SECJ2203: Software Engineering

System Documentation (SD)

Hotel Booking System

Version 1.0

6 May 2021

School of Computing, Faculty of Engineering

Prepared by: Bara Api (Group 8)

Revision Page

a. **Overview**

The content of the current version of System Documentation includes the introduction of a hotel booking system named Sunny Hotel. Other than that, this document also includes the external Interface Requirements, System Features, design and others.

b. **Target Audience**

The target audience for this project are students and staff of UTM.

c. **Project Team Members**

Member Name	Role	Task	Status
Muhammad Haziq bin Azli	Leader	Definitions, Acronym and Abbreviation, Performance and Other Requirements, State machine Diagram, Use case for Make Payment and Make Reservation, Specific Requirement of Hardware Interfaces	Completed
Nurul Najwa Binti Hussein	Member	Purpose, References, User Interfaces, Use case diagram, Use case of search hotel, Check Availability and Cancel Booking.	Completed
Nur Azizah binti Mohammad Mokhtar	Member	Communication Interface, Domain Model, Design Constraints, Software System Attributes and Use case o register/login and extended charge	Completed
Difa Ega Adrian	Member	Scope, overview, Software Interface, activity diagram, user case chek-in/check-out information, UC favorite/whishlist, UC customer booking history.	Completed

d. **Version Control History**

Version	Primary Author(s)	Description of Version	Date Completed
1.0	Team leader 1 (Muhammad Haziq bin Azli)	Completed Chapter 1 and 2	06/05/2021

Note:
This System Documentation (SD) template is adapted from IEEE Recommended Practice for Software Requirements Specification (SRS) (IEEE Std. 830-1998), Software Design Descriptions (SDD) (IEEE Std. 10161998 1), and Software Test Documentation (IEEE Std. 829-2008) that are simplified and customized to meet the need of SECJ2203 course at School of Computing, UTM. Examples of models are from Arlow and Neustadt (2002) and other sources stated accordingly.

Table of Contents

Introduction

System documentation is a vital and important part of successful software development and software engineering. It is a collection of documents that describes the requirements, capabilities, design, operation, and maintenance of a system, such as a communications, computing, or information processing system.

1.1 **Purpose**

The purpose of documentation is to provide the necessary information, such as the design, operation, capabilities of the system to develop training programs for operators and users. Other than that, system documentation also makes the system modification and implementation easier and can narrow down the communication gaps among users, designers and management.

1.2 **Scope**

The basic of this system that we will develop named "**Sunny Hotel**" is for booking a hotel through a website or application that we are designing, with the system that we are making will facilitate users or customers in finding the right hotel and the best, and of course this system will facilitate people in making hotel reservations.

The system that we make can help customers in finding hotels that want to be booked, our application or website will also display the best hotels that will be recommended to customers, and in our system customers can do a favorite / wishlist to hotels they want to book, and on this application or website, the user can arrange hotel reservations such as arranging or changing the check-in or check-out date at the hotel, and also the customer can also be facilitated because it can make payments on the application by selecting several methods or several types of payments.

Customer can use this application to search for hotels by entering all criteria and details such as room type and bed size according to the customer's wishes, this application will also provide information about room availability at the hotel that the customer chooses, and also the customer can arrange a booking such as confirming the booking or canceling the booking, and the feature that we provide in this application is, customers can enter the hotel into the favorite / wishlist. The purpose

of this system is to facilitate customers in making transactions, and can also help the hotel in making it easier to arrange hotel reservations by customers.

1.3 **Definitions, Acronyms and Abbreviation**

XAMPP

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, and it consists mainly of Apache HTTP Server, MariaDB database and interpreters for scripts written in the PHP and Perl programming languages. XAMPP is an abbreviation for cross-platform, Apache, MySQL, PHP and Perl, and it allows you to build WordPress sites offline, on a local web server on your computer.

Gmail

Gmail is a free Web-based email service that provides users with a gigabyte of storage for messages and provides the ability to search for specific messages. The Gmail program also automatically organizes successively related messages into a conversational thread. Gmail is an abbreviation of Google web email service.

1.4 **References**

XAMPP Tutorial: How to use XAMPP to set up WordPress on localhost- UndsgnTM.(2018, March 23). Undsgn™. https://undsgn.com/xampp-tutorial/#:~:text=XAMPP%20is%20an%20abbreviation%20for,web%20server%20on%20your%20computer.

What does GMAIL mean? - GMAIL Definition - Meaning of GMAIL - InternetSlang.com. (2021). Internetslang.com. https://www.internetslang.com/GMAIL-meaning-definition.asp#:~:text=GMAIL%20means%20%22Google%20web%20email%20service%22

System Documentation: Features, Purpose and Contents | MIS. (2015, December 14). Your

Article Library. https://www.yourarticlelibrary.com/management/mis-management/system-documentation-features-purpose-and-contents-mis/70408

Hotel Booking Mobile App User Interfaces. (2014). Softwareadvice.com. https://www.softwareadvice.com/hotel-management/spotlight/mobile-booking-apps-2014/

Wikipedia Contributors. (2021, May 4). *Desktop computer*. Wikipedia; Wikimedia Foundation. https://en.wikipedia.org/wiki/Desktop_computer

Smartphone. (2020). Techterms.com. https://techterms.com/definition/smartphone

Ethernet. (2020). Techterms.com. https://techterms.com/definition/ethernet

1.5 **Overview**

The writing of this document is divided into several chapters as follows:

Chapter 1 contains an introduction, explaining the software's purpose, scope, glossary, references, and document description/overview.

Chapter 2 explains the general description, giving an overview of the functionality of the product. Describes informal needs and is used to describe relationships for needs technically in chapter 3.

Chapter 3 contains the needs of specifications, written primarily for developers and explains the technical terms of product functionality details**.**

2. Specific Requirements

2.1 External Interface Requirements

2.1.1 User Interfaces

Describe how the system will interact with its users:

I. Registration interface

For the new user, they will have to register first in the system. The registration interface consists of a form that allows the user to fill in their personal information. The personal information that should be keyed in are username, password, full name, contact number, home address and also their email address. The user will receive the authentication email from the system once they select the register button.

II. Login interface

For the existing system, they should log in by using the registered email and password. They have to fill in the email and password in input box provided to allow them to access their account. Then, in this interface, it is also provide the “log in” button (for signing in the website) and the “Forgot password?” if the user forget their password. In that feature, the system will send the reset password email to the user within30 seconds to one minute.

III. Home interface

In the home interface, the user can search the hotel available in the search box. There are also dates and types of room input box. The promotion deals and top destinations also are shown in this interface.

IV. List of hotel interface

The list of hotels that are being searched according to the search detail will be shown in this interface. The hotel also shows the little details and allows the user to add the hotel to the cart or wish list.

V. Cart interface

The cart interface will show the hotel that was chosen by the user. The users are allowed to change the duration of staying in the hotel, type of room and date.

