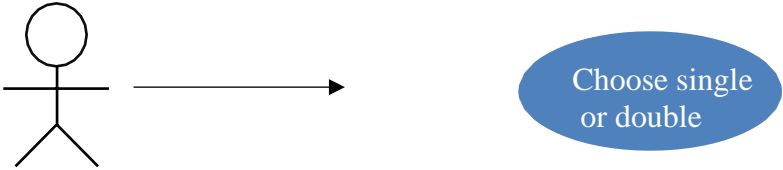
Use Case 1

| | |
|--|--|
|  | |
| User-case Name | Choose accommodation |
| Actors | User (Non-registered) |
| Main success scenario | <ol style="list-style-type: none">1. User picks the "Accommodations" option from the toolbar.2. System displays room options for the user to choose from.3. System displays a brief description of each room.4. System gives the user an option to view details of each room. |

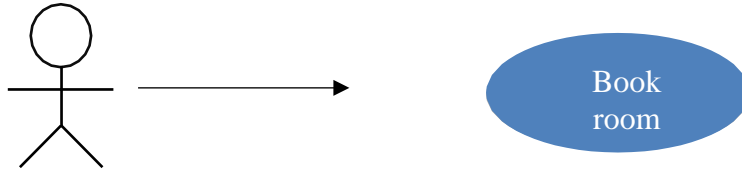
Use Case 2

| | |
|--|---|
|  | |
| User-case Name | View Room details |
| Actors | User (Non-registered) |
| Main success scenario | <ol style="list-style-type: none">1. User presses the "view details" button of the room they choose.2. the system displays the details of the chosen room.3. the system displays 2 options to choose from "single" or "double". |
| Pre-Condition | <ul style="list-style-type: none">• user has to choose one of the room options |

Use Case 3

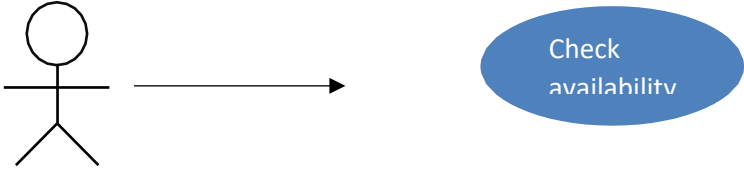
| | |
|--|--|
|  <p>The diagram shows a stick figure actor on the left. An arrow points from the actor to a blue oval use case on the right. The use case contains the text "Choose single or double".</p> | |
| User-case Name | choose single or double |
| Actors | User(Non-registered) |
| Main success scenario | <ol style="list-style-type: none">1. the user choose between a single room or a double room by clicking its button2. system opens the page that displays details of the chosen room |

Use Case 4

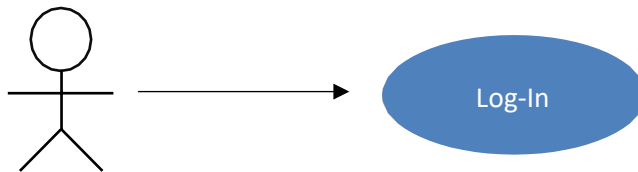


| | |
|------------------------------|---|
| User-case Name | Book room |
| Actors | User (Non-registered) |
| Main success scenario | <ol style="list-style-type: none">1. User presses the "book room" button.2. the website asks the user to input the check-in and checkout dates. |
| Exceptions | <ol style="list-style-type: none">1. User clicks on the "Check Availability" button without filling all of the required information "check-in and checkout dates".2. check-in date or/and checkout date is/are not valid. |
| Actions | <ol style="list-style-type: none">2.1. System display to the user/shelter an alert "Username or Password are not valid".2.2. User/shelter enters the correct username or password.3.1. System display to the user an alert "Please fill in all the required fields".3.2. User/shelter fills the missing information. |
| Pre-Condition | <ul style="list-style-type: none">• User has to pick a room from accommodations.• User has to pick the type of the room. |
| Post Condition | <ul style="list-style-type: none">• User press check availability to complete booking process. |

