N° d'ordre :
N° de série :

People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research

EL-OUED UNIVERSITY

FACULTY OF EXACT SCIENCES

End of study memoir

ACADEMIC BACHELOR'S DEGREE

Domaine: Mathematics and Computer Science

Sector: Computer science

Specialty: Fundamental computer science

Presented by: DJABOUB Amine, MEFTAH Mohammed Bachir and CHENGUEL Mohammed Laid

Theme

An android application for booking hotel rooms

supported the
Before the jury composed of:

President

Examiner

Rapporteur : Dr.Maamra Oum Elhana

College year 2023 - 2024

ملخص

يعد مجال حجز الفنادق جزءًا كبيرًا من عالمنا الحديث، والتي سنستفيد كثيرًا من تنفيذها في البرامج الموجوده في أجهزتنا المحمولة

سيركز هذا المشروع على شرح خطوات دراسة بيئة الحجز الفندقي وتصميم وتطوير نظام يسمح بتحقيق هذا الهدف

نتيجة هذا المشروع هي نظام تطبيقات الهاتف المحمول الذي يسمح للعملاء بإكمال عملية حجز غرفة الفندق عبر الإنترنت

Résumé

L'industrie de la réservation hotele représente une part importante de notre monde moderne, et elle en bénéficierait grandement d'être implémenté dans le logiciel de nos appareils mobiles.

Ce projet se concentrera sur la démonstration des étapes d'étude de l'environnement de réservation hôtelière, de conception et de développement d'un système permettant d'atteindre cet objectif.

Le résultat de ce projet est un système sous forme d'application mobile qui permet à invités d'effectuer le processus de réservation d'une chambre d'hôtel en ligne.

Abstract

The hotel reservation industry is a big part of our modern world, that would benefit greatly from being implemented into software in our mobile devices.

This project will focus on showing the steps of studying the environment of hotel reservations and designing and developing a system that allows for that goal to be achieved.

The result of this project is a system in a mobile app form that allows guests to perform the booking process of a hotel room online.

Liste of Figures

Table of contents

Introduction

The hotel reservation process has evolved significantly with digital advancements, yet many systems still face issues such as inefficiencies, booking errors, and user dissatisfaction. Our project aims to address these challenges by developing a more robust and user-friendly hotel booking application.

The primary problem prompting this project is the inadequacy of current systems to provide a seamless and efficient booking experience. Our project aims to simplify the booking process, minimize errors, improve data management, and enhance user satisfaction and operational efficiency.

The first chapter examines the various workstations involved in hotel operations, from front desk to backend administration. We map out the data flow within the system and outline procedures for gathering the necessary information to build an effective system. Identifying current problems and setting project objectives form the foundation for development.

The second chapter discusses the methodologies used to design the hotel reservation system. We present diagrams and conceptual models to illustrate the system architecture, user interfaces, and data interactions. These visual tools provide a clear roadmap from design to implementation.

In the last chapter, we start this chapter by exploring the ergonomics and user interface design critical to our application's success. We then detail the programming languages and technologies chosen for their benefits. The chapter also covers database implementation, focusing on data integrity, security, and access efficiency. Finally, we provide an overview of the most common user interfaces to demonstrate how they enhance user interactions.

By addressing these aspects, our project aims to resolve prevalent issues in hotel reservation systems and set a new standard for user-friendly and efficient booking applications.

Finally, we will end this project with a general conclusion.

Chapter 1

STUDY OF THE EXISTING

1.1 Introduction

This study aims to design and implement an Android application for managing travellers' hotel reservations.
In this chapter, we will detail the different stages of the application to be studied, among others: The workstations, data flow organization in the system, and the work procedures to gather the information necessary to create the system. Ultimately, we determine this system's problems and the project's objectives.

1.2 The hotel reservation industry

The constant evolution of the tourism industry, which is driven by the emergence of digital technologies and its accelerated global access, is a result of one key element: hotel booking. Today, travellers worldwide rely heavily on online platforms to find, compare and book their accommodation, making this process an essential part of the travel experience.

1.2.1 Definition of hotel reservation

Hotel reservation is right now the most popular means of finding places to stay when travelling for work, vacation or other reasons, due to its ease, simplicity and convenience.

1.2.2 History of hotel reservation

The hotel booking industry has gone through a lot of evolution throughout the decades, from its start in the mid-17Th century to this day.
Initially, travellers turned to inns for accommodation, often making reservations in person or through written correspondence. With the advent of travel agencies in the 19Th century, hotel reservations became more organized and accessible to more people. Travel agents acted as intermediaries between hotels and customers, facilitating the booking process.
The arrival of the Internet in the 1990s revolutionized the way hotel reservations were made, the first hotel booking sites were launched in 1996, allowing customers to compare prices, read

reviews and book online. Platforms like Booking and Expedia have emerged, offering a wide range of accommodations and many options mobile apps have also made it easier to book hotels on the go, providing an easy and convenient user experience.
Currently, hotel apps are the main tool people use when reserving anywhere. Hotel booking apps are used to search for hotels and book accommodation online. These applications offer various services and features that make it easier for users to search and book.
Mobile applications allow users to search and book hotels anytime and anywhere with minimal time and effort, while also being capable of sending real-time notifications about special offers, promotions and travel reminders, helping users stay informed and not miss exciting opportunities.[1]

1.2.3 The importance of hotel reservation

Hotels play a significant role in the travel and hospitality industry, providing temporary accommodation, dining, and other services for travellers and tourists.
They offer a place for people to stay away from home, whether for business or leisure and contribute to the economy by creating jobs and generating revenue for local communities. Hotels also serve as social and cultural hubs, often showcasing local art, cuisine, and customs, while also providing meeting spaces for events and conferences. Overall, hotels facilitate travel, foster cultural exchange, and support local economies. [2]

1.2.4 Challenges of hotel reservation

As widespread and commonly used as it is, the hotel reservation process isn't flawless, guests and hotels usually run into problems during the hotel reservation procedure, these problems are:

1. Limited Availability: The desired dates or room type may not be available, especially during peak seasons or events.

2. Unclear Pricing: Hidden fees, taxes, or unclear pricing structures can confuse guests and lead to unexpected costs.

3. Security Concerns: Providing personal and payment information online can raise security concerns for some guests, especially if unfamiliar with the booking platform.

4. Language Barriers: For international travellers, language barriers may complicate the reservation process, leading to misunderstandings or difficulties in communication.

5. Addressing these issues through transparent pricing, clear communication, flexible policies, and responsive customer service can help hotels enhance the booking experience for their guests.

1.3 Data flow organization in the system

The classic hotel reservation process typically involves several steps initiated by the guest, often triggered by travel plans, events, or business purposes. Each actor (Guest and Hotel) performs multiple steps before their role in the system is satisfied. Here's a data flow diagram (flowchart) that helps understanding the process:

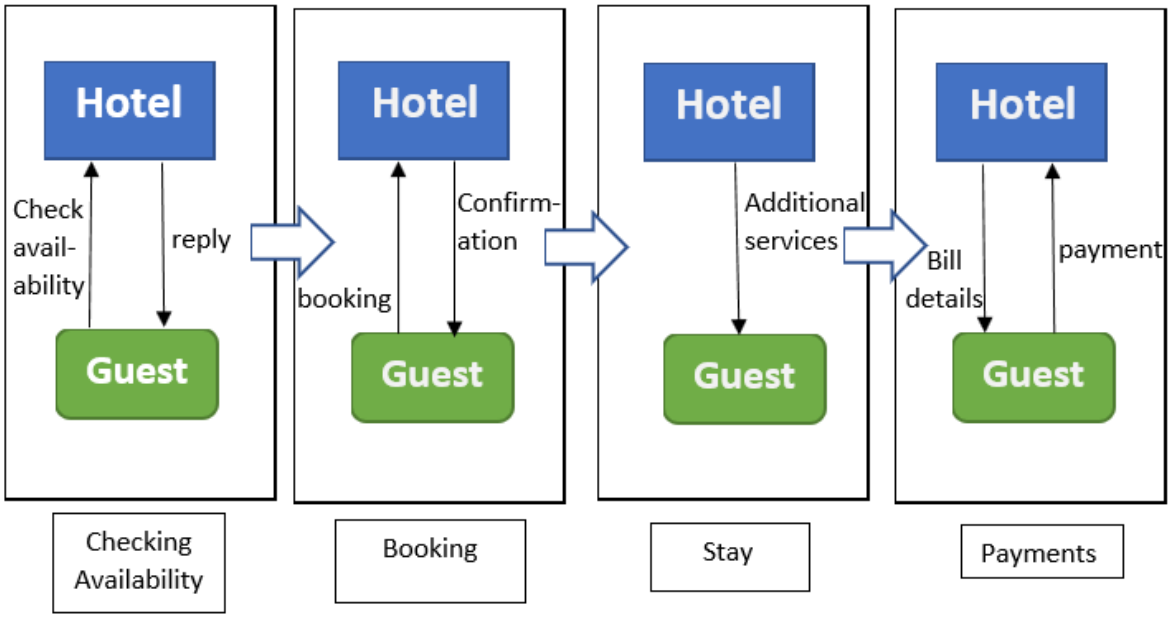

Figure 1.1: a graph clarifying the data flow Process

1.4 Works post study

The workstation is a center of activity with limited responsibility allowing the execution of a set of tasks. It can be a legal or natural person.
The objective of the study of workstations is to identify and describe the operations carried out and to observe the flow of information in the systems. The workstations involved in this system are:

1.4.1 Guest

- Search for rooms.
- Request a room.
- Confirming the request.
- Payment of the reservation price.

1.4.2 Hotel administrator

- Search for rooms.
- room status control.
- Save reservations.

1.5 Work procedures study

The study of work procedures will allow us to better understand the flow of information in the current system thanks to the flow diagram.

1.5.1 Room booking

Once the guest selects a hotel, they initiate the booking process. The guest provides personal details (name, contact information) and reservation specifics (dates, room type, special requests). Then the guest confirms the reservation and finally makes the payment to receive the reservation receipt.

The intervening actors

- Guest
- Hotel administrator

Realised Tasks

1. Request: the guest requests a room according to their preference.
2. Confirmation: the guest confirms the reservation of the room by choosing a room.
3. Register the reservation: the hotel administrator saves the guest reservation.

1.5.2 Room Management

The Administration of the Hotel also plays a role in attracting guests and guiding them through the reservation process, by showing them available rooms they might like.

The intervening actors:

- Hotel administrator

Realized tasks:

1. **Showing rooms based on guest preferences:** depending on the guest's search, the system shows rooms according to their preference.
2. **Providing room availability status:** showing if a room is available, and informing the guest when a reserved room will be available.
3. **Room control:** updating each room's status after a reservation and after a guest's stay.

1.6 Problem position

Despite the benefits of hotel booking, it faces some problems and challenges that must be addressed to ensure its efficient operation and better meet users' needs. Here are some suggested problems for this program.

1.6.1 problematic

1. **Availability of offers:** Users may face difficulties finding suitable offers, especially during peak periods or tourist seasons, affecting their ability to book at suitable prices.
2. **Providing inaccurate information:** Some reservations may be subject to cancellation or change due to providing inaccurate or out-of-date information about the hotels or services provided.
3. **Security and Privacy:** Security and privacy issues remain a concern for users, especially when entering their personal and financial information online, and thus providing a secure environment for payments and transactions is an important challenge.
4. **Communication and Customer Service:** Reservation software may face challenges in providing efficient and responsive customer service to resolve any issues or queries users may have during or after the booking process.
5. **Diversity and mobility:** Users may suffer from a lack of variety in the options available, especially in terms of destinations or additional services offered, which makes them look for other alternatives that may be lower in quality or safety.

1.6.2 Objectives of study

In this field, there are many goals that programmers seek to develop at present, including:

1. Working to develop and improve its operations and provide effective solutions to meet users' needs and ensure a smooth and reliable booking experience.
2. **Facilitating the reservation process:** The program aims to provide an easy-to-use interface that allows users to book rooms quickly and easily without problems.
3. **Providing comprehensive information:** The program seeks to provide comprehensive information about available hotels, the rooms available in them, and the services provided, with the possibility of updating this information regularly.
4. **Improving customer experience:** The program contributes to improving customer experience by providing various options for reservations and meeting their needs effectively.
5. **Improve hotel management:** The software can help improve hotel management by providing detailed reports on reservations, occupancy and revenue, facilitating strategic decision-making.
6. **Save time and effort:** The program reduces the need for direct human interaction in reservation processes, saving time and effort for both customers and hotel staff.
7. **Optimize Resource Distribution:** The software helps improve resource distribution within the hotel by providing accurate information about occupancy and forecasts of future demand.

1.7 Conclusion

In This chapter, we have discussed proposing solutions to problems that users may face during the booking process and also working to improve the customer experience of the application. In the next chapter, we will explore the charts and diagrams that help clarify the organization of our hotel reservation system and analyze the data related to it, which helps in better understanding the performance of the program and continuously improving it.

Chapter 2

DESIGN

2.1 Introduction

Design and conception make the software understandable, adaptable, reliable and reusable. Software design translates the user's requirements into a 'blueprint' for building the software. It is the link between problem space and solution space.
In this chapter, we will go over the methods used for designing the system we're aiming to develop and show the resulting diagrams and concepts needed to proceed to the next step, implementation.

2.2 Two Tracks Unified Process (2TUP)

2.2.1 Definition

2TUP is a software development process that implements the Unified Process; it proposes a Y-shaped development cycle, which dissociates technical aspects from functional aspects. It begins with a preliminary study which essentially consists of identifying the actors who will interact with the system to be built, the messages that the actors and the system exchange, producing the specifications and modelling the context (the system is a black box, the actors surround it and are connected to it, on the axis which links an actor to the system we put the messages that the two exchange with meaning). [3] The process is then structured around 3 essential phases:

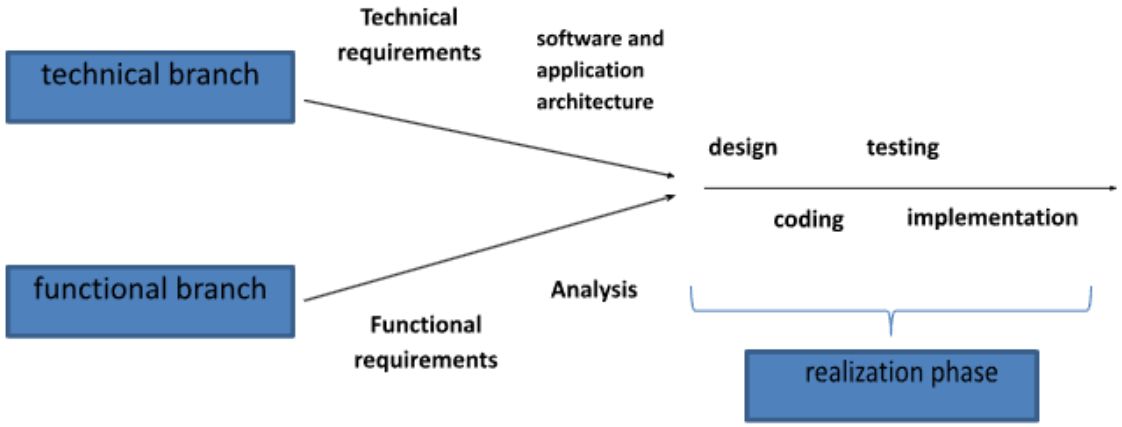

Figure 2.1: a graph clarifying the Two Tracks Unified Process

2.2.2 Requirement specification

Software requirements are the specifications of what a system should do, how it should behave, and what constraints it should satisfy. They are essential for communicating the users' expectations and needs and guiding the software's design, development, testing, and maintenance.[4]

Functional requirements

They are a set of specifications that the system delivered by our app should be and should do. The application should include these functional requirements:

- **Authentication:** This interface allows users to access their spaces after login.
- **Search:** This interface allows users to search the rooms available in the database by entering a keyword regarding the research subject.
- **User management:** This space allows the admin to create or revoke user accounts.
- **Customer management:** this interface enables the admin to manage the hotel guests and clients as well as consult their information
- **Room reservations management:** This space allows the admin to manage room reservations made by customers and consult them.
- **Price management:** This space allows the reception manager to update room prices.
- **Invoice management:** This interface gives the manager the possibility reception to establish and manage invoices as well as consultations and prints.