2.1.2 Hardware Interfaces

1. **Desktop** is the main working space on your computer screen. It is where the icons for the files and folders on your hard drive is displayed. It is mostly used in the office to do computational work.
    - Name: Desktop
    - Specification: Windows, Mac OS, Linux
    - Source: https://en.wikipedia.org/wiki/Desktop_computer

2. **Smartphone** is a mobile phone that includes advanced functionality beyond making phone calls and sending text messages. Most smartphones have the capability to display photos, play videos, check and send email, and surf the Web. Unlike a desktop, a smartphone is convenient to bring wherever we go due to its small size. Even though it has smaller storage and limited capabilities compared to a desktop, it is the most convenient gadget to bring wherever we go.
    - Name: Smartphone
    - Specification: Android, iOS
    - Source: https://techterms.com/definition/smartphone

3. **Ethernet** is a family of wired computer networking technologies commonly used in local area networks (LAN), metropolitan area networks (MAN) and wide area networks (WAN). Ethernet is the standard way to connect computers on a network over a wired connection. It provides a simple interface and for connecting multiple devices, such computers, routers, and switches. Ethernet cable has RJ45 connector on both ends. Ethernet jacks look similar to telephone jacks only slightly wider.
    - Name: Ethernet cable
    - Specification: CAT5, CAT5e, CAT6
    - Source: https://techterms.com/definition/ethernet

2.1.3 Software Interfaces

1. **XAMPP** is a local web server used to develop web applications offline on a local host computer and also XAMPP can act as a database system, This software is commonly used for testing web applications through localhost. Web applications developed natively, using frameworks, or CMS can be tested using XAMPP.

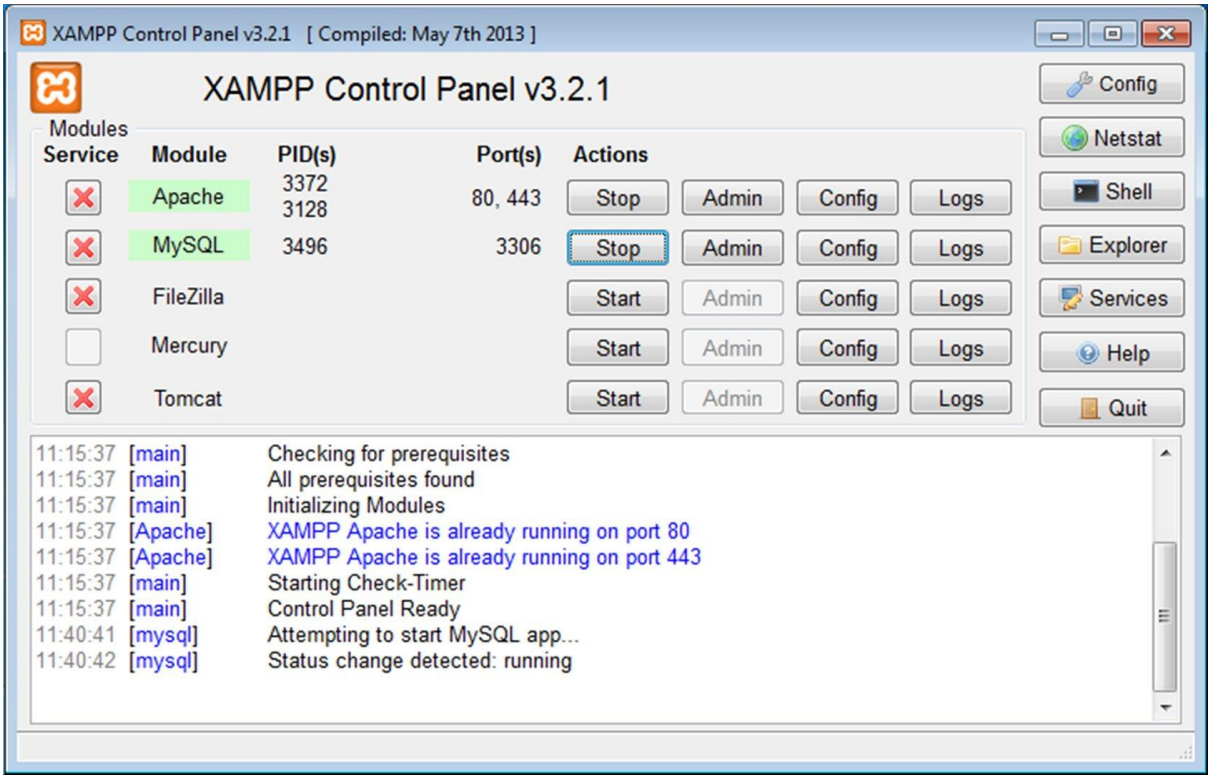
   • Name: XAMPP

   • Mnemonic: XAMPP

   • Specification number: -

   • Version number: XAMPP PHP 8.0.5

   • Source: https://www.apachefriends.org/



2. This system uses **Google mail** to send or update all transactions that have been done by the customer such as making a payment or confirming a hotel room booking.