Use Case 5

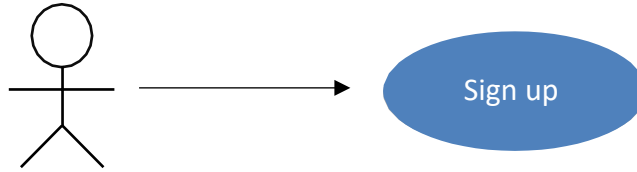
| | |
|--|---|
|  | |
| User-case Name | Check Availability |
| Actors | User/ Hotel (Non-registered) |
| Main success scenario | <ol style="list-style-type: none">1. user presses "Check Availability" button.2. system checks if the room will be occupied within the selected dates or not. |
| Actions | <ol style="list-style-type: none">1. If the room is currently unavailable system display to the user the message "The room is not available until.." and specifies the date when the room is expected to become available again.2. if the room is available the system takes the user to the log in page |
| Pre-Condition | <ul style="list-style-type: none">• User have to input the check-in and checkout dates. |
| Post Condition | <ul style="list-style-type: none">• The system takes the user to log in page if the room is available to complete reservation. |

Use Case 6



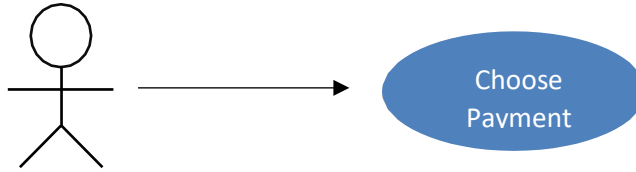
| | |
|------------------------------|---|
| User-case Name | Log-In |
| Actors | User/Hotel (Registered) |
| Main success scenario | <ol style="list-style-type: none">1.System ask user/hotel to enter email and password and log in or create new account and sign up.2.User clicks "Log in".3.User enters email and password then click log in.4.System check if email and password are valid. |
| Exceptions | <ol style="list-style-type: none">3a. Email or password aren't valid3b. User click log in without filling required data |
| Actions | <ol style="list-style-type: none">3.1. System display an alert "Email or password aren't valid"3.2. User enter correct email or password3.3. System display an alert "please fill all required data"3.4. User fills all data |
| Pre-Condition | <ul style="list-style-type: none">• User should open the website• User should have an account |
| Post Condition | <ul style="list-style-type: none">• User is logged in to the system• User have access to the system functions |

Use Case 7



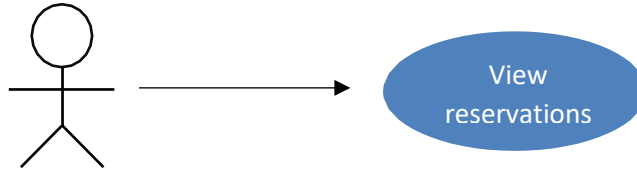
| | |
|------------------------------|--|
| User-case Name | Sign up |
| Actors | User (Non-Registered) |
| Main success scenario | <ol style="list-style-type: none">1. System ask user to enter email and password and log in or create new account and sign up.2. User clicks "Sign Up."3. User provides necessary details like: name, phone number, email and password.4. User gives the system the required data and click sign up.5. System checks data for accuracy and completeness. |
| Exceptions | <ol style="list-style-type: none">4a. User clicks sign up without filling the data.5a. Data isn't accurate like email exists or phone exists. |
| Actions | <ol style="list-style-type: none">4.1. System display an alert "Please fill in all the required fields".4.2. User fills the missing information.5.1. System display to the user "Email already registered".5.2. User enters another email. |
| Pre-Condition | <ul style="list-style-type: none">• User should open the website |
| Post Condition | <ul style="list-style-type: none">• Another user is added to the system.• User is successfully registered.• User information is stored in the database. |

Use Case 8



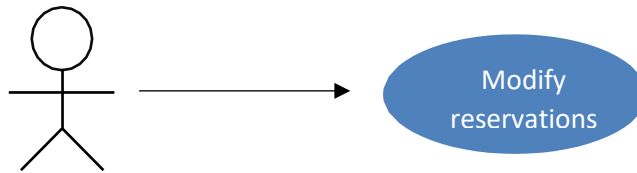
| | |
|------------------------------|--|
| User-case Name | Payment method choice |
| Actors | User (Registered) |
| Main success scenario | <ol style="list-style-type: none">1.User clicks on payment button.2.System ask the user to choose payment method.3.if user choose pay by credit, the system asks the user to write card number and validation number.4.if user choose pay by fawry, the system will display a number to pay on it.5.User clicks Reserve.6.System checks that the data is correct. |
| Exceptions | <ol style="list-style-type: none">3a. Card number or validation number isn't correct. |
| Actions | <ol style="list-style-type: none">3.1. System display an alert "The card number or validation number aren't correct".3.2. User enters the correct card number or validation number. |
| Pre-Condition | <ul style="list-style-type: none">• User should be logged-in. |
| Post Condition | <ul style="list-style-type: none">• User pay for the room successfully.• User reserve the room. |