Non-Functional requirements (Technical)

They are a set of specifications that describe the system's operation capabilities and constraints, they are the requirements that outline how well it operates.[5]Our app must have the following Technical Requirements:

- **Security:** Securing and respecting the confidentiality of user and hotel data.
- **Ergonomics:** The app should be nice to look at and easy to use.
- **Maintainability/ Modifiability:** refers to the ease with which a solution or its component can be fixed and enhanced
- **Reliability:** the app should function as intended indefinitely.

2.2.3 Analysis

which consists of precisely studying the functional specification to obtain an idea of what the system will achieve in terms of business.

2.3 Unified Modeling Language (UML)

2.3.1 Definition

UML, short for Unified Modeling Language, is a standardized modelling language consisting of an integrated set of diagrams, developed to help with system and software development by specifying, visualizing, constructing, and documenting the artefacts of software systems, as well as for business modelling and other non-software systems.
The UML is a very important part of developing object-oriented software and the software development process, it uses mostly graphical notations to express the design of software projects, and using it helps project teams communicate, explore potential designs, and validate the architectural design of the software.[6]

2.3.2 UML diagrams

The current UML standards call for 13 different types of diagrams: class, activity, object, use case, sequence, package, state, component, communication, composite structure, interaction overview, timing, and deployment.[7]
To create our system, only three of those are needed:

- Use Case diagram
- Sequence diagram
- Class diagram

2.4 Identifying use cases

2.4.1 Global diagram

In the following figure, we present our global use case diagram

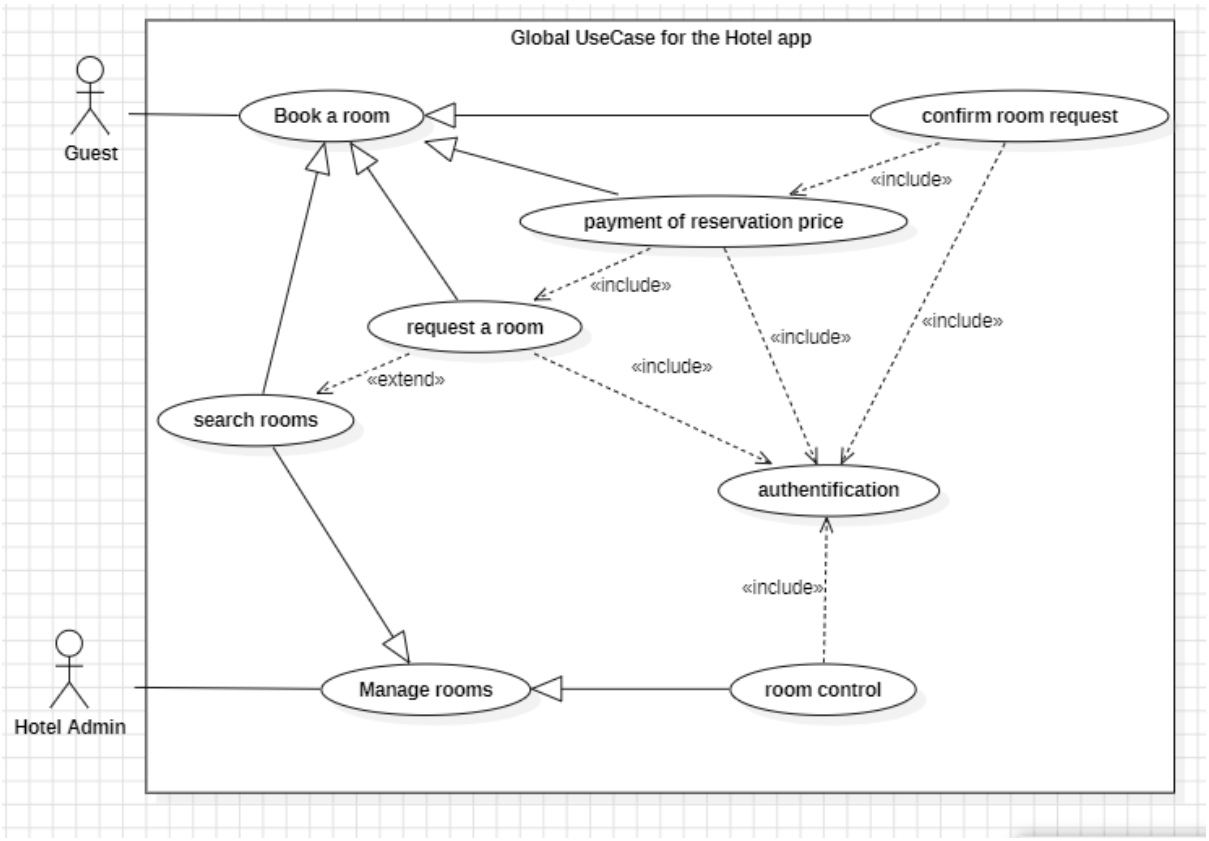

Figure 2.2: Global Use Case diagram

2.5 Detail of the use cases

2.5.1 use case ”Search room”

- The user initiates the search by interacting with the application.
- The application sends a request to the server, asking for available rooms based on the user’s criteria.

Use case diagram

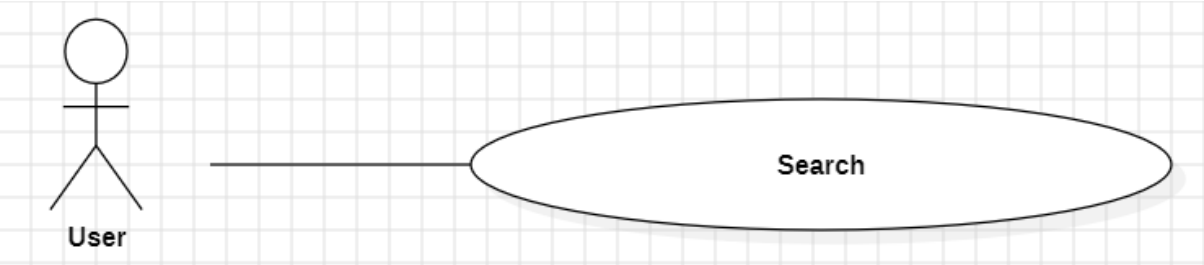

Figure 2.3: use case diagram (search)