    • Name: Google Mail

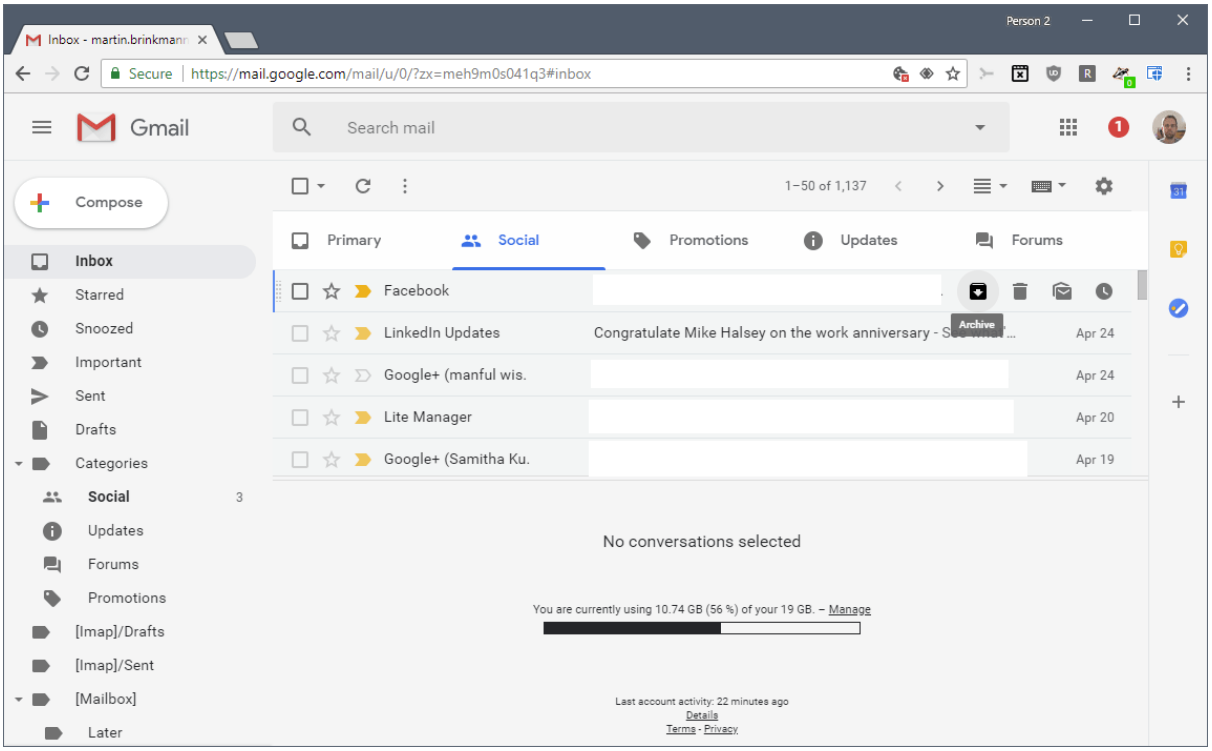
    • Mnemonic: Gmail

    • Specification number: -

    • Version number: -

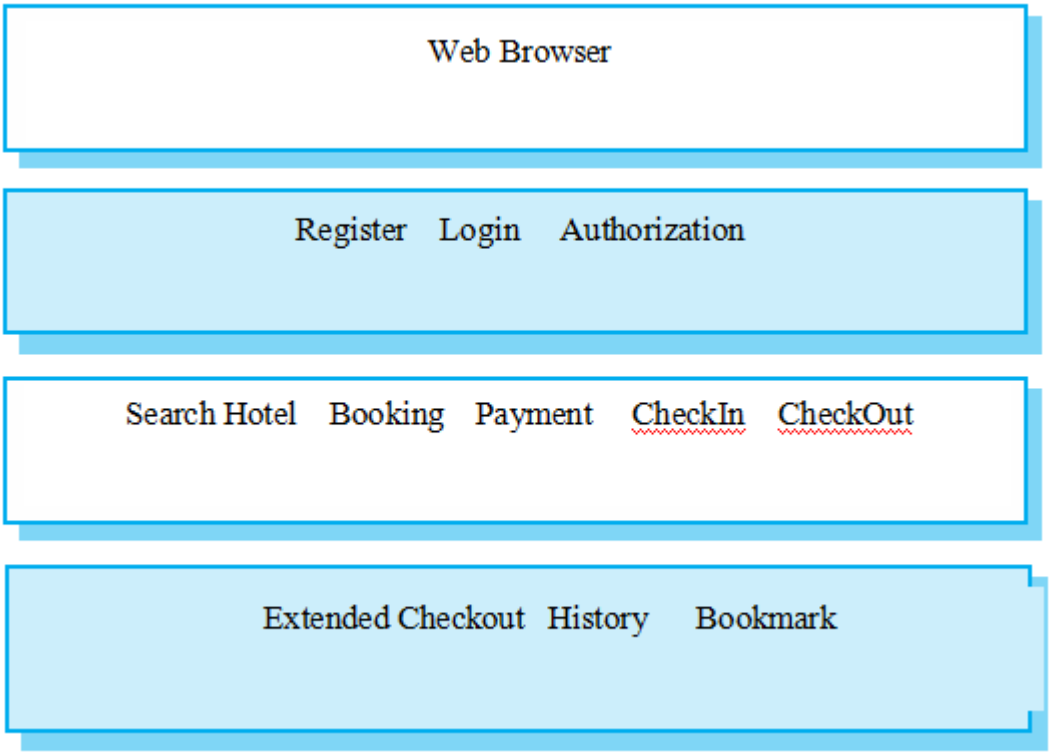
    • Source: Gmail.com

    • Purpose: For sending all the customer transactions and confirmation



2.1.4 **Communication Interfaces**

The communication architecture of this system follows the layered model. The system is organized in layers where all users need to start off by log on to the system. Each account needs to be authorized before accessing all other available functions in the system.



2.2 **System Features**

The system features include use case diagram, activity diagram, domain model and state machine diagram.

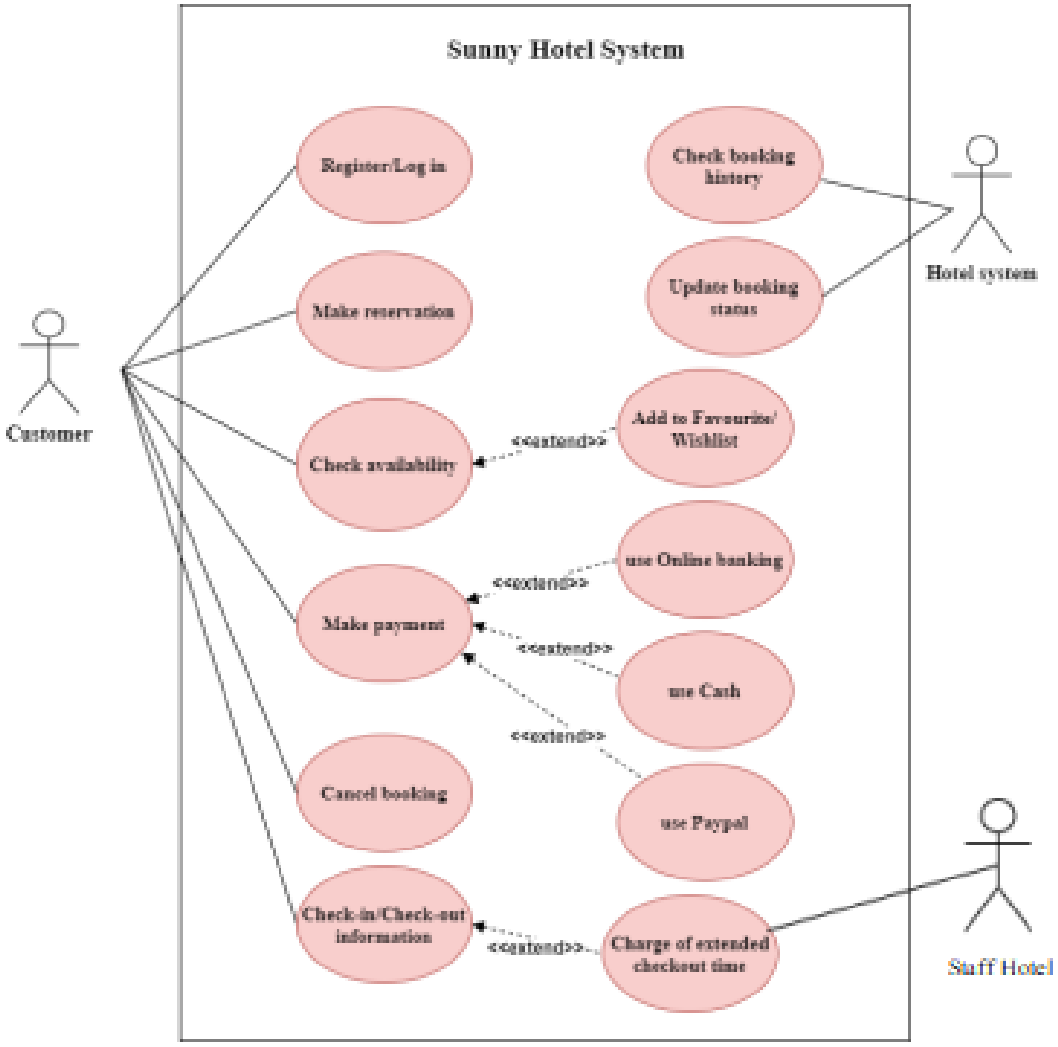


Figure 2.1: Use Case Diagram for Hotel Booking System (Najwa)