Use Case 9



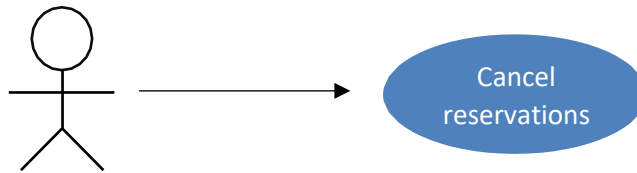
| | |
|------------------------------|--|
| User-case Name | View reservations |
| Actors | User (Registered) |
| Main success scenario | <ol style="list-style-type: none">1. User clicks on Reservations button.2. System displays check in and check out Date for each reservation.3. If User want to make changes, click reservations. |
| Exceptions | 2a. Date isn't accurate |
| Actions | <ol style="list-style-type: none">2.1. System display an alert "Date isn't accurate".2.2. User enters the correct date. |
| Pre-Condition | <ul style="list-style-type: none">• User should be logged-in |
| Post Condition | <ul style="list-style-type: none">• User reserve the room |

Use Case 10



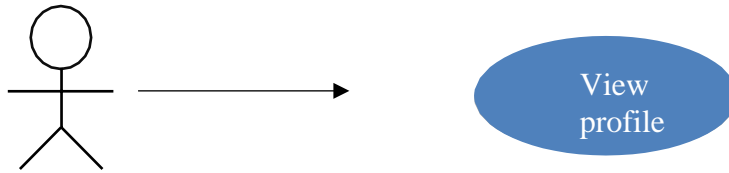
| | |
|------------------------------|--|
| User-case Name | Modify reservations |
| Actors | User (Registered) |
| Main success scenario | <ol style="list-style-type: none">1.User clicks on Reservations button.2.System ask the user to modify check in and check out Date.3.User clicks modify. |
| Exceptions | <ol style="list-style-type: none">2a. Date isn't accurate. |
| Actions | <ol style="list-style-type: none">2.1. System display an alert "Date isn't accurate".2.2. User enters the correct date. |
| Pre-Condition | <ul style="list-style-type: none">• User should be logged-in |
| Post Condition | <ul style="list-style-type: none">• User modify the date and reserve the room |

Use Case 11



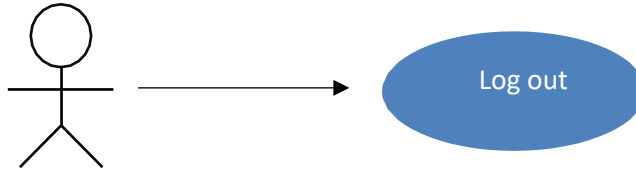
| | |
|------------------------------|---|
| User-case Name | Cancel reservations |
| Actors | User (Registered) |
| Main success scenario | <ol style="list-style-type: none">1. User clicks on Reservations button.2. System tells the user the date that can cancel free until this date.3. User clicks cancel. |
| Exceptions | <ol style="list-style-type: none">2a. Date that the user can cancel is ended. |
| Actions | <ol style="list-style-type: none">2.1. System display an alert "You will pay a fine". |
| Pre-Condition | <ul style="list-style-type: none">• User should be logged-in |
| Post Condition | <ul style="list-style-type: none">• User cancel the reservation for the room |

Use Case 12



| | |
|------------------------------|---|
| User-case Name | View profile |
| Actors | User /Hotel (Registered) |
| Main success scenario | <ol style="list-style-type: none">1. User picks the "profile" option from the toolbar.2. The system displays the user information “name, email, and phone number”.3. The system displays the option to log out the profile. |
| Exceptions | <ol style="list-style-type: none">1. The user is not logged in.2. The user does not have a profile “not signed up”. |
| Actions | <ul style="list-style-type: none">• If the user tries to view profile without logging in the system asks it to log in. |
| Pre-Condition | <ul style="list-style-type: none">• User has to be logged-in. |
| Post Condition | <ul style="list-style-type: none">• Customer details will be displayed. |

