



TECHNICAL DOCUMENTATION OF HOTEL RESERVATION SYSTEM (HRS)

Project manager:
Fatima Al-Ibrahim
COE Hotel | 7th may





Version	date	author	Change Description	Section/Page	Reason for Change
1	24/12	Fatima	n/a	n/a	n/a
1.1	30/12	Fatima	Add security to the	Page No. 14	to protect it from



TABLE OF CONTENTS

Introduction. 2

Scope. 3

Functional and non-functional features of Hotel Reservation System.. 4

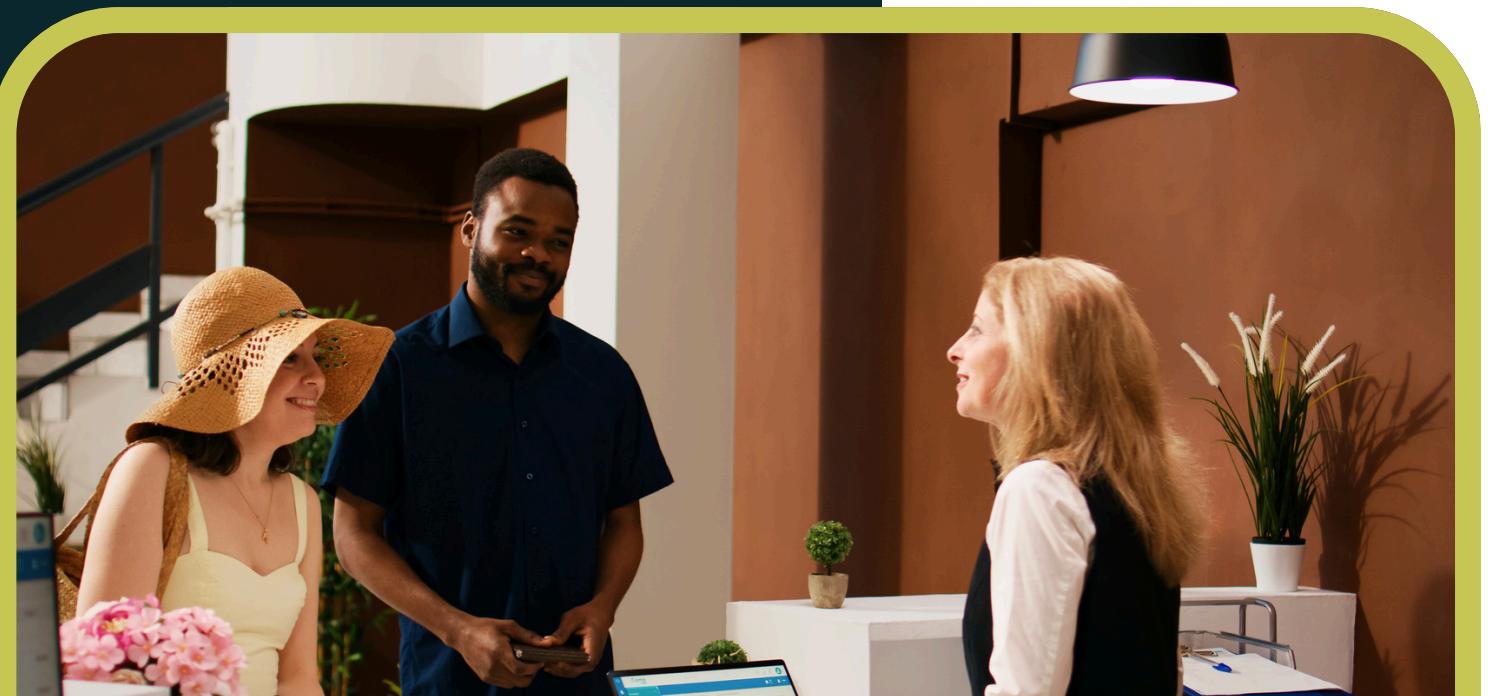
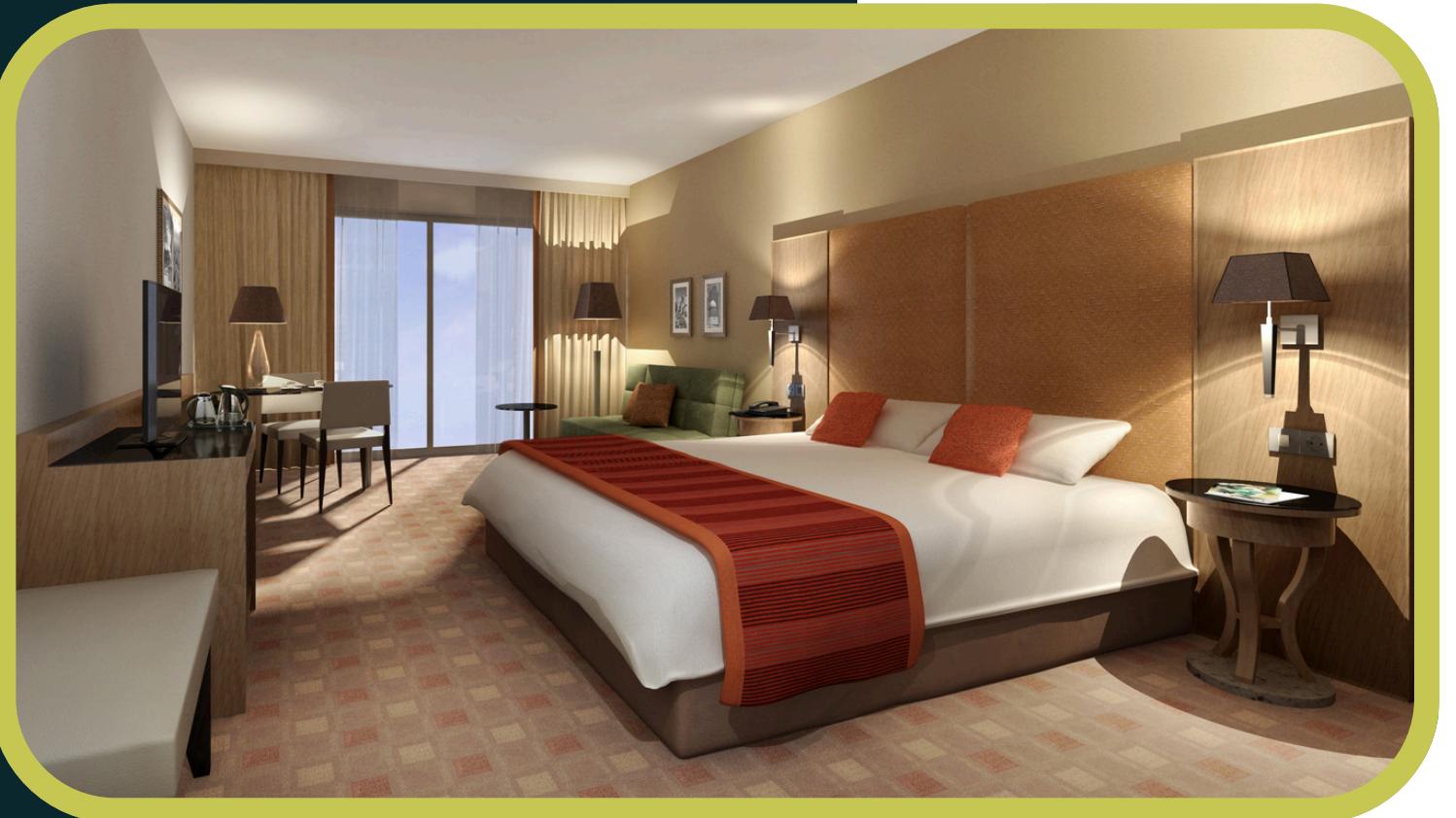
System design – Physical and logical 5