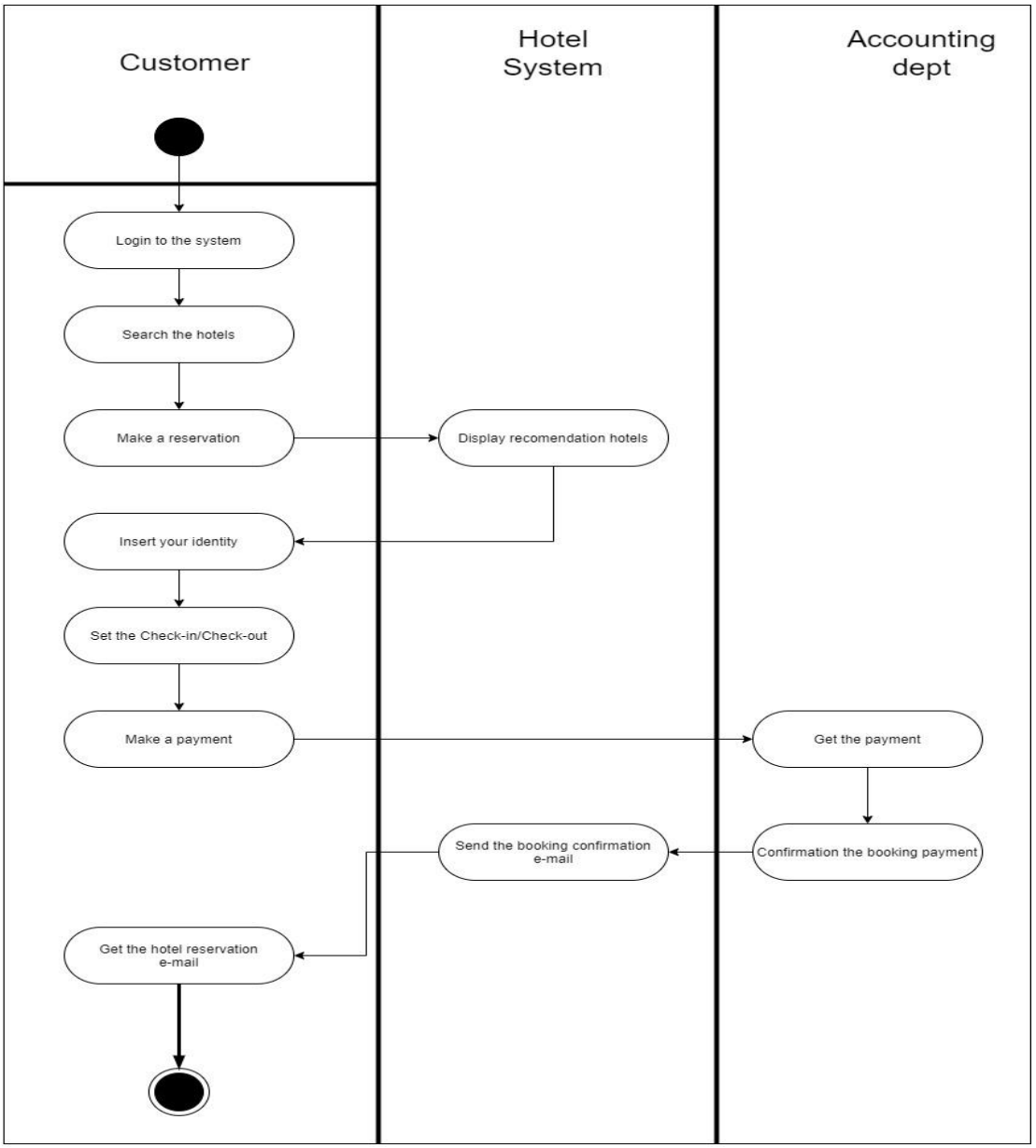


Figure 2.2: Activity Diagram for Sunny Hotel System (Ega)