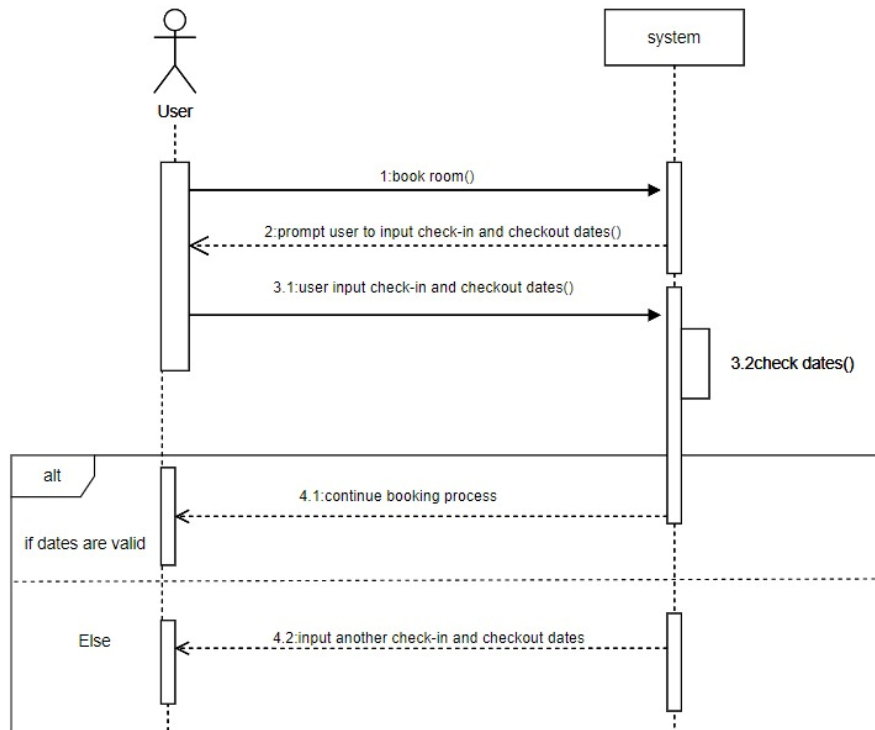
Use Case 13



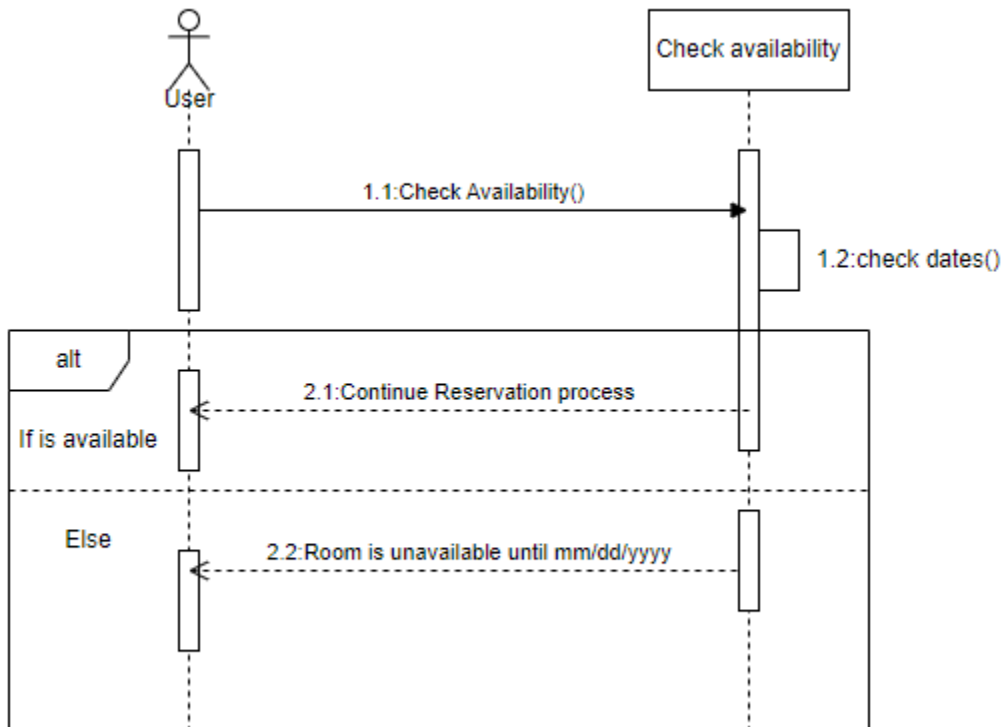
| | |
|------------------------------|---|
| User-case Name | Log out |
| Actors | User (Registered) |
| Main success scenario | <ol style="list-style-type: none">1. User clicks on log out.2. System confirms the user's intention to log out.3. Upon confirmation, the system ends the user's current session.4. The system redirect the user to the login page. |
| Exceptions | <ol style="list-style-type: none">2a. User click cancel log out. |
| Actions | <ol style="list-style-type: none">2.1. System display to the User an alert "Are you sure you want to cancel log out".2.2. If user clicks yes the system cancel log out process , if user clicks No the system will continue log out process. |
| Pre-Condition | <ul style="list-style-type: none">• User should to open the website.• User should have an account.• User should be logged in the system. |
| Post Condition | <ul style="list-style-type: none">• User is logged out from the system. |