Sequence diagram

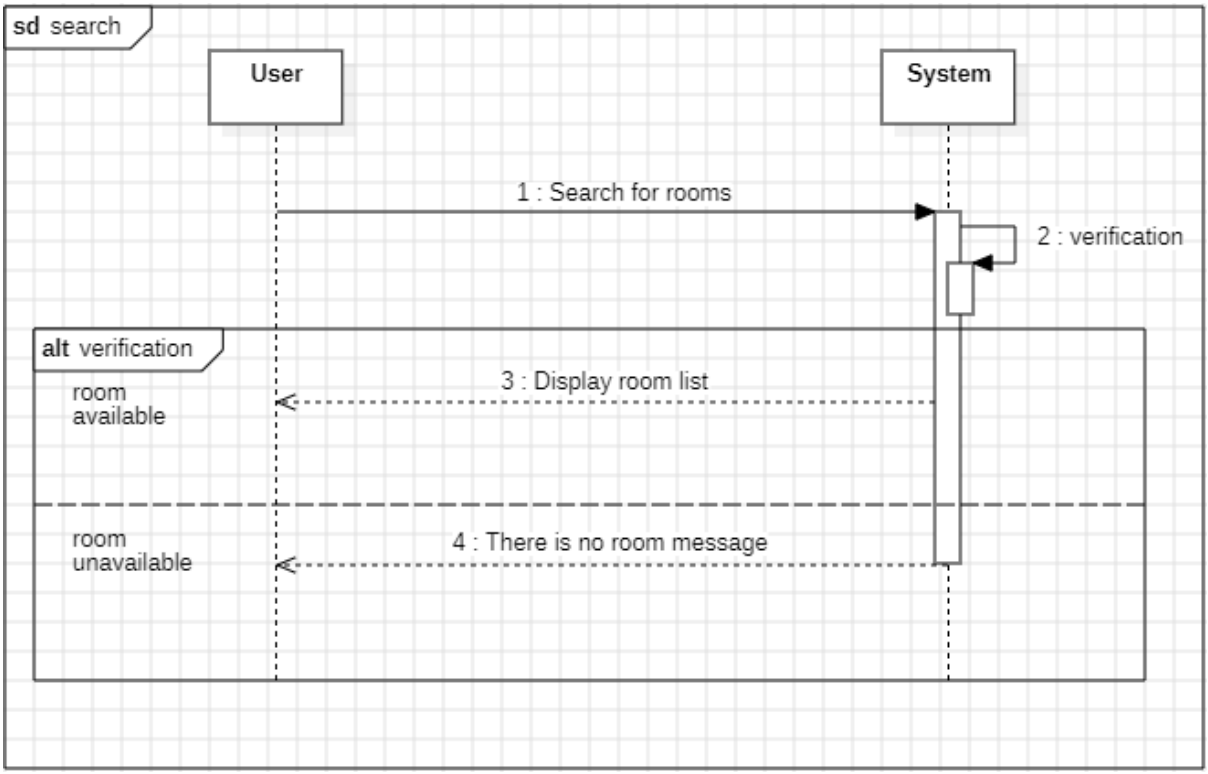

Figure 2.4: Sequence diagram for Searching a room

2.5.2 Use case ”Requesting a room”

- The server processes the request, retrieves relevant information from the database and sends back a list of available rooms to the application.
- The application displays the available rooms to the user, who selects one and requests to book it.
- The application sends the booking request to the server.
- The server processes the booking request, updates the database and sends a confirmation back to the application.

Use case diagram

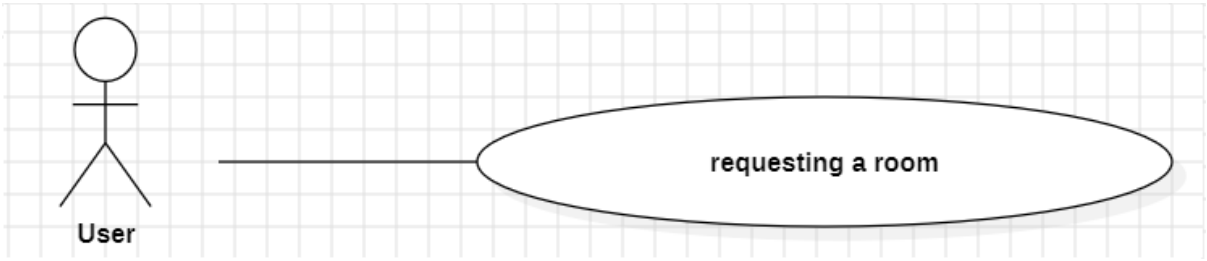

Figure 2.5: use case diagram(requesting a room)

Sequence diagram

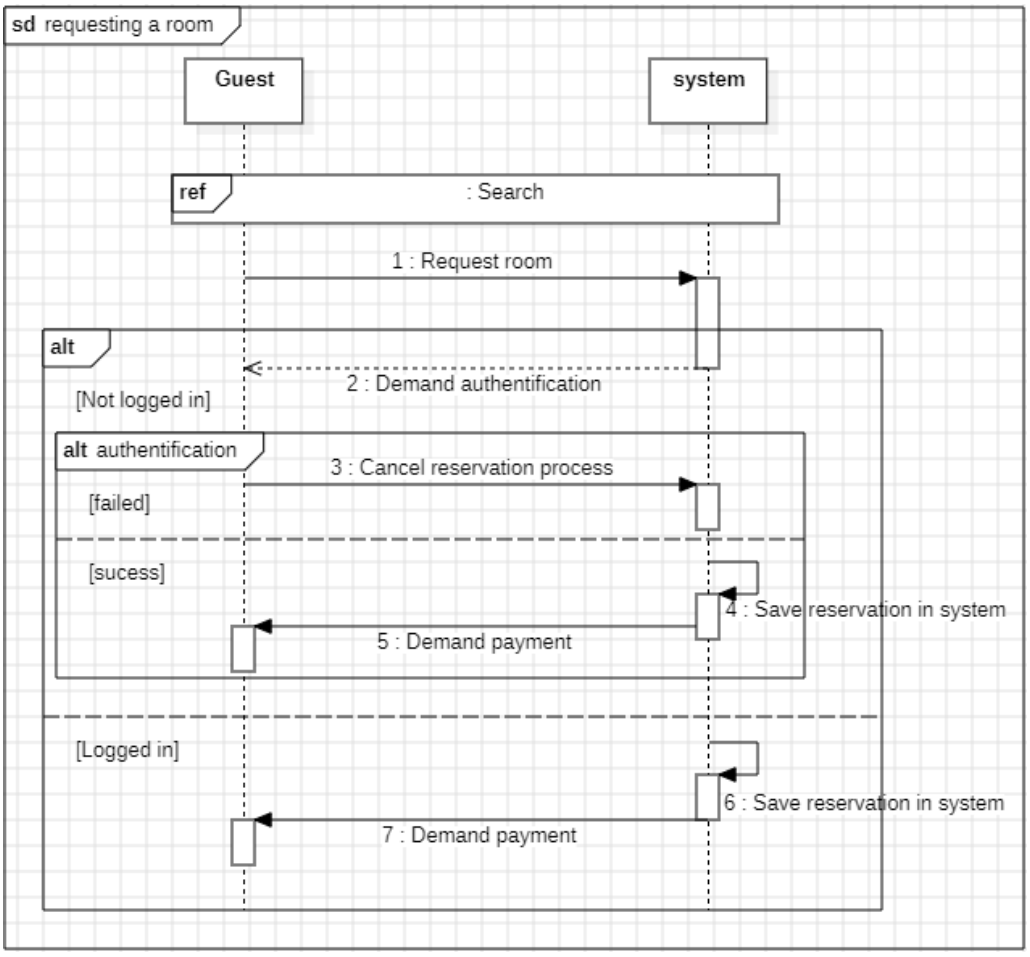

Figure 2.6: Sequence diagram requesting a room

2.5.3 Use case ”Payment”

- Upon receiving the confirmation, the user proceeds to the payment stage.
- The user enters payment details through the application.
- The payment service processes the payment transaction and sends a response (success or failure) back to the application.
- If the payment is successful, the application updates the booking status as paid.

Use case diagram

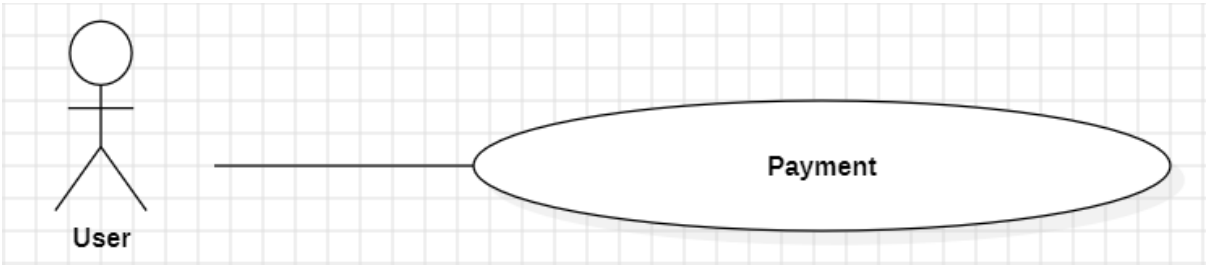

Figure 2.7: use case diagram (payment)