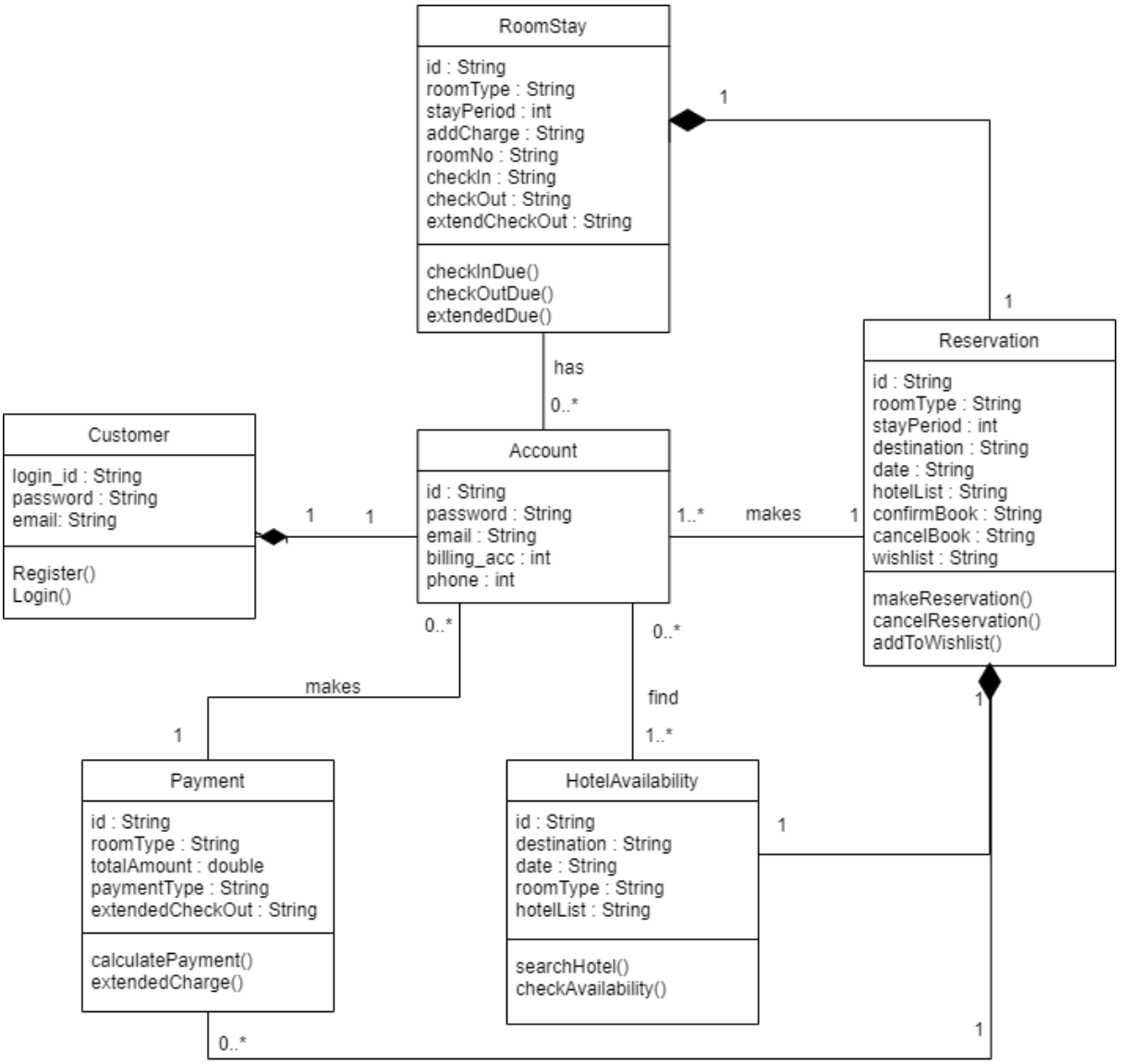


Figure 2.3: Domain Model for Sunny Hotel System

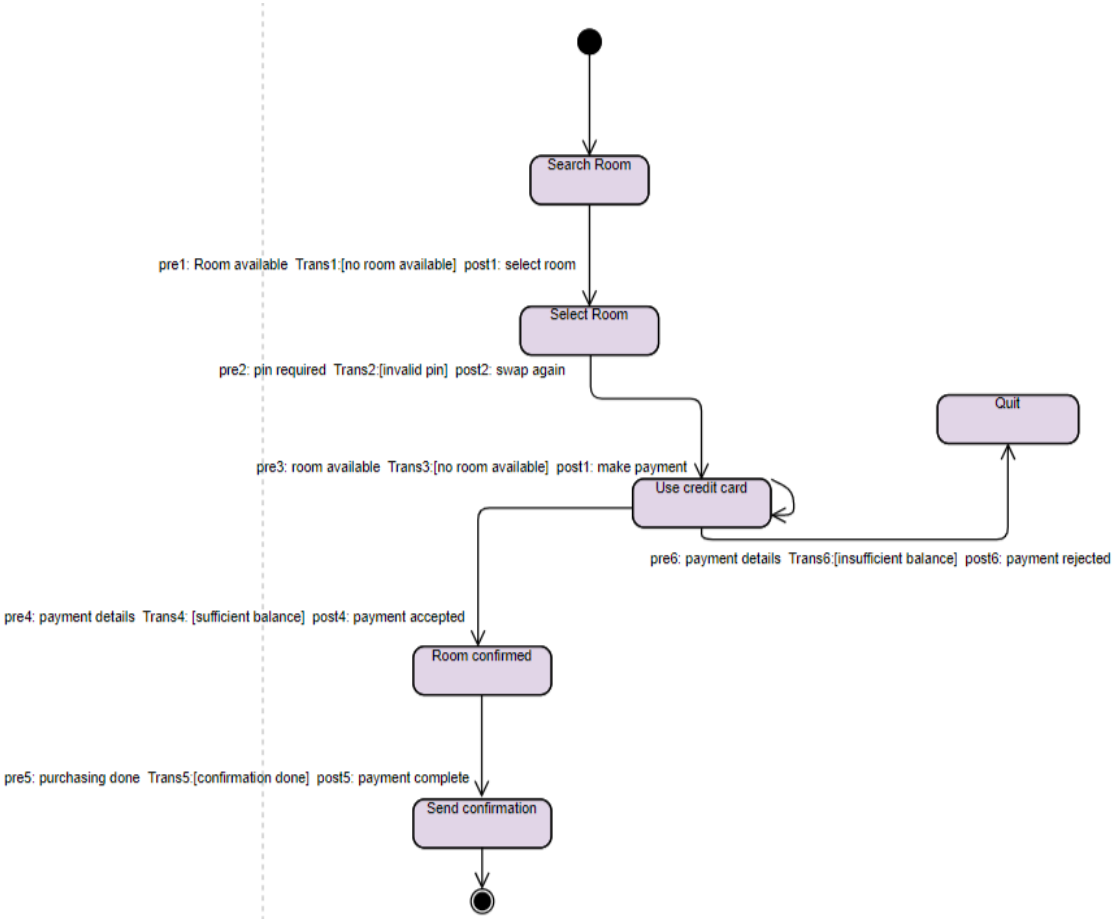


Figure 2.4: State Machine Diagram for Sunny Hotel System

2.2.1 **UC001: Use Case Register/Login (Azizah)**

Table 2.1: Use Case Description for Register/Login

Use case: Register/Login
ID : UC001
Actors :Customers, Hotel System
Preconditions: 1. The user's device must be connected to the internet
Flow of events: 1. The user open the website/application of Sunny Hotel 2. If the user is a first timer 2.1. Choose sign up 2.2. Insert username, password and email 2.3. Click register and the system will save the details 2.4. The system will authenticate the user through username and password 3. If the user already has an account 3.1. Choose login 3.2. Insert username and password 3.3. Click login 3.4. the system will check whether the username and password has been authorized or not 3.4.1. If authorized 3.4.1.1. Display the main page of Sunny Hotel 3.4.2. Else if the user has typo in the input 3.4.2.1. Notify the user insert the wrong username and/or password 3.4.2.2. Give option to renew password 3.4.3. else 3.4.3.1. Notify user the username insert is invalid 3.4.3.2. Give option to register
Postcondition : The user will be sent an email to verify their account

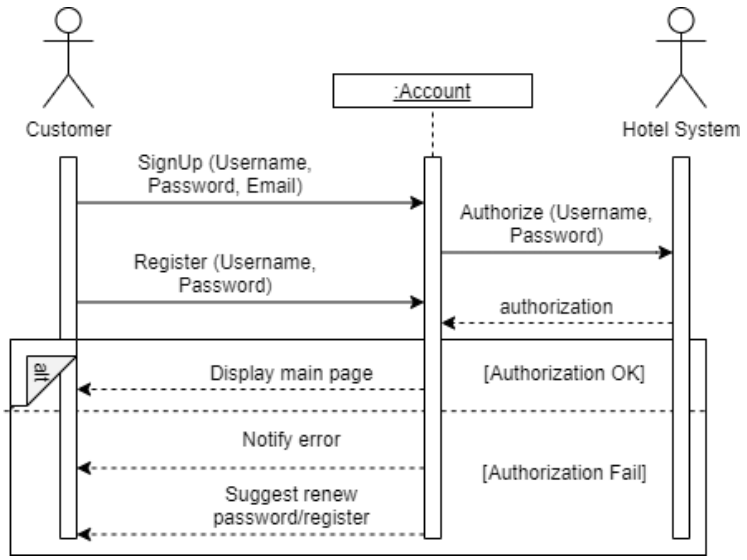


Figure 2.5 : Sequence Diagram for Register/Login

2.2.2 **UC002: Use Case Search hotel (Najwa)**

Table 2.2 : Use Case Description for Search Hotel

Use case: Search Hotel
ID: UC002
Actors: Customer
Preconditions: 1. The Customer is logged on the system
Flow of events: 1. The use case starts when the customer enters the destination or places in the search box. 2. The customer chooses the date and type of room. 3. The customer selects “search” to find the suitable available hotel. 4. The system displays the list of hotels available.
Postconditions: 1. The customer proceeds to choose the hotel.

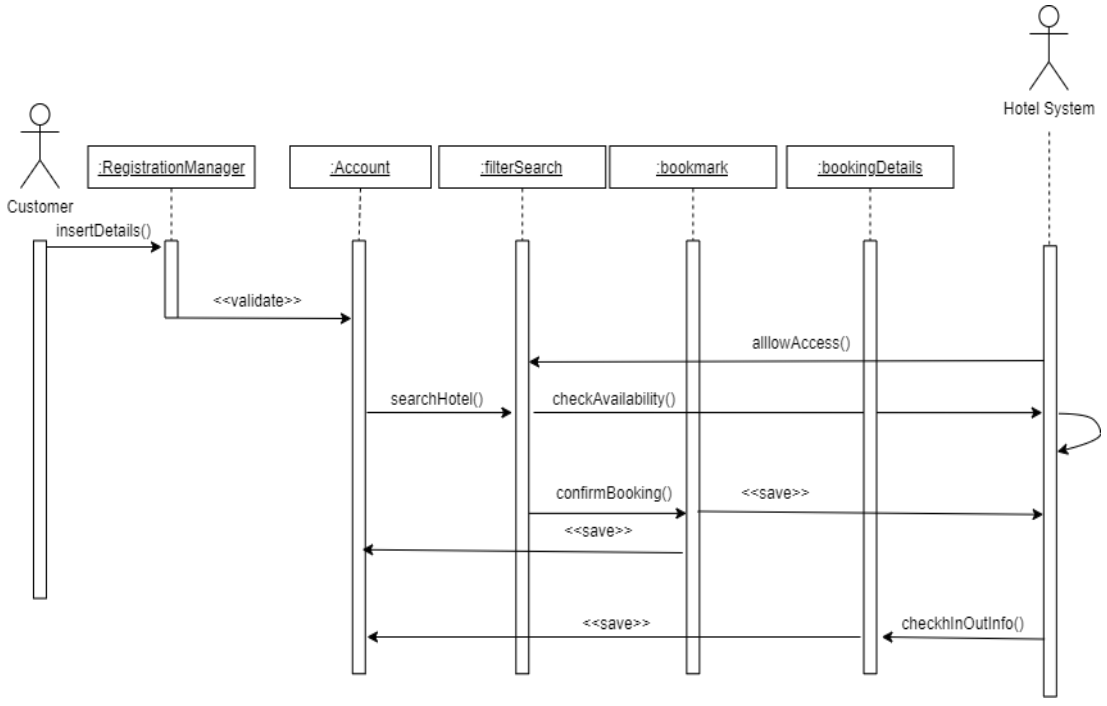


Figure 2.6 : Sequence Diagram for Search Hotel

2.2.3 UC003: Use Case Check availability (Najwa)

Table 2.3 : Use Case Description for Check availability

Use case: Check Availability
ID: UC003
Actor: Customer
Preconditions: The Customer is logged on the system
Flow of events: 1. The use case starts when the Customer select “Search hotel” after fill in the hotel details 2. If there are no hotels that meets the criteria 2.1 The system informs the customer that there are no hotels available 2.2 The use case terminates 3. The system displays a list of all the hotels that meet the criteria and similar to the details that customers filled in including the hotel’s name, type of room and price.
Alternative flow: 1. At any point the Customer may move to a different page.