6.2. Sequence Diagram

6.2.1 Book Room



6.2.2 Check Availability



6.3. Class Diagram



7. Future work

- **Payment method function:**

This feature incorporates two procedures: one for Fawry and the other for credit card transactions. The system prompts the customer to select a payment method and subsequently requests input regarding the specific details associated with the chosen method.

- **Modify Reservation:**

This feature enables customers to make changes to their reservation, allowing for the adjustment of check-in and check-out dates. It also allows modifications to the reservation, ensuring there are no conflicts with other dates.

- **Cancel Reservation:**

This functionality allows customers to cancel their reservation. Cancellation is free up to a certain date, and beyond that date, customers are required to pay a penalty.

- **View Profile:**

This feature provides the customer with his/her information, where they can view the information that was provided during the sign-up process.

8- Test Cases

Test Case ID: Fun_5.1

Test Designed by: <Name>

Test Priority (Low/Medium/High): Med

Test Designed date: <Date>

Module Name: Check availability function

Test Executed by: <Name>

Test Title: Verify check in and check out dates

Test Execution date: <Date>

Description: check if room is available within entered check in and check out dates that the room is not reserved in

Pre-conditions: User have to input the check-in and checkout dates

Dependencies:

| Step | Test Steps | Test Data | Expected Result | Actual Result | Status (Pass/Fail) | Notes |
|------|---------------------------------|----------------------------|-------------------|-----------------------------------|--------------------|--------------------------------|
| 1 | Navigate to accommodations page | Check in date: 13/11/2023 | Room is available | As expected | Pass | If the user already logged in. |
| 2 | Click on view details button | Check out date: 20/11/1023 | | Navigate to choose payment method | | |
| 3 | Choose type of the room | | | | | |
| 4 | Click on book room button | | | | | |
| 5 | Enter check in date | | | | | |
| 6 | Enter check out date | | | | | |
| 7 | Click on check availability | | | | | |

Post-conditions: User entered an available check in and out date and complete reservation process.

Test Case ID: Fun_5.2
Test Priority (Low/Medium/High): Med
Module Name: Check availability function
Test Title: Verify check in and check out dates
Description: check if the room is available within entered check in and check out dates that the room is already reserved in.

Test Designed by: <Name>
Test Designed date: <Date>
Test Executed by: <Name>
Test Execution date: <Date>

Pre-conditions: User have to input the check-in and checkout dates
Dependencies:

| Step | Test Steps | Test Data | Expected Result | Actual Result | Status (Pass/Fail) | Notes |
|------|-----------------------------|----------------------------|-----------------------|---|--------------------|-------|
| 1 | Navigate to rooms page | Check in date: 13/11/2023 | Room is not available | system display to the user the message "The room is not available until..." and specifies the date when the room is expected to become available again. | Pass | |
| 2 | Click on view details | Check out date: 17/11/1023 | | As expected | | |
| 3 | Choose type of the room | | | | | |
| 4 | Click on book room button | | | | | |
| 5 | Enter check in date | | | | | |
| 6 | Enter check out date | | | | | |
| 7 | Click on check availability | | | | | |

Post-conditions: User will choose other check in and out dates and check the availability again.