Sequence diagram

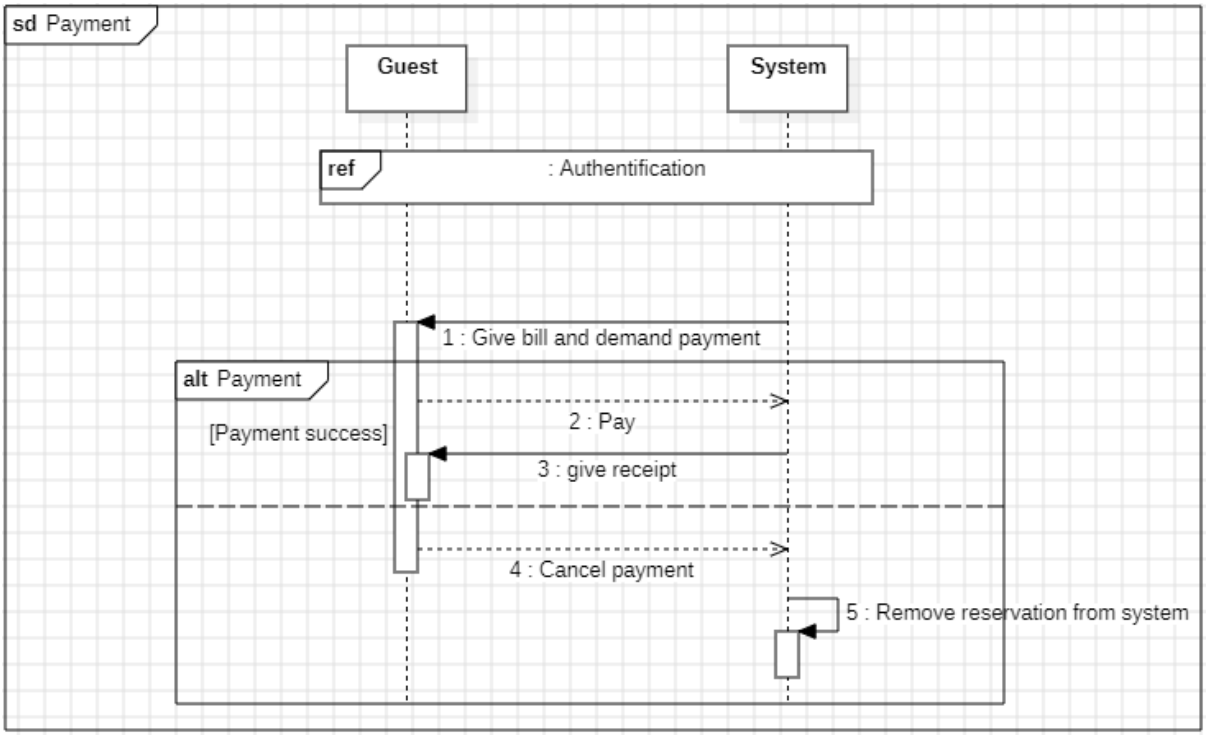

Figure 2.8: Payment sequence diagram

2.5.4 Use case ”Authentification”

- Before proceeding with payment, the user might need to authenticate themselves (e.g., login).
- The application prompts the user to log in or create an account if they haven’t already.

Use case diagram

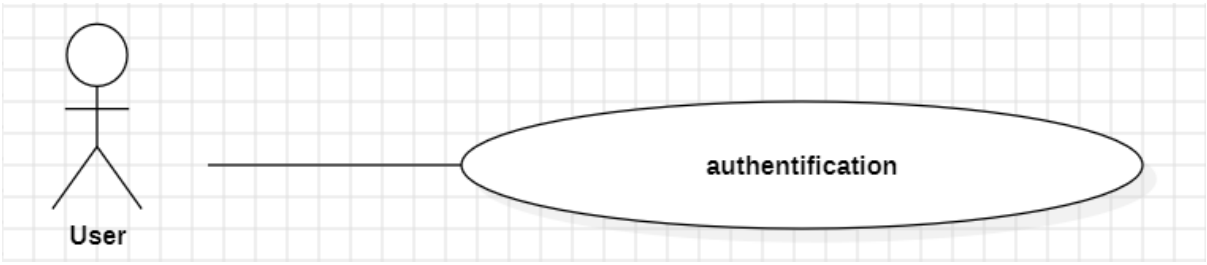

Figure 2.9: use case diagram(authentication)

Sequence diagram

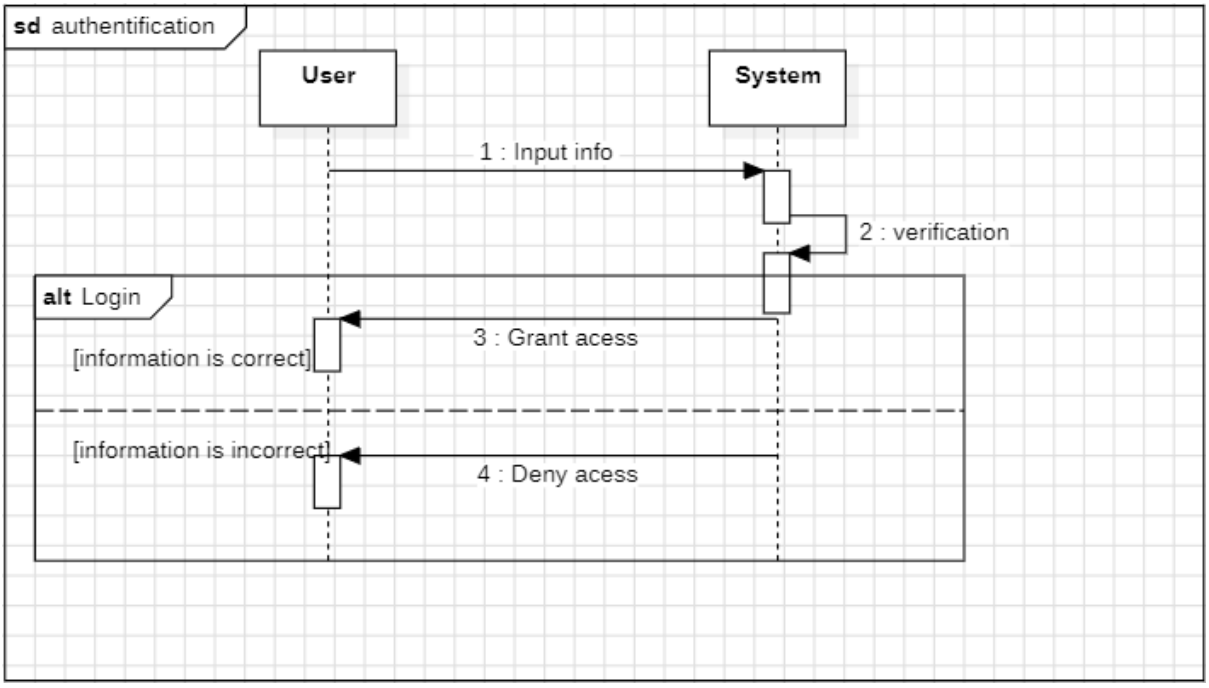

Figure 2.10: authentication sequence diagram

2.5.5 Use case ”Confirmation”

- The guest provides their credentials, and the application verifies them with the system.
- If authentication is successful, the user is authorized to proceed with the payment process.

Use case diagram

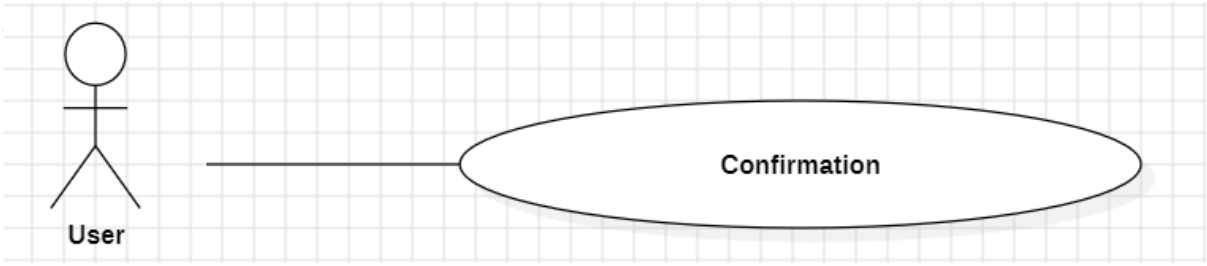

Figure 2.11: use case(Confirmation)

