Product Requirements Document

Submission 1 Specification

NüRoom

SWEN90007 SM2 2021 Project

In charge:

Name	Student ID	UoM Username	GitHub Username	Email
Guo Xiang	1227317	XIAGUO2	moneynull	xiaguo2@student.unimelb.edu.au
Basrewan Irgio	1086150	IBASREWAN	irgiob	ibasrewan@student.unimelb.edu.au
Bai Zhongyi	1130055	zhongyib	Marvin3Benz	zhongyib@student.unimelb.edu.au
Hernán Romano	1025543	hromanocuro	hromanoc	hromanocuro@student.unimelb.edu.au

Release Tag: SWEN90007_2022_Part1_Nuroom

SCHOOL OF
COMPUTING &
INFORMATION
SYSTEMS

Revision History

Date	Version	Description	Author
29/07/2022	01.00-D1	Initial document	Hernán Romano
31/07/2022	01.00-D2	Updating Use Cases Table, adding an initial UML Diagram and Use Case 01 description	Hernán Romano
02/08/2022	01.00-D3	Adding Use Case 02 description	Hernán Romano
07/08/2022	01.00-D4	Updated Use Case diagram. Updated Use Case 01, 02, 03, 04.	Zhongyi Bai
07/08/2022	01.00-D5	Reviewed and commented on use case diagram and added descriptions for use cases 08, 09, and 10	Irgio Basrewan
07/08/2022	01.00-D6	Re-ordering Use Cases and Adding Use Case 01, 03 and 04	Hernán Romano
07/08/2022	01.00-D7	Updated Use Case diagram. Updated Use Cases 05, 06, 07, 11,14,15	Zhongyi Bai
10/08/2022	01.00-D8	Added description to use cases of admin part	Xiang Guo
12/08/2022	01.00-D9	Changed domain model to use PlantUML, reviewed and updated contents of domain model	Irgio Basrewan
14/08/2022	01.00-D10	Review and comments on Domain Models and overall report	Hernán Romano
14/08/2022	01.00-D11	Added description to domain model	Irgio Basrewan
15/08/2022	01.00-D12	Updated a few expression	Zhongyi Bai
15/08/2022	01.00	Version 1 & Create Release tag	Team

Contents

1. Introduction

1.1 Proposal

This document specifies the SWEN90007 project use cases, actors to be implemented, and the system's domain model.

1.2 Target Users

This document is mainly intended for SWEN90007 students and the teaching team.

1.3 Conventions, terms and abbreviations

This section explains the concept of some important terms that will be used throughout this document. These terms are detailed alphabetically in the following table.

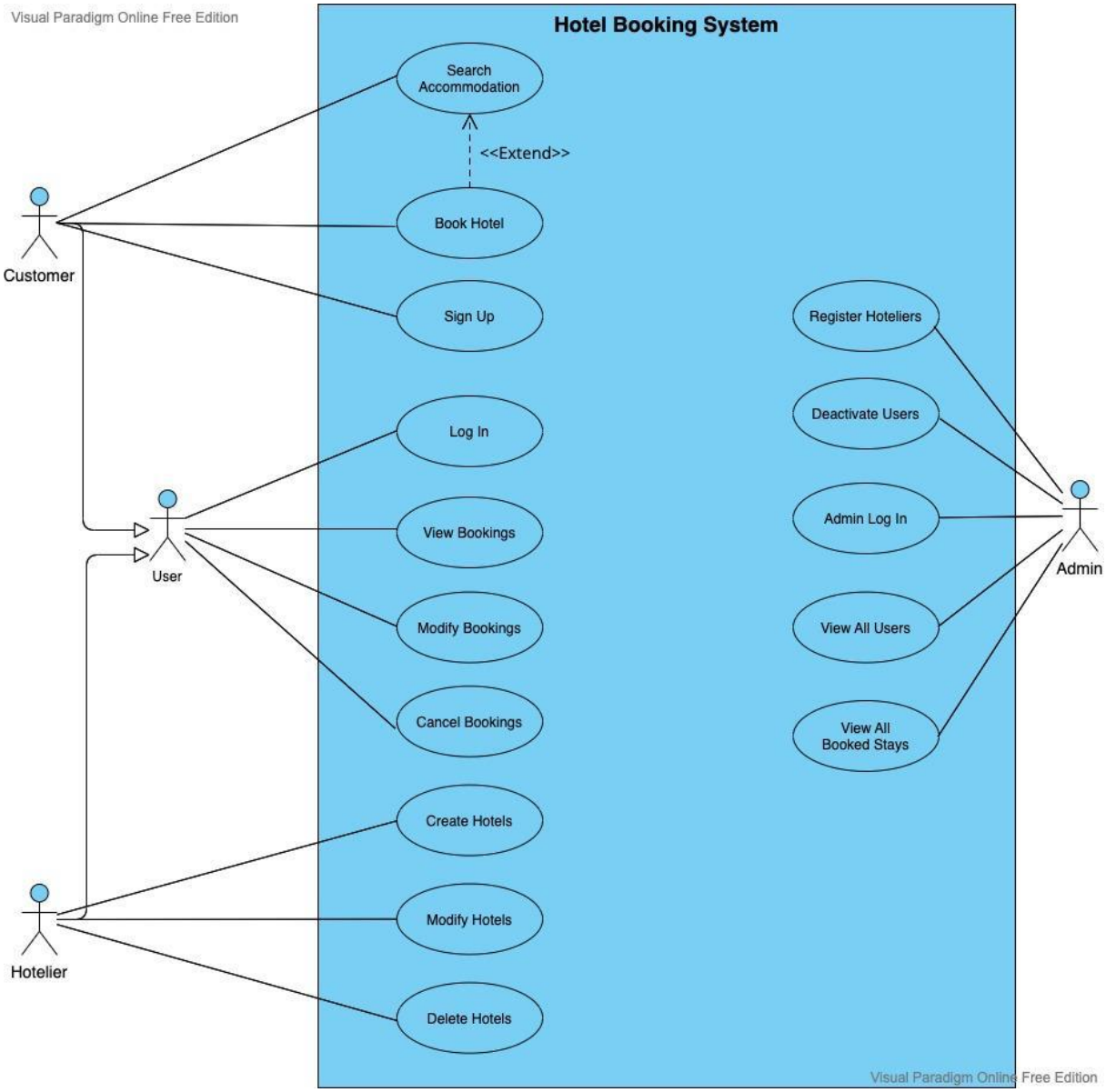
Term	Description
UML	Unified Modelling Language