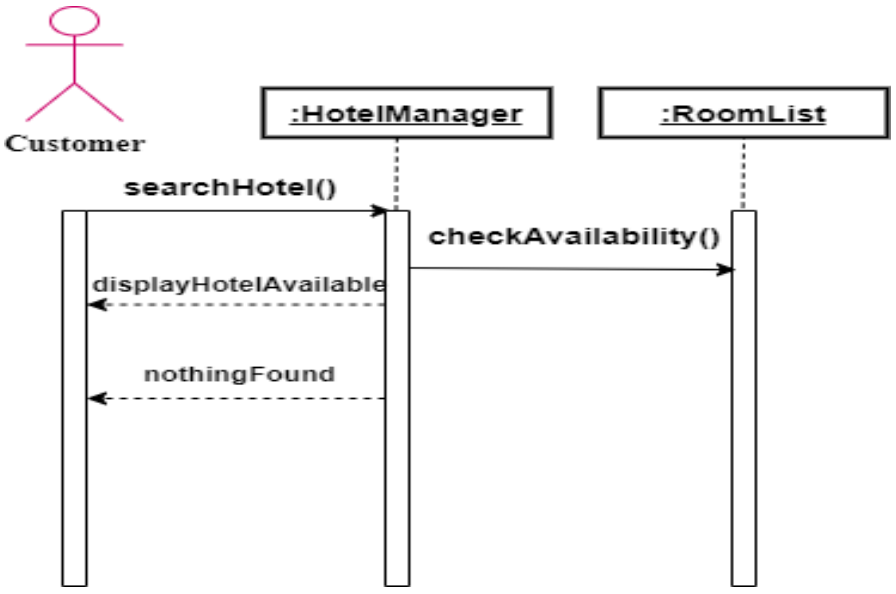


Figure 2.7 : Sequence Diagram for Check Availability

2.2.4 UC004: Use Case Check-in/Check-out information (Ega)

Table 2.4: Use Case Description for Check-in/check-out information

Use case : Check-in/check-out information
ID: UC004
Actors: Customer, Hotel System
Preconditions: 1. The customer is logged on the system
Flow of events: **1.** Customer can select the reservation to move to the hotel booked page 2. Customer can choose information about the hotel 3. The system will display the information about check-in/check-out the hotel you have booked
Postconditions: **1.** Customer will get the information about check-in/check-out

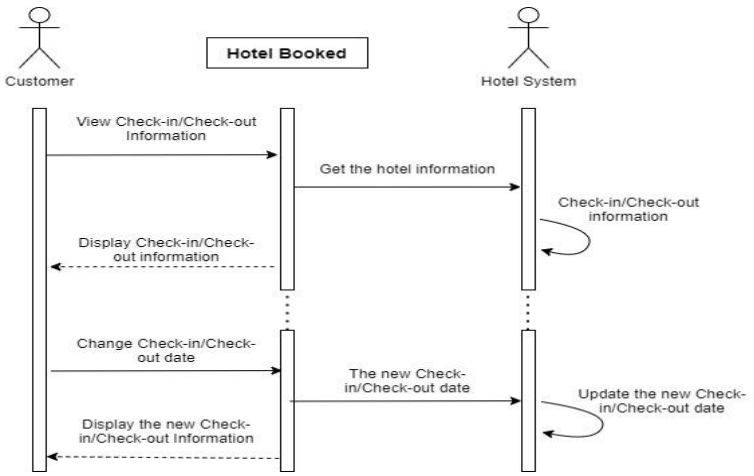


Figure 2.8 : Sequence Diagram for Check-in/check-out information

2.2.5 UC005: Use Case Charge of Extended Checkout Time (Azizah)

Table 2.5 : Use Case Description for Charge of Extended Checkout Time

Use case : Charge of Extended Checkout Time
ID : UC005
Actors : Customer, Hotel System
Preconditions : 1. The customer is logged on the system. 2. The customer has checked-in to the hotel booked.
Flow of events: 1. If the user request late check-out 1.1. 4 hours and above before check-out due 1.1.1. Approve request 1.1.2. Update system to add 2 hours after due 1.2. Less than 4 hours before check-out due 1.2.1. Apply charge 1.2.2. Approve request 2. Else 2.1. Apply charges and notify the customer
Postconditions : 1. The customer will be notify whether there will be additional charge or not

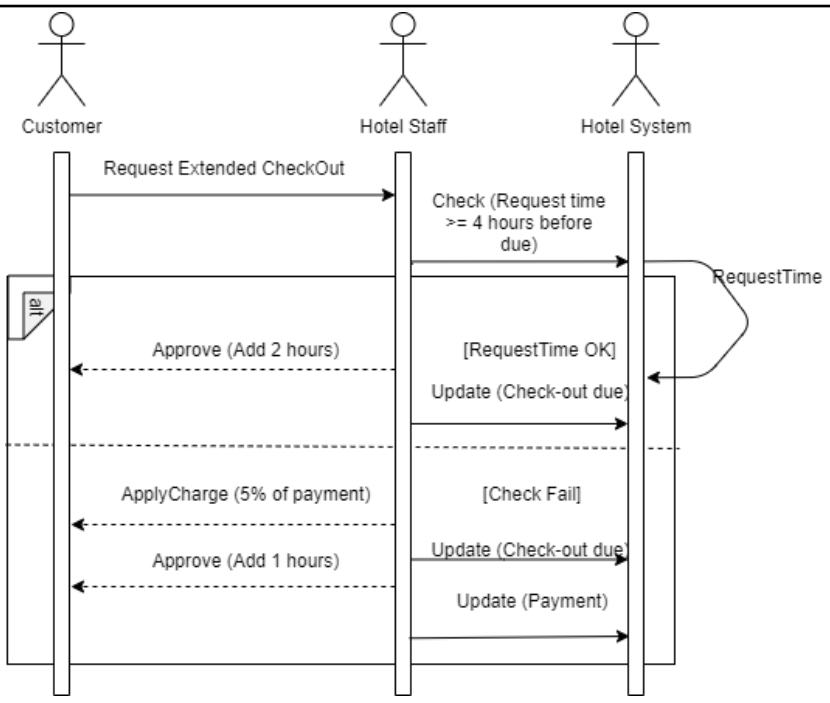


Figure 2.9 : Sequence Diagram for Check Availability

2.2.6 **UC006: Use Case Cancel Booking (Najwa)**

Table 2.6 : Use Case Description for Cancel Booking

Use case: Cancel Booking
ID: UC006
Actors: Customer
Preconditions: The Customer is logged on the system
Flow of events: 1. The use case starts when the customer selects “cancel booking”. 2. The system displays “Are you sure?” message. 3. If the customer selects “Yes” then 3.1 The system proceeds the booking cancellation process. 4. Else 4.1 The system displays the previous window.
Postconditions: The hotel booking is canceled.