Debugging and Testing. 8

Security. 14

Technical Terms. 15

Reference. 16



[Home](#)[About](#)[Contact](#)

INTRODUCTION

The Hotel Management System is a tool for booking the rooms of hotel through online by the customer. It provides the proper management tools and easy access to the customer information.

The system provides to managing the reservation process. The system will able user to book a room with services including gym and spa.

The system provide user the application of hotel to make reservation for visitors and confirm it by searching rooms, add visitor information, and payment process. The system will find solution for recording the bill automatically. The procedure of hotel reservation has been analyzed by logging into booking application for hotel.

The system is build and developing using Java.



SCOPE



Name of the Project: Hotel Reservation System

Type of the project: Hospitality

Business Requirements (Functions): Functional Requirements: Thorough communication with stakeholders, analyzing existing processes, and understanding the organization's goals.

Business Process Requirements: Create and study the enterprise business process list.

User Requirements: Surveys and interviews with potential users.

Non-Functional Requirements: Are a set of specifications that describe the system's operation capabilities and constraints.

Regulatory & Compliance Requirements: Regulatory compliance is the adherence to a set of rules and requirements as defined by governments and regulatory bodies.

Target Audience: A system designed for adults who want to book a hotel for their vacation and enjoyment.

In-Scope Functional Areas: Room booking, Payment

Features Included: Includes services such as gym, spa and dining.

Exclusions: There is no increase in rooms or modification to the reservation upon confirmation, and there is no increase in the price when adding a room after confirming the reservation.





FUNCTIONAL AND NON-FUNCTIONAL FEATURES OF HOTEL RESERVATION SYSTEM



Functional:

- Add payment
- Search rooms
- Manage room detail
- The application allows users to create accounts and log in
- The user can review items in the cart

Non-Functional:

A program running on Windows 10 must be able to run on Windows 11 without any change in its behavior and performance.

The system should be able to handle 20 million users without performance deterioration.

The website pages should load in 3 seconds with the total number of simultaneous users <5 thousand.



SYSTEM DESIGN – PHYSICAL AND LOGICAL

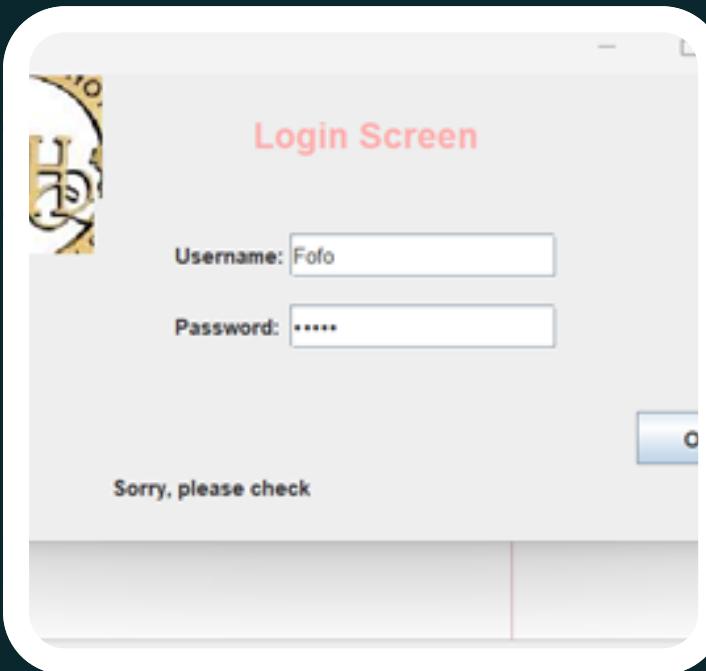
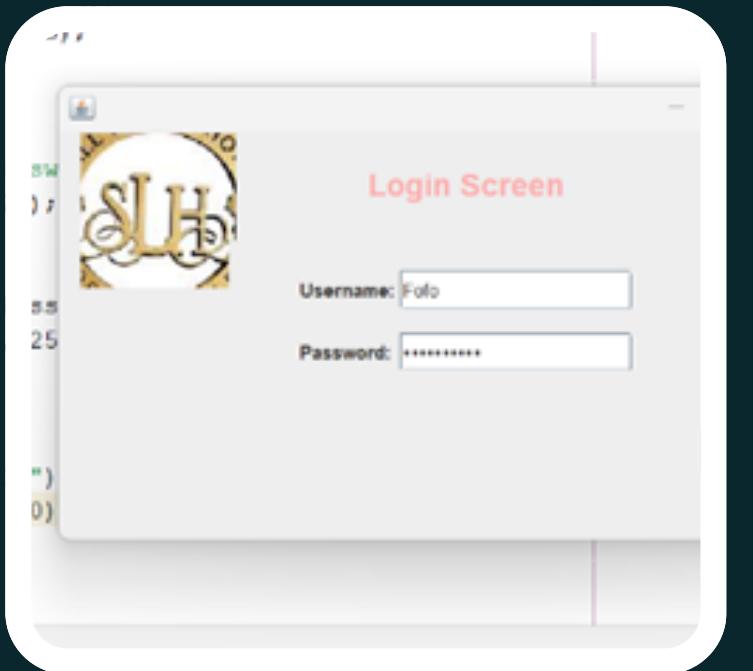
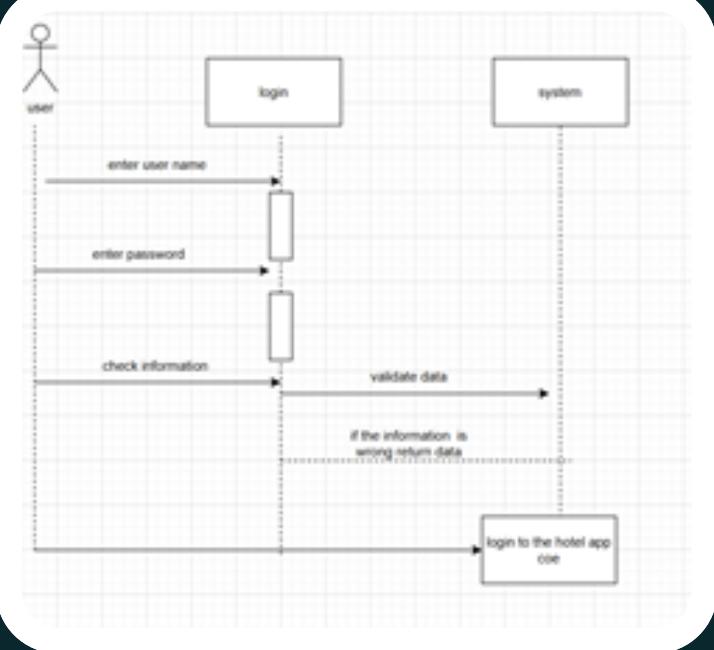
SEQUENCE DIAGRAM SHOWS HOW OBJECTS INTERACT WITHIN A SYSTEM OVER TIME, ILLUSTRATING THE ORDER OF MESSAGES EXCHANGED BETWEEN THEM