2. Actors

Actor	Description
Customer	A person who makes use of the Hotel Booking System and acquires services
Administrator	A person who manages the Hotel Booking System
Hotelier	A person who manages hotel(s)

3. Use Cases

3.1 Use Case Diagram



3.2 List of Use Cases

The **table 01** shows a summary of the use cases considered in the project:

Table 01: List of Use Cases

Use Case ID	Use Case Name
01	Customer Signs Up
02	User Logs In
03	Customer search for accommodations
04	Customer books Hotel Rooms
05	User Views Bookings
06	User Modifies Bookings
07	User Cancels Bookings
08	Hotelier Creates Hotels
09	Hotelier Modifies Hotels
10	Hotelier Deletes Hotels
11	Admin Logs In
12	Admin Views All Users
13	Admin Views All Booked Stays
14	Admin Registers Hoteliers
15	Admin Deactivates Users

3.2.1 Use Case 01: Customer Signs Up

Actors

1. Customer

Basic Flow

A customer wants to view the bookings; however, she cannot process the booking because she needs to login into the platform. It is her first time using it, and she will need to create a new account. After completing a form with basic personal information, which the system validates and records, she creates her new account successfully. This person receives a confirmation of the newly created account. The system redirects her back to the login page.

3.2.2 Use Case 02: User Logs In

Actors

1. Customer
2. Hotelier

Basic Flow

On Monday, a user (customer or hotelier) wants to login into **NüRoom** to view his bookings. He fills out the form with his account and correct password on the Login page. After clicking the login button, he is authenticated to do further actions. (e.g. view bookings, cancel bookings, and modify bookings)

3.2.3 Use Case 03: Customer Searches for Accommodations

Actors

1. Customer

Basic Flow

A customer has an urgent work trip for the coming week. For this, she decides to search for the best accommodation deal close to downtown on her favourite Hotel Booking Platform: **NüRoom**

After successfully logging into the Hotel Booking Platform, the customer sees the landing page of the website. The customer visualizes a search bar and searches for accommodation by postcode, number of rooms, and amenities, and picks the booking dates. The system validates the data. The system retrieves the requested information. The customer visualizes a list of hotels with matched characteristics.

3.2.4 Use Case 04: Customer Books Hotel Rooms

Actors

1. Customer

Basic Flow

Days before Christmas, a customer plans to visit his parents with his spouse and children. Therefore he decides to login into his favourite online Hotel Booking platform: **NüRoom**. After logging in, he views the landing page and a search box. He types his parents' zip code number into the search box, picks the booking dates, and waits for the results. The system validates the data. After successful validation, the system retrieves the resulting Hotels. He selects one of the hotels from the list. The system retrieves details information about the selected Hotel, including accommodation availability. The customer selects two rooms, one for him and her spouse, and a bigger room for his children. The system validates the data and availability. After successful validation, the system redirects the customer to a confirmation page. He now has to review the detail of the reservation and confirm his order.