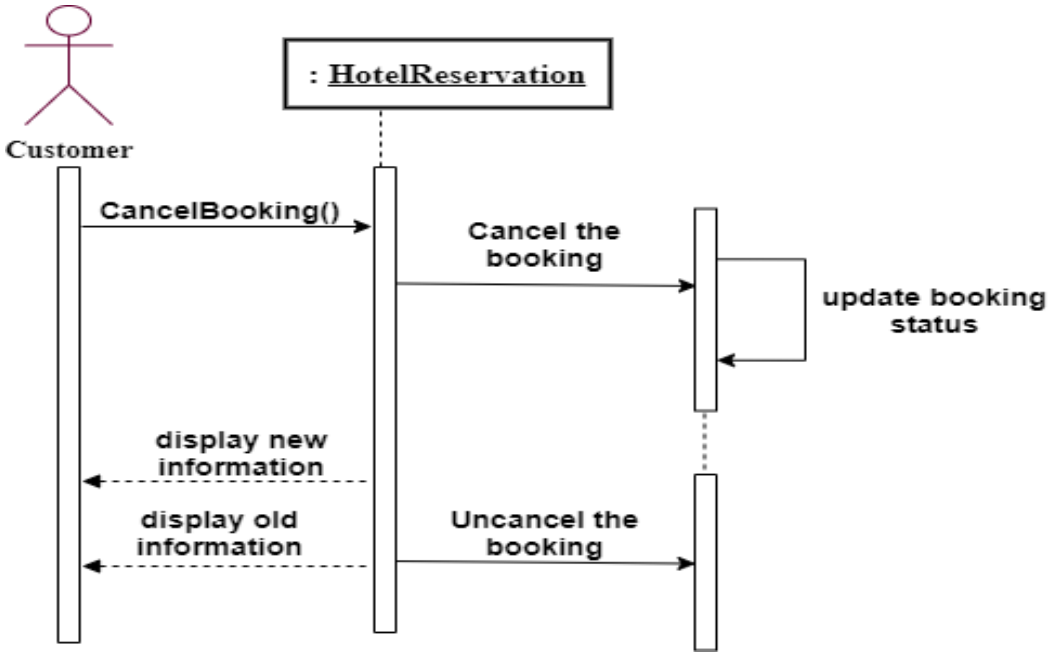


Figure 3.0: Sequence Diagram for Cancel Booking

2.2.7 **UC007: Use Case Make payment (Haziq)**

Table 2.7: Use Case Description for make payment

Use case: Make payment
ID: UC007
Actors: Customer
Preconditions: The customer chooses what room and time for staying
Flow of events: 1. Customer picks and chooses their desired room and staying time 2. After the customer gets the desired room, they can click the pay icon to pay for the room. 3. The hotel system will direct the customer to the payment link.
Postconditions: Customers can pay for the hotel room from the link directed to them.
Alternative flow n: Customer can cancel the payment anytime.

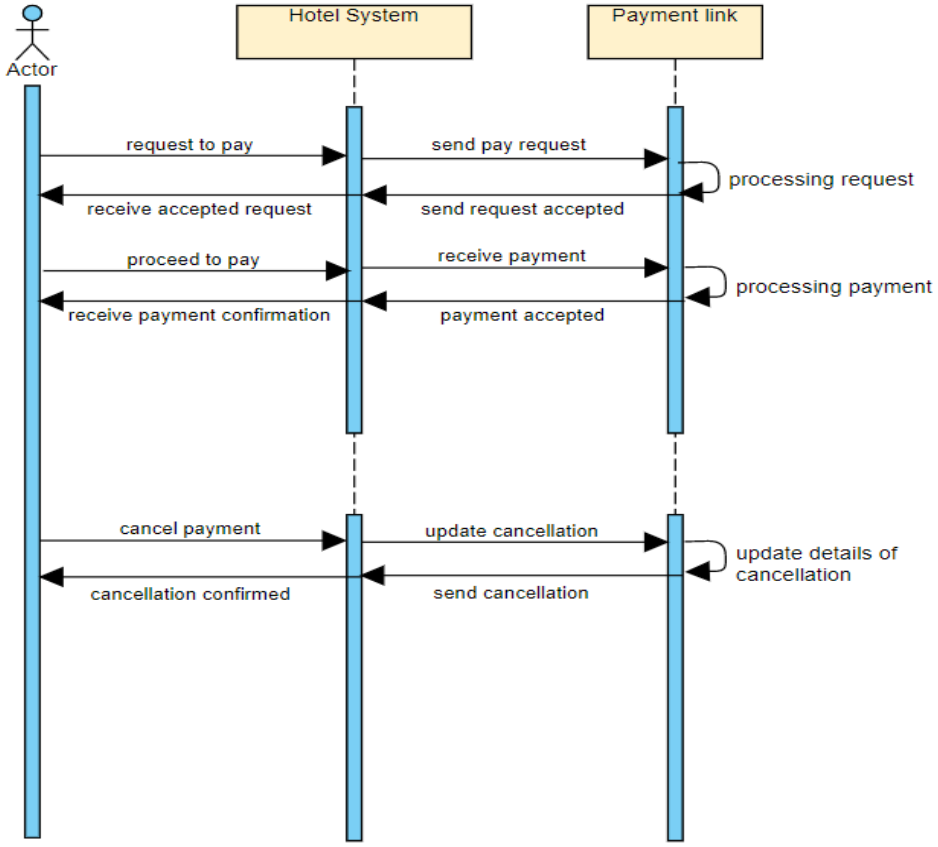


Figure 3.1 : Sequence Diagram for Make Payment

2.2.8 **UC008: Use Case Add to favorite/wishlist (Ega)**

Table 2.8: Use Case Description for Add to favorite/wishlist

Use case : Add to favorite/wishlist
ID: UC008
Actors: Customer
Preconditions: The customer is logged on the system
Flow of events: 1. Customer need to search hotels they interested 2. After customer get the hotel they interested, customer can click the love/cart icon on the page to add the hotel to the favorite/wishlist 3. The hotel that customer choose automatic added on the favorite/wishlist page
Postconditions: Customer will get a list of which hotels have been in favorite / wishlist
Alternative flow n: Customer can remove the list in the favorite/wishlist