USE CASE DIAGRAM SHOWS HOW USERS INTERACT WITH A SYSTEM AND WHAT FUNCTIONS IT PROVIDES

DATA ENTRY PAGE TO LOG IN TO THE APPLICATION

ROOM SERVICE RESERVATION PAGE

THE ERROR MESSAGE IF PASSWORD OR USERNAME IS INCORRECT





DEBUGGING AND TESTING



Logging and handling errors

```
java
/
class Fatima_ {
    private static final Logger logger = Logger.getLogger(javaapplication51.class.getName());
    private LocalDate startingDate;
    private LocalDate endingDate;
    String Reservation_id;
    int Room_id;
    int Room_rate;

    public Fatima_(String f, String r, int id, int rate) {
        logger.info("Application started");
        logger.setLevel(Level.ALL);
        ConsoleHandler handler = new ConsoleHandler();
        SD-AD1- Unit 04 -GA1          Page 8 of 29          COEK,V1,April 2025/ ET. NA
    }
}

International Technical Colleges (ITC)
```

```
user starting date
026-12-20
user ending date
026-12-24
user details for reservation id
111
user room id
1111
user room rate
0
Starting Date: 2026-12-20
Ending Date: 2026-12-24
Total Price: 0.0
Reservation: 111
Room id: 1111
Room rate: 0
BUILD SUCCESSFUL (total time: 1 minute 55 seconds)
```

Error handling using try and catch statement

```
user starting date
-12-12
user ending date
-12-20
user details for reservation id
111
user room id
1111
user room rate
0
Starting Date: 2026-12-20
Ending Date: 2026-12-24
Total Price: 0.0
Reservation: 111
Room id: 1111
Room rate: 0
Division by zero is not allowedBUILD SUCCESSFUL (total time: 19 seconds)
```

```
try {
    p=++numberNights*Room_rate;
}
catch (ArithmaticException e){
    System.err.print("Something wrong!");
}
return ++numberNights*Room_rate;
```