Sequence diagram

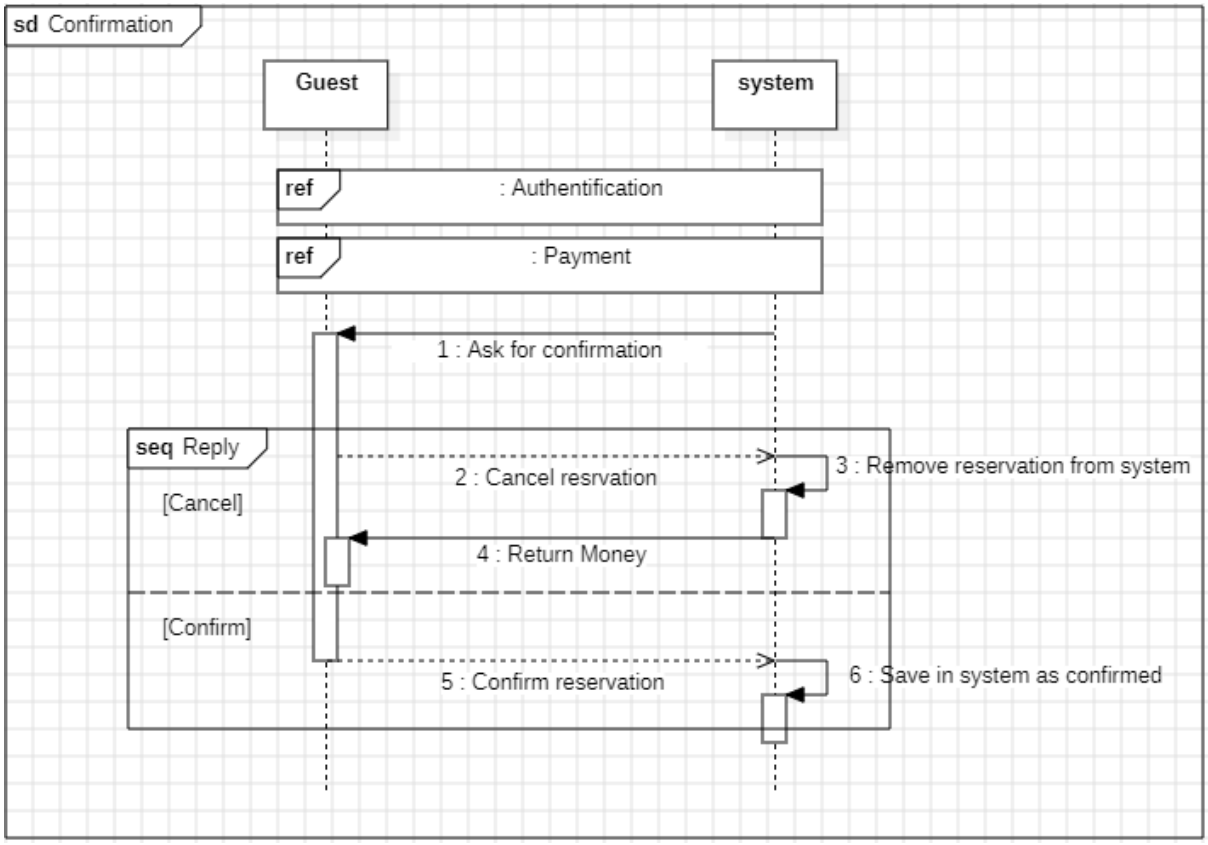

Figure 2.12: Confirmation sequence diagram

2.5.6 Use case ”Room control”

- After admin authentication, they may have options to control and change room details, such as number of beds, or guest capacity.
- The admin interacts with the application to access these controls.
- The application sends corresponding requests to the system.
- The server processes these requests and changes the database correspondingly.

Use case diagram

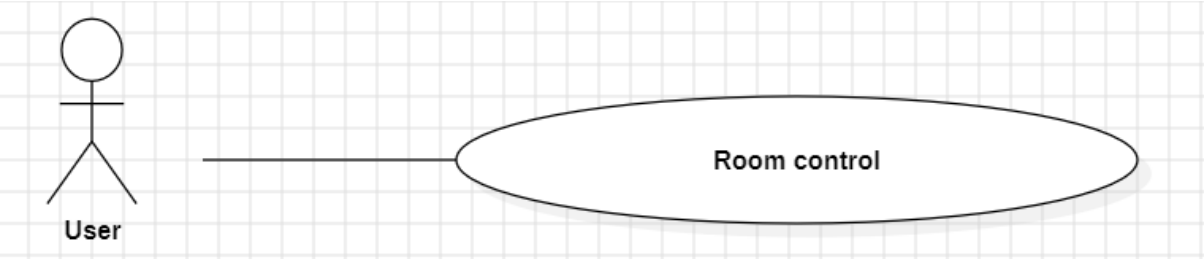

Figure 2.13: use case Room control

Sequence diagram

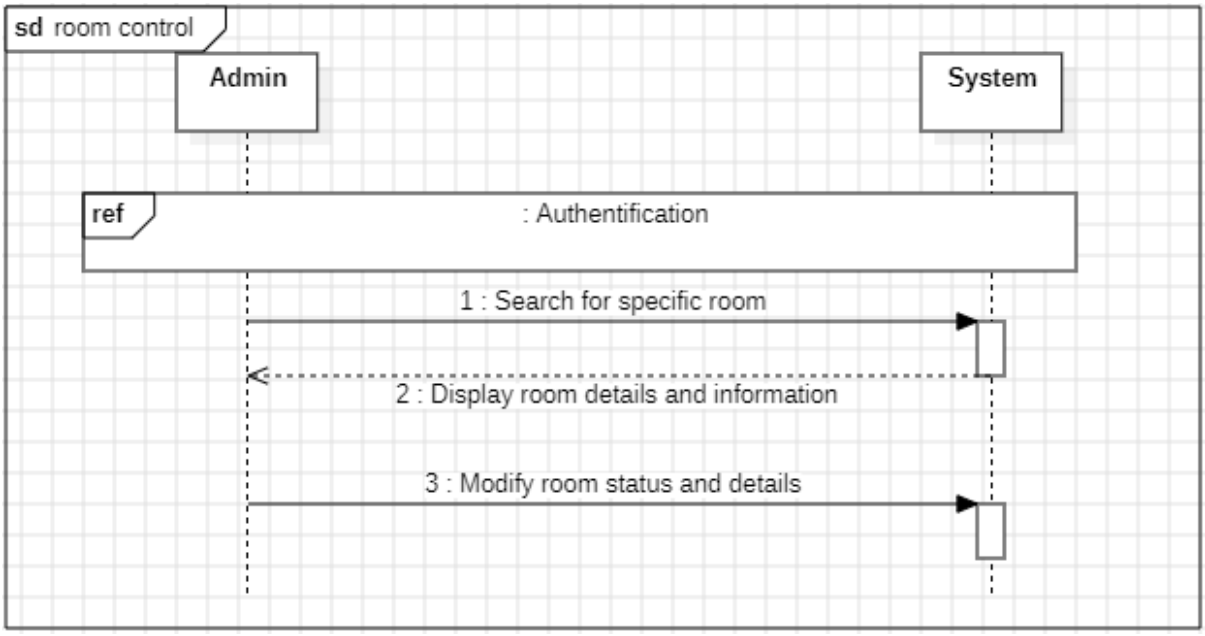

Figure 2.14: Room control sequence diagram

2.6 Class Diagram

- Login : Represents the interface where users (guests) enter their credentials to access the system.
- Guest : Represents a user who wants to book a room.contains information such as name, contact details, and account information.
- Reservation: Represents the booking details made by a guest, including check-in and check-out dates, room type, and any special requests.
- Room : Represents a room in the hotel.contains details such as room number,type, availability status, and rate.
- Payment : Manages payment transactions.contains information such as payment method, amount, and transaction date.
- Confirmation : Represents the page that displays the payment confirmation to the guest.it shows details of the transaction and booking confirmation.