After confirming the order, the customer sees a success message of booking and returns back to the main landing page.

3.2.5 Use Case 05: User Views Bookings

Actors

1. Customer
2. Hotelier

Basic Flow

A user (customer or hotelier) has a hotel reservation, and he wants to double-check whether the start date is correct or not. After he logs in to the system, he is able to view all bookings he has. The bookings' information will be displayed, such as hotel name, location, number of rooms, number of beds, and others.

3.2.6 Use Case 06: User Modifies Bookings

Actors

1. Customer
2. Hotelier

Basic Flow

A user (customer or hotelier) has a hotel reservation and he wants to modify the start date. After he logs in to the system, he is able to view all bookings he has. The bookings' information will be displayed. Then, he selects to edit the start date. After he picks a suitable date and confirms to submit this modification. The user (customer or hotelier) will receive a notification that a booking modification needs to be approved. The user (hotelier or customer) will approve the modification and the start date will be updated.

3.2.7 Use Case 07: User Cancels Bookings

Actors

1. Customer
2. Hotelier

Basic Flow

A user (customer or hotelier) has a hotel reservation and wants to cancel a booking because his plan changes. After he logs in to the system, he is able to view all bookings he has. The bookings' information will be displayed. Then, he selects to cancel on booking and confirms that this booking will be cancelled. The user (hotelier or customer) will receive a notification that a booking has been cancelled.

3.2.8 Use Case 08: Hotelier Creates Hotel

Actors

1. Hotelier

Basic Flow

As a hotelier of a brand new hotel prepares for its grand opening, they decide to choose NüRoom as the platform that will host bookings for rooms in their hotel. After logging into the platform, a hotelier

follows the directions in setting up a new hotel on the site. The hotelier inputs information about the general hotel, such as its name, address, and available amenities. Next, the hotelier sets up the multiple room types (that vary in terms of the number of rooms, size, etc.) that are available to stay in at the hotel and denotes the number of rooms of each type that are available to be booked. After submitting the details of the hotel, the hotel is now available to be booked by customer users on the site and the hotelier is able to view the bookings made by customers.

3.2.9 Use Case 09: Hotelier Modifies Hotel

Actors

1. Hotelier

Basic Flow

After a dramatic increase in customers due to listing their hotel on NüRoom, the hotelier decides to buy the land surrounding the hotel to expand the hotel and the number of rooms available. After construction is complete in the hotel's new wings, the hotelier updates the hotel listing on NüRoom by adding new room types and increasing the number of rooms available for existing room types (that were replicated in the new wings). After submitting the update to the hotel, customers are able to make bookings to the new rooms available at the hotel, and see the updated details of the hotel on the site while browsing and booking.

3.2.10 Use Case 10: Hotelier Deletes Hotel

Actors

1. Hotelier

Basic Flow

As the hotelier sees increasing success in their hotel, they decide to try making their own booking website specifically for their hotel, rather than using NüRoom. To make sure customers use their new specialized booking system, the hotelier decides to remove their hotel from NüRoom. The hotelier goes to the settings of their hotel page to search for the delete hotel button. Once they've clicked it and have done some verification to ensure they truly want to delete the hotel, the hotel becomes unavailable on the website, and customers of NüRoom can no longer make new bookings for the hotel on the site. Existing bookings will be honored until there are none left.

3.2.11 Use Case 11: Admin Logs In

Actors

1. Administrator

Basic Flow

On Monday, an administrator wants to log into the system via the admin portal to deactivate a user, because this hotelier uploaded an image against the policy. He fills out the form with his account and correct password on the admin login page. After clicking the login button, he is authenticated to do further actions. (e.g. deactivate users, view all users)

3.2.12 Use Case 12: Admin Views All Users

Actors

1. Administrator

Basic Flow

For the regular report of the current usage of the system, the administrator would like to have an overall view of how many users and hoteliers are registered in the system. When an administrator wants to check the profile and information of specific users and specific hoteliers, they also have access to view all the details of users to guarantee all the information provided by users is legal and accurate.

3.2.13 Use Case 13: Admin Views All Booked Stays

Actors

1. Administrator

Basic Flow

The administrator might want to make an analysis of the popularity of different types of rooms and different location hotels, and calculate the average number of daily booked stays. So, it is necessary for the administrator to monitor the vacancy and status of all hotels, and they can view all stays that customers have booked.

3.2.14 Use Case 14: Admin Registers Hoteliers

Actors

1. Administrator

Basic Flow

When new hotelier wants to join the platform to list their hotels, they can ask the administrator to register hotelier accounts for them, and the hoteliers can log into the application and create new hotels. Thus, the hoteliers cannot register by themselves, only the administrator can register hoteliers in the system.