DEBUGGING AND TESTING



Error handling using if statement

```
if(Reservation_id == 0){  
    System.out.println("Division by zero is not allowed");  
  
}  
  
dates();  
  
Duration duration = ChronoUnit.DAYS.between(startDate,  
endData);  
System.out.println("Duration: " + duration);  
if(duration < 0){  
    System.out.println("Ending date must be after starting date");  
}  
else{  
    double totalPrice = roomRate * duration;  
    System.out.println("Total Price: " + totalPrice);  
}
```

```
at -> JavaApplication40 [run]  
run:  
Enter starting date  
2025-12-20  
Enter ending date  
2025-12-24  
Enter details for reservation id  
111  
Enter room id  
111  
Enter room rate  
-1  
Starting Date: 2025-12-20  
Ending Date: 2025-12-24  
Total Price: 0.0  
Reservation: 111  
Room id: 111  
Room rate: -1  
BUILD SUCCESSFUL (total time: 16 seconds)
```

Manual unit testing frame

```
System.out.println("Test 2 passed");  
else  
    System.out.println("Test 2 failed");  
  
System.out.println("Test 3 passed");  
else  
    System.out.println("Test 3 failed");  
  
System.out.println("Test 1 is failed");  
Error: Ending date must be after starting date.- Test is failed  
Starting date is entered = Test is passed  
Starting Date: 2025-12-20  
Ending Date: 2025-12-26  
Total Price: -8.0  
Reservation: 111  
Room id: 111  
Room rate: -1  
BUILD SUCCESSFUL (total time: 23 seconds)
```

```
System.out.println("Test 2 passed");  
else  
    System.out.println("Test 2 failed");  
  
System.out.println("Test 3 passed");  
else  
    System.out.println("Test 3 failed");  
  
System.out.println("Test 1 is failed");  
Error: Ending date must be after starting date.- Test is failed  
Starting date is entered = Test is passed  
Starting Date: 2025-12-20  
Ending Date: 2025-12-26  
Total Price: -8.0  
Reservation: 111  
Room id: 111  
Room rate: -1  
BUILD SUCCESSFUL (total time: 23 seconds)
```





DEBUGGING AND TESTING



The debugging is the number of nights multiplying by Room rate

number of nights multiplying by Room rate

```
    ce() {  
        s = ChronoUnit.DAYS.between(startingDate, endingDate);  
        nights*Room_rate;
```

```
- JavaApplication40 (run) X  
run:  
Enter starting date  
2025-12-20  
Enter ending date  
2025-12-24  
Enter details for reservation id  
111  
Enter room id  
1111  
Enter room rate  
50  
Starting Date: 2025-12-20  
Ending Date: 2025-12-24  
Total Price: 250.0  
Reservation: 111  
Room id: 1111  
Room rate: 50  
BUILD SUCCESSFUL (total time: 48 seconds)
```

Breakpoints

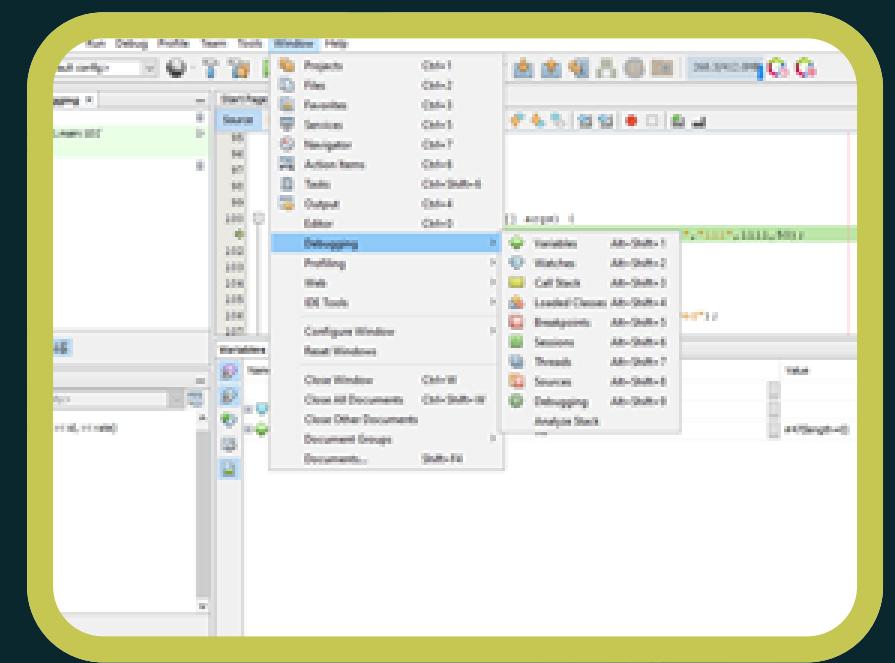
```
public double calcPrice() {  
    long numberNights = ChronoUnit.DAYS.between(startingDate, endingDate);  
    leng p;
```