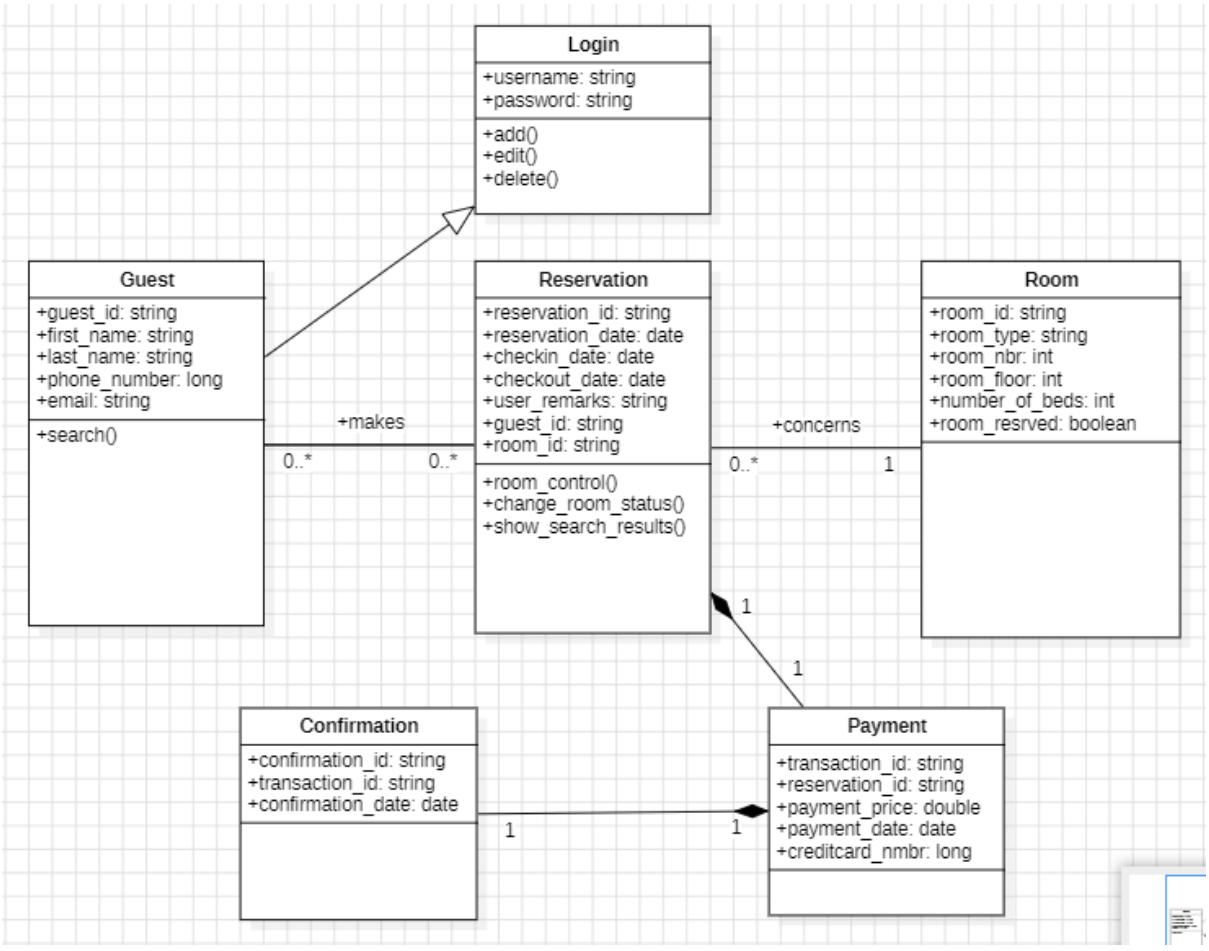

Figure 2.15: Class diagram

2.7 Conclusion

The Two-Track Unified Process (2TUP) facilitated the design of the hotel reservation software by integrating two parallel tracks: requirements analysis and design and implementation phases. The Unified Modeling Language (UML) served as a bridge, allowing clear communication and visualization of a system's structure, behaviours, and interactions. A use case diagram provides a comprehensive view of the system's functionality, depicting the guests and the administrators and their interactions with the system. At the same time, we used sequence diagrams which show the sequence of messages exchanged between objects during specific use cases. These steps streamlined the development process and ensured a cohesive and functional hotel booking solution.

The next chapter will focus on showing the complete application the steps taken to code and develop it, and its implementation in a real hotel reservation process.

Chapter 3

IMPLEMENTATION

3.1 Introduction

The implementation phase is the last phase of our project and is presented as a phase Most important because it handles the process tab of the project. We first start with a brief explanation of the ergonomics as well as the programming languages we used in creating our application for Hotel booking and Database Implementation. We then move on to an overview of the most common interfaces. Most important of our application.

3.2 Development Environment

3.2.1 Android Studio:

Android Studio is the official integrated development environment (IDE) for Android app development, provided by Google. It offers a comprehensive suite of tools for designing, building, and testing Android applications. With features like code editing, debugging, performance profiling, and device emulation, Android Studio streamlines the development process for both beginners and experienced developers. Its user-friendly interface and extensive documentation make it a popular choice among Android developers worldwide.[8]

3.2.2 Programming Languages:

JAVA:

Java, is an object-oriented programming language renowned for its versatility and platform independence. It's widely used for developing various software applications, from web applications to enterprise systems.
In the context of Android development, Java serves as the primary programming language for creating Android apps. Android Studio, the official integrated development environment (IDE) for Android development, supports Java alongside Kotlin, another programming language. Java code is used to write the logic and functionality of Android apps within Android Studio, making it an essential skill for Android developers.[10]

XML:

In Android Studio, XML (eXtensible Markup Language) is used for designing user interfaces (UI) in Android applications. XML files define the layout and structure of UI components such as buttons, text fields, and layouts. Developers use XML to create and configure UI elements, and Android Studio's visual layout editor provides a drag-and-drop interface for designing UIs graphically.[11]

3.2.3 SQLite:

SQLite is commonly used as the underlying database engine for storing structured data within Android applications. Developers can interact with SQLite databases using SQL queries, and Android provides a set of APIs for performing CRUD (Create, Read, Update, Delete) operations on the SQLite databases. So, SQLite can be seen as a critical component of the development environment for Android app development.[9]

3.3 Application showcase

3.3.1 Main / Booking interface

This next figure represents the main page that would appear after launching the app for the first time, the main menu is also the search page that allows the guest to search for their preferred room, pressing the ”search for availability” button takes to the page in Figure 3.2.

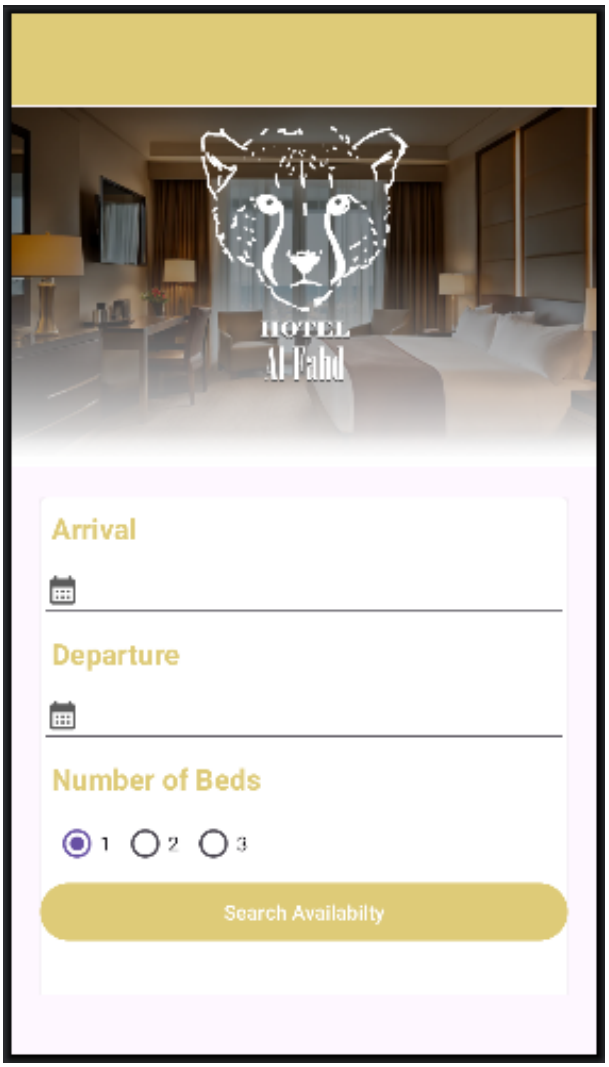

Figure 3.1: Main interface

3.3.2 Search Results and Payment interfaces

This figure represents the search results page that would appear after searching for room availability, it displays the available rooms in the desired time with the specified preferences. The rooms are displayed with their respective descriptions next to their image and their price per night.