Test Case ID: Fun_8
Test Priority (Low/Medium/High): Med
Module Name: Payment method choice
Test Title: Verify payment with valid card number and validation number
Description: Test the validity of the card's numbers

Test Designed by: <Name>
Test Designed date: <Date>
Test Executed by: <Name>
Test Execution date: <Date>

Pre-conditions: User must be logged in.
Dependencies:

| Step | Test Steps | Test Data | Expected Result | Actual Result | Status (Pass/Fail) | Notes |
|------|---|---------------------------|---|---------------|--------------------|-------|
| 1 | User chooses credit card as a payment method | Card number:176347191 | Payment is confirmed and room is reserved | As expected | Pass | |
| 2 | User writes card number and validation number | Validation number:2439741 | | | | |
| 3 | Click on reserve button | | | | | |

Post-conditions: User complete payment for the room successfully and reserve the room.

Test Case ID: Fun_6

Test Designed by: <Name>

Test Priority (Low/Medium/High): Med

Test Designed date: <Date>

Module Name: Hotel website login screen

Test Executed by: <Name>

Test Title: Verify login with valid email and password

Test Execution date: <Date>

Description: Test the hotel website's login page

| Step | Test Steps | Test Data | Expected Result | Actual Result | Status (Pass/Fail) | Notes |
|------|--------------------------|--------------------------|------------------------------|-----------------|--------------------|---|
| 1 | Navigate to login page | Email= example@gmail.com | User should be able to login | Not as expected | Fail | User entered invalid email and password |
| 2 | Provide correct email | Password: 278hfe32 | | Login failure | | |
| 3 | Provide correct password | | | | | |
| 4 | Click on Login button | | | | | |
| | | | | | | |

Post-conditions: User is logged in to the system, the account session details are logged in database, and the user has access to more system functions.

Test Case ID: Fun_7

Test Designed by: <Name>

Test Priority (Low/Medium/High): Med

Test Designed date: <Date>

Module Name: Hotel website sign up screen

Test Executed by: <Name>

Test Title: Verify sign up with valid email and password

Test Execution date: <Date>

Description: Test the hotel website's sign up page

Pre-conditions: User should open the website and shouldn't have signed up before

Dependencies:

| Step | Test Steps | Test Data | Expected Result | Actual Result | Status (Pass/Fail) | Notes |
|------|-----------------------------|--------------------------|--------------------------------|---------------------------|--------------------|-------|
| 1 | Navigate to login page | First name = Sarah | User should be able to sign up | As expected | Pass | |
| 2 | Provide first and last name | Last name = Mohamed | | User is navigated to | | |
| 3 | Provide valid password | Phone number =010***** | | dashboard with successful | | |
| 4 | Click on Login button | Email= example@gmail.com | | sign up | | |
| | | Password: 278hd | | | | |

Post-conditions:

The new user is added to users' database and the user successfully signed up to the hotel website.

Test Case ID: Fun_10

Test Designed by: <Name>

Test Priority (Low/Medium/High): Med

Test Designed date: <Date>

Module Name: Modify reservations function

Test Executed by: <Name>

Test Title: Modify check in or/ and check out dates

Test Execution date: <Date>

Description: Test if the dates are can be modified when the user modify them by entering new check in and out dates

Pre-conditions: User should be logged-in

Dependencies:

| Step | Test Steps | Test Data | Expected Result | Actual Result | Status (Pass/Fail) | Notes |
|------|---|----------------------------|--|--|--------------------|-------|
| 1 | Navigate to Reservations page | Check in date: 13/11/2023 | Check in and check out dates Are modified | As expected | Pass | |
| 2 | Click modify button | Check out date: 17/11/1023 | | User is navigated to reservations page to see the | | |
| 3 | Enter new check in and check out dates | | | modified check in and check out dates | | |
| 4 | Check availability again | | | | | |
| | | | | | | |

Post-conditions:

Check in and check out dates are modified and the new dates are displayed in Reservations page.