3.2.15 Use Case 15: Admin Deactivates Users

Actors

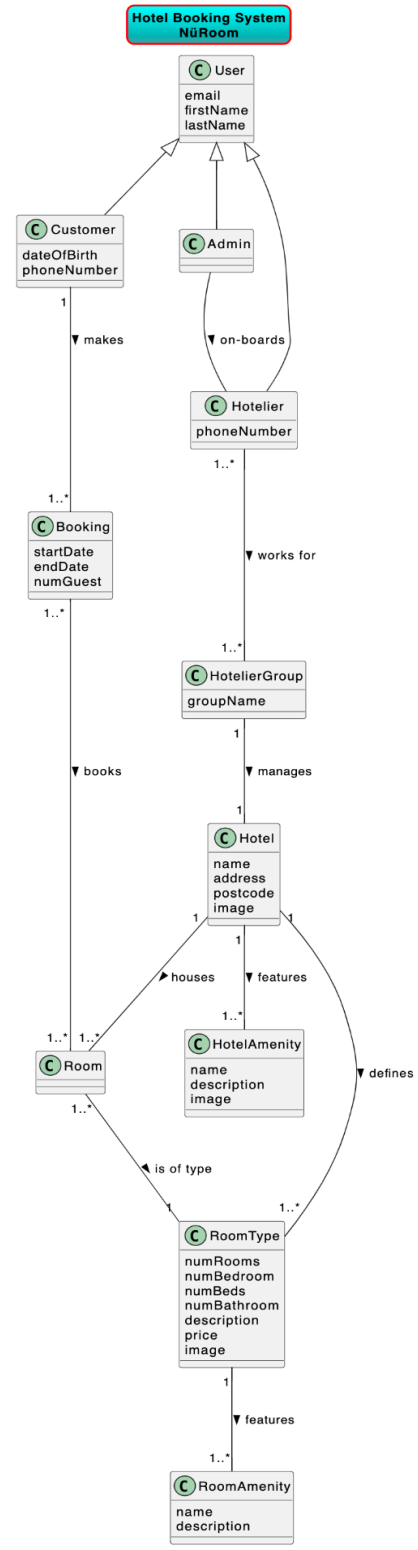
1. Administrator

Basic Flow

A hotelier has uploaded an image against the policy, so the administrator needs to deactivate this hotelier to prevent this user from uploading more improper images. After logging into the system via the admin portal. The administrator finds this hotelier's information and clicks deactivate button to make this user inactive. Then, that hotelier cannot log into the system anymore and cannot do any further actions (e.g. create hotels, view bookings).

4. Domain Model

4.1 Domain Model Diagram



4.2 Domain Model Description

The domain model for the NüRoom Hotel Booking System is structured surrounding the main 3 **user** types and their relationships with the different entities of the system. The first **user**, the **admin**, serves as the entity that **manages** the application as a whole and **on-boards** vetted **hoteliers**, the second **user** type onto the system. These **hoteliers** are responsible for managing their **hotels** through **hotel groups** (with each **group** comprising the **hoteliers managing** said **hotel**). Each **hotel defines** its **rooms** through a set of **room types**, each varying in size, **amenities**, price, etc. The **hotel** also defines its own set of **amenities** that its **hotel features**. Finally, there is the third **user** type of **customer**. The **customer** is responsible for making **bookings**, which connect the **customer** with the **room**(s) (and therefore also the **hotel**) that they wish to book.

In summary, the specifications of the hotel booking system dictate the following business requirements:

- **Users** *can be either* an **Administrator**, **Customer**, or **Hotelier**;
- **Administrators** add **Hoteliers** to the system and can view all **Bookings** and **Users;**
- **Hoteliers** can manage *one or more* **Hotels** through **HotelGroups**;
- Each **HotelGroup** manages *one* **Hotel**, and each **Hotel** has *one or more* rooms that can be of *one of multiple* **RoomTypes;**
- **Hotels** can feature *one or more* **HotelAmenity**(s) and **Rooms** can feature *one or more* **RoomAmenity**(s);
- **Customers** can make *one or more* **Bookings** for *one or more* **Rooms;**
- **Rooms** can be in *one or more* **Bookings** at different points in time;

Entities have been bolded; attributes have been underlined; and important *associations* have been italicized.

5. References

[1] Larman, C. (2005). *Applying UML Patterns: an introduction to object-oriented analysis and design and iterative development (3rd ed.).* Pearson Education, Inc.

[2] Cockburn, A. (2016). *Writing effective use cases.* Boston Addison-Wesley

[3] PlantUML. *Domain Model Tool Creator*. https://plantuml.com/commons

[4] Visual Paradigm Online. https://online.visual-paradigm.com/diagrams/solutions/free-visual-paradigm-online/