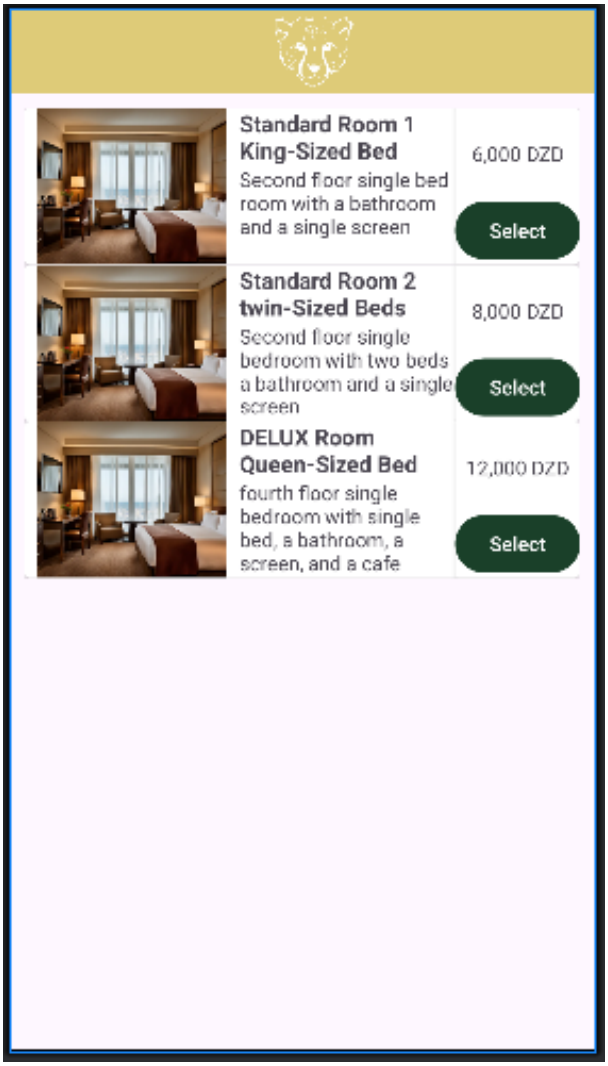

Figure 3.2: search results interface

While this next figure is the interface shown to the guest after choosing a room that appeals to their liking, the guest is presented with payment options and a total amount to be paid for the booking process to be completed.

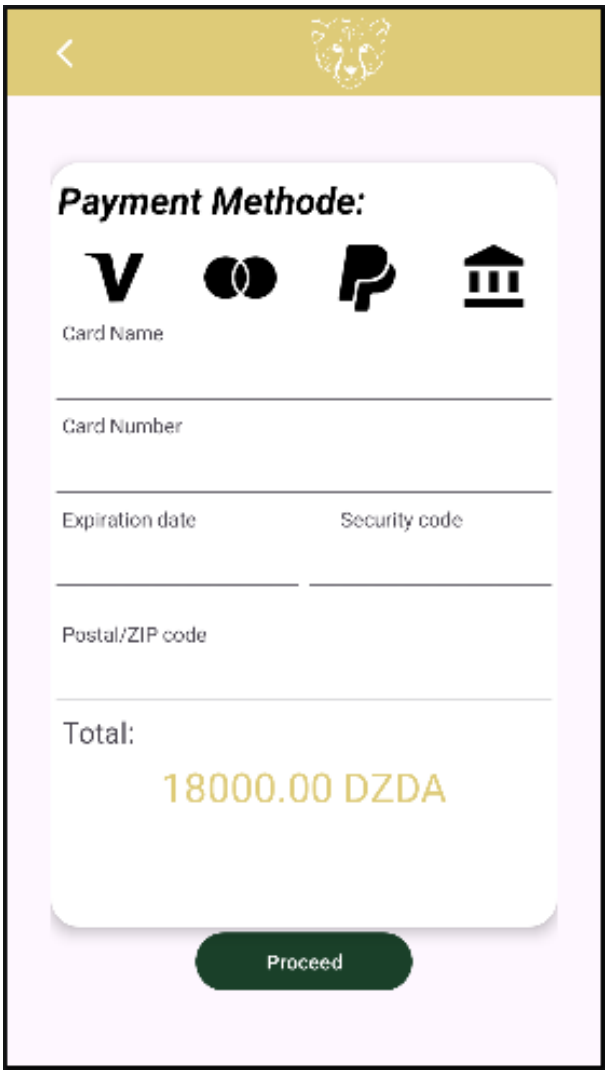

Figure 3.3: Payment interface

3.3.3 Login and Register interfaces

The next figure is the login page, this page will automatically show up to the guest after choosing a room if they are not already logged in, or the user can manually access it from the main menu. This page is also where the admin and guest are assigned to their separate roles in the system.

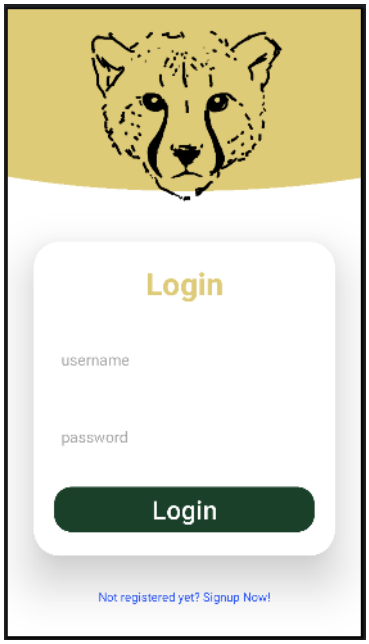

Figure 3.4: Login interface

If the guest does not have an account in the system already, this figure shows the interface displayed to the guest that allows them to create a new account in the system to then proceed with their booking process.

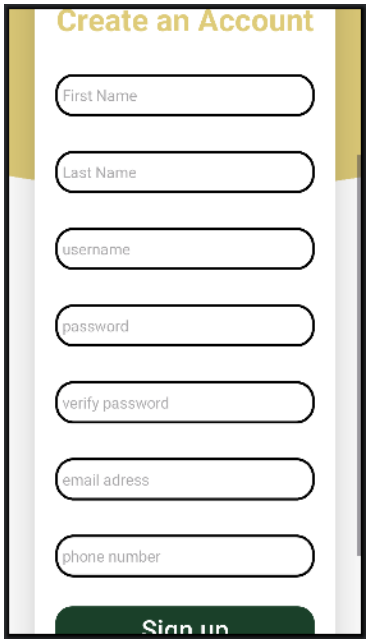

Figure 3.5: Signup interface

3.4 Conclusion

In this chapter we briefly describe the process To create our application by selecting an environment Development. A description of our interfaces However, the work remains open to potential extensions and improvements

General Conclusion

The work done in this thesis aims to achieve An application for managing hotel reservations to facilitate the reservation process for the customer and meet his desires on the one hand, and automated management for the hotel owner on the other hand.
We analyzed the problem and were able to design The application, which we hope will be an effective and useful solution for Booking management.
Then we started the second chapter, using the 2TUP process and discussing it. The modelling language for designing our application is UML language. We have also identified the actors you interact with the Application, then we describe the needs of each actor in the use case. Also, for each use case, we have created a sequence diagram to represent interactions between objects and a system that indicates the chronology of exchanges. Then an investigation was made into a static model represented by the class diagram to provide an overview of the database.
Finally, we took the time to complete our project while selecting Development tools, as well as programming languages, used and monitored An overview of the interfaces it includes.
We hope to develop this application to include more than one hotel in the future.

Bibliography

[1] https://intelity.com 2-4-2024 12:00PM

[2] https://www.revfine.com 28-4-2024 4:40PM

[3] : Kazi Aouel Bassim et Rostane Zakaria/Suivie des enseignements du LMD par application de la méthode 2TUP/ Projet de Fin d'Etudes pour l'obtention du Diplôme d'Ingénieur d'Etat en Informatique / 08 -11- 2007. 23-4-2024 1:30PM

[4] Eric Papet / 2UP-ARCHI-DEV1 / Document: version 1.00/12-12-2000. 22-5-2024 8:30PM

[5] https://www.altexsoft.com/blog/non-functional-requirements/ 12:00PM

[6] R. fanader et H. Leroux. UML principes de modélisation. Presses de l'Université de Québec, paris, 2009. 25-4-2024 5:30PM

[7] Laurent Piechocki / COURS UML / Document : version 13 / janvier 2003. 25-4-2024 6:00PM

[8] https://www.loginradius.com/ 18-5-2024 16:30PM

[9] https://www.sqlite.org/ 18-5-2024 17:57PM

[10] https://www.geeksforgeeks.org/ 18-5-2024 14:03PM

[11] https://www.w3schools.com/ 18-5-2024 13:49PM