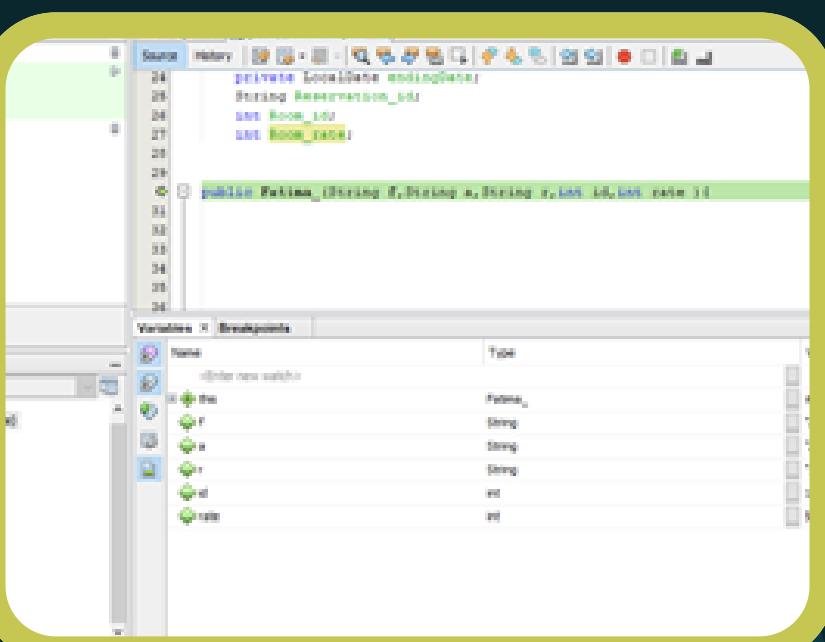
DEBUGGING AND TESTING



Step through the code



IDE's built-in debugger





TEST CASES

Test Case	Input	Expected Outcome	Actual Outcome	Status (Pass/Fail)	Analysis - Type of error (if any)
multiplying by 100	Number nights*100	Error is not showing the right result	Wrong result	fail	Logical error
multiplying by room rate	Number Nights*Room rate	Is showing the right result of room rate	Right result	pass	n/a
multiplying by 0 its wrong	If its multiplying by 0 it will show message its wrong	Its showing the message	Right result	pass	N/a



SECURITY



Authentication versus access control: To protect content from unauthorized application users, and to control access to administrative functions, search supports user authentication and authorization.

- Secure API authentication with OAuth
- you can use access control lists (ACLs) to configure permission to access workspace objects

Data protection and encryption: Data encryption protects data from being stolen, changed, or compromised.

- Symmetric encryption uses the same key for encryption and decryption
- Asymmetric encryption has a private key held by the owner of the data, and a public key granted to the recipient of the data.





TECHNICAL TERMS

[Home](#)[About](#)[Contact](#)

Object-oriented programming: This programming paradigm or methodology models software based on data rather than functions.

Operating system: This refers to the system the computer uses to perform all its basic functions.

URL: This is a term for a website's unique identifier.

VPN: This is an acronym for virtual private network, which is a service that creates a secure, protected Internet connection for users on a public network.

Wi-Fi: Wi-Fi is a group of wireless network protocols that allow users to access the Internet.

Keywords: These are the phrases search engines use to locate online content.

DevOps: This is a software development process that combines software development with IT operations to improve the development life cycle.





REFERENCE

<https://www.slideshare.net/slideshow/hotel-management-system-project/71246325> <https://www.scribd.com/document/312077812/Project-Report-on-Hotel-Reservation>

<https://www.scribd.com/document/430037015/Functional-Non-Functional-Requirements-of-Hotel-Management-System>

<https://www.mu.edu.sa/sites/default/files/content/2019/10/Hotel%20Booking%20management%20system.pdf>
Unit 2, 4

<https://www.indeed.com/career-advice/career-development/technical-terms> <https://www.ibm.com/docs/en/wca/3.0.0?topic=security-authentication-versus-access-control> <https://cloudian.com/guides/data-protection/data-encryption-the-ultimate-guide/#:~:text=The%20science%20of%20encrypting%20and,and%20protected%20against%20unauthorized%20access.>

<https://docs.databricks.com/aws/en/security/auth/>