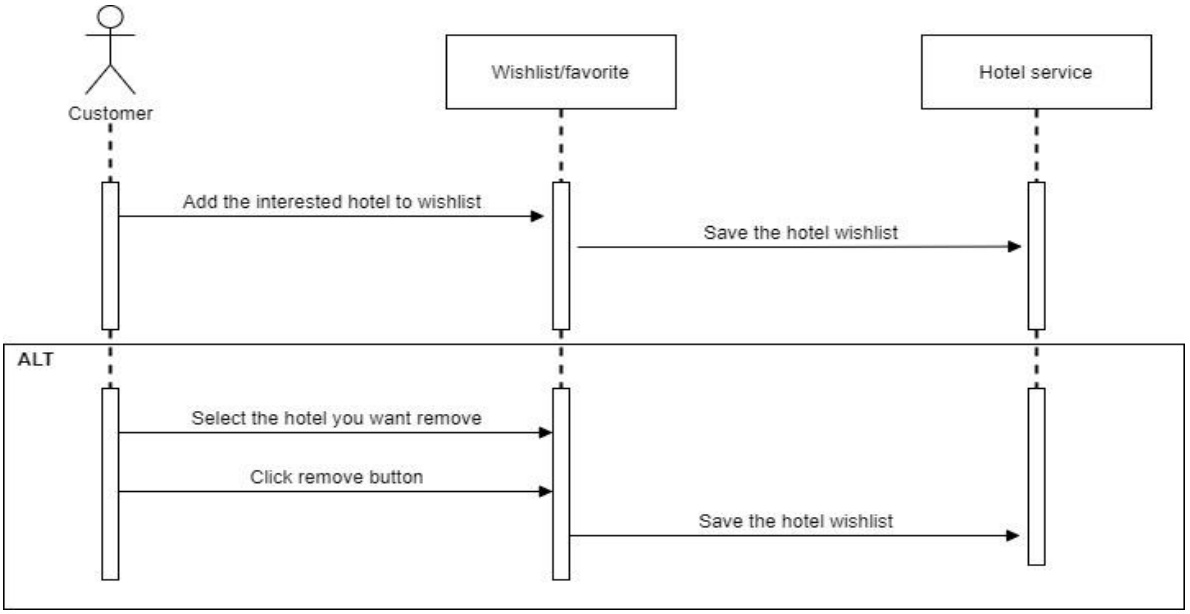


Figure 3.2 : Sequence Diagram for Add to Wishlist

2.2.9 UC009: Use Case Make Reservation (Haziq)

Table 2.9: Use Case Description for make reservation

Use case: Make reservation
ID: UC009
Actors: Customer, Hotel System
Preconditions: Customers want to make hotel reservation
Flow of events: 1. Customers need to choose what hotel and room to pick as well as the time of staying. 2. After deciding, the customer needs to click the reserve button. 3. Then, the system will book the reservation for the customer.
Postconditions: Customers can cancel the reservation anytime.

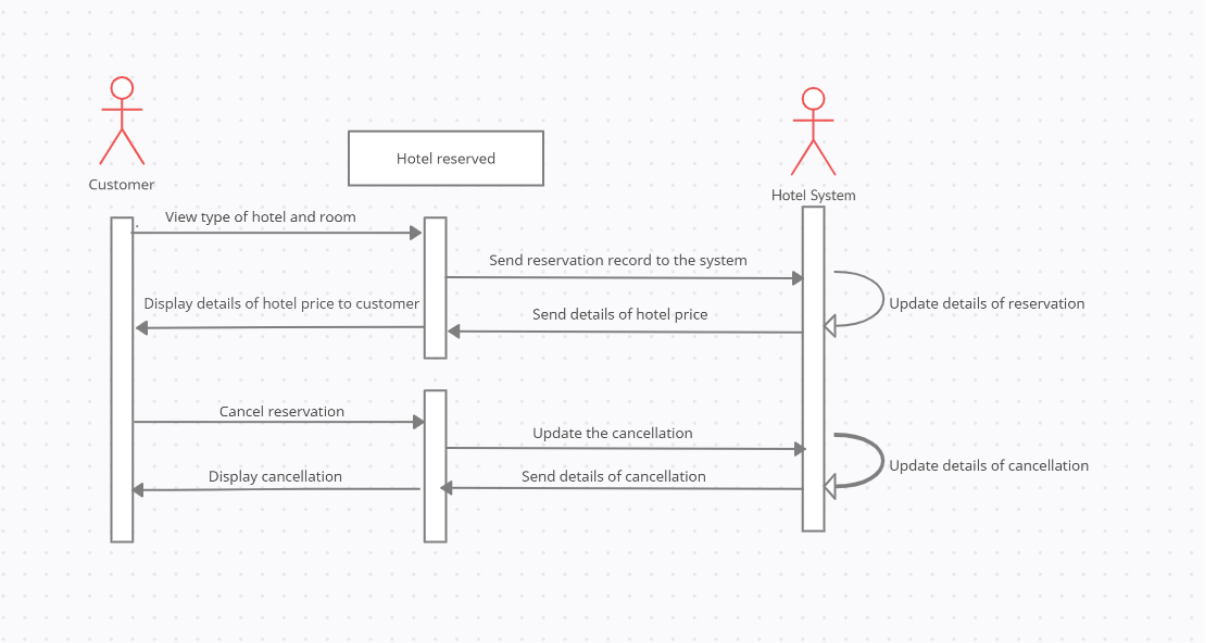


Figure 3.3 : Sequence Diagram for Make Reservation

2.2.10 **UC010: Use Case to Check History (Ega)**

Table 2.10: Use Case Description for Check history

Use case: Check history
ID: UC010
Actors: Customer
Preconditions: Customers have made several hotel reservations
Flow of events: 1. Customer need to move to the history page 2. After that, there are the list of customer booking history 3. Customer be able to remove/delete the hotel history by select the hotel and click remove from history
Postconditions: Customer get the list of booking history

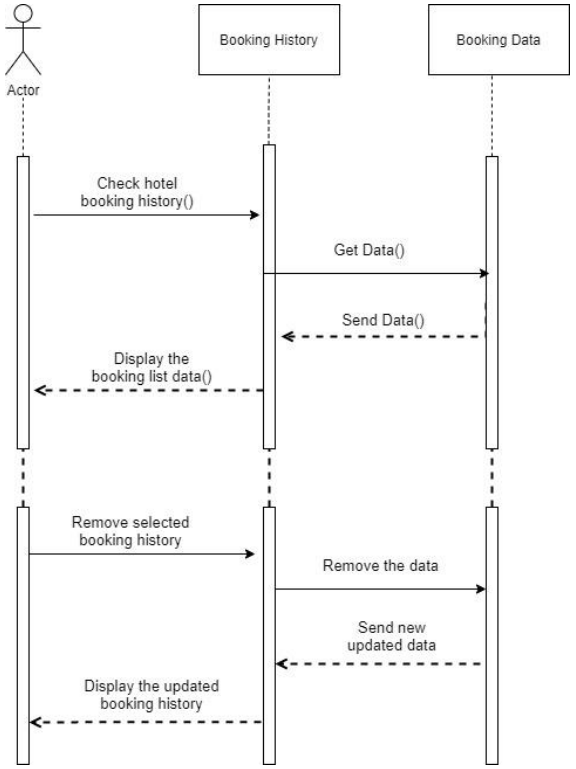


Figure 3.4 : Sequence Diagram for Check History

2.3 **Performance and Other Requirements**

Performance requirements define acceptable response time for system functionality:

- The load time for user interface screens shall not take longer than 3 seconds.
- When the customer logs in to the system, it should be verified in less than 5 seconds.
- Results should be returned by queries within 5 seconds.

Usability requirements define how usable the software in its intended purpose:

- The system must be compatible with both Android and iOS.
- When a customer is browsing the system, it must show which room is available for booking.

Maintainability requirements define the capability of being repaired or rectified:

- The Hotel Booking System is being developed in Java. Java is an object oriented programming language and is easy to maintain.
- If the system needs a new overhaul, it would not be a big problem because Java is a flexible programming language.

Adaptability requirements refer to how can it adjust to different condition:

- The system must be adaptable to different gadgets used.
- If the user is using the system on the desktop, the layout will look different than using a phone.

2.4 **Design Constraints**

- Time constraints

  We need a fully functional system to be delivered in two months.
- Functional Constraints

  The user need to login before using each function

  The system need to be able to do all the function in a correct order

  The system need not stay on the same page if the user wishes to go back to the previous page.
- Non-Functional Constraints

  The system must be able to operate 24 hours.

  The system must authenticate each user through the username and password.

  The system must verify each account by sending verification links to the email of the user after each registration.

2.5 **Software System Attributes**

- The system must be attractive and user-friendly for user's convenience.
- The system must be designed in a systematic view to avoid confusion towards users.
- The system must be efficient and compatible with all types of devices and software such as pc, mobile phone, android and apple.
- The system security must maintain integrity and authenticity of all users.