A blue and white logo

Description automatically generated

EGYPT HOTELS RESERVATION

Software Engineering CSCI313

fares khater 222000087

Moaz Badran 222000052

Mostafa Amer 221001207

Ahmed Ghrbawi 222000066

Anfal Hamada 222000065

Table of Contents

1- **Introduction**

* 1.1 Purpose……………………………………………..
* 1.2 Scope………………………………………………..
* 1.3 Technologies we will use……………………………
* 1.4 Overview……………………………………………….

**2-Overall Description**

* 2.1 Product Perspective………………………………………
* 2.2 Constraints………………………………………………..
* 2.3 User Characteristics……………………………………….
* 2.4 Assumptions & Dependencies………………………………

**Functional Requirements**-3

* 3.1 User Class 1 – The User…………………………………………..
* 3.2 User Class 2 – Receptionist……………………………………
* 3.3 User Class 3 – The Hotel Manager……………………………..

Non-Functional Requirements…………………….

**5-Diagrams**

* 5.1 Use Case Diagram……………………………………………..
* 5.2 Use Case Scenarios……………………………………………..
* 5.3 Class Diagram……………………………………………………

5.4.1sequence Diagram for booking…………………………..

5.4.1 sequence Diagram for remove booking……………

5.5 state Diagram…………………………………………………..

5.6 Activity Diagram………………………………………………….

5.7 context model………………………………………………………

5.8 Architectural model……………………………………………….

**6.wire frames**………………………………………………………………

7.**Test case**………………………………………………………………………29

8.Observer……………………………………………………..

9.Adabtor…………………………………………

10.Single ton……………………

11.Risk and Security Requirements

* 11.1 Risk Identification …………………………………………..
* 11.2 Risk Assessment ……………………………………
* 11.3 Mitigation Techniques ……………………………..
* 11.4 Requirement Refinement……………………………..
* 12.apandex……………………………………….

1. Introduction

1.1. Purpose

This SRS will provide the foundation and solid base of the requirements for our Hotel Reservation management system web application. It will allow for a clear understanding and provide a comprehensive view of what is to be expected from our newly introduced system, which is to be constructed. This document will explain the different functional and non-functional requirements of the reservation management system, the purpose of the features, the interfaces, and the constraints under which it will operate. Moreover, it can also be used in the future as the basis for a detailed understanding of how the project was started. It offers a guide for future developers and maintenance teams to follow in order to maintain and adjust this project as needed

1.2. Scope

Travelers often face the challenge of finding suitable

accommodations, especially in regions where they are not familiar. However, Egypt Hotels is here to make your travel experience seamless and enjoyable. Our goal is to provide a user-friendly website that offers an easy and convenient solution for booking and managing your stay at Egypt Hotels. Our web application will offer an intuitive and user-friendly interface for booking rooms at Egypt Hotels. Users can easily select their preferred check-in and check-out dates, room types, and any additional services they require. We will provide comprehensive information about Egypt Hotels, including room types. This will help travelers make informed decisions about their stay. Registered users will have the ability to manage their bookings, including making changes to their reservation or canceling it if necessary.

The software product to be developed will streamline the day-to-day operations and reservations at Egypt Hotels, catering to the needs of various stakeholders, including guests and the resort manager. Egypt Hotels Management System will encompass two primary user roles: Guests and the Resort Manager. Guests will have the ability to check room availability and make reservations, they can select rooms based on their preferences and requirements. Guests can view and select any available offers and activities provided by Egypt Hotels during their stay. The Resort Manager will have access to financial reports, including revenue, expenses, and profit. He can update or modify booking details related to room costs, special categories, check-in and check-out dates, room changes, and guest information.

1.3. Technologies we will use

▪ Html to create and structure webpages

▪ CSS to style and layout our web pages.

▪ Java Script to create interactive web content.

▪ Bootstrap framework to enable responsive development to style and layout our web pages.

▪ Django for back-end, and to rapidly develop and maintain our website.

▪ MySQL to manage the database.

1.4. Overview

The remaining sections of this document will describe in deep detail the overall specifications for Egypt Hotels, which will include different functional and non-functional requirements, class diagram and use case.

2. Overall Description

2.1. Product Perspective

Our website will stand out by offering our guests a straightforward and personalized booking experience tailored to their preferences and budget. Accommodation details play a central role in determining the pricing, and ensuring transparency for our guests. We provide a user-friendly interface for room reservations, allowing guests to easily select their preferred room type, check availability, and complete their booking. In addition to room reservations, guests have the option to customize their stay by selecting additional services and amenities. These may include add-ons like spa treatments, dining packages, or recreational activities. Our aim is to give guests the flexibility to craft their ideal stay based on their individual needs and interests. For added convenience, our website allows guests to view a detailed program of available activities and events during their stay. Whether it's a fitness class, local excursions, or on-site events, guests can browse and reserve spots in advance. This ensures a seamless and well-planned experience, tailored to their preferences. By focusing on simplicity, personalization, and ease of use, our website aims to provide a stress-free booking process, allowing guests to focus on the excitement of their upcoming stay with us.

▪ Three sign-in choices are available on the system: user, receptionist, and hotel manager. The first option allows users to sign in. Each category has unique functionalities and can use the system to accomplish different functions.

▪ The system will include a database system that will store all the guests'data, room details and employee accounts, in addition to all the information required for the reservation

A diagram of a server

Description automatically generated

2.2. Constraints

▪ Language Requirement: Software must be only in English

▪ Reliability Requirements: System should sync frequently to backup server in order to avoid data loss during failure, so it can be recovered.

▪ Implementation Constraint: The languages used to implement the website are HTML, CSS, JavaScript and Bootstrap, and the system will use the current standard MySQL database engine.

▪ The Internet connection is a constraint for the System to operate.

▪ No one should be permitted to sign in as a user unless the password and username are correct.

▪ The system should allow you to sign in if you already have an accountor join if you are creating an account for the first time.

2.3. User Characteristics

As previously stated, the website can be logged in by three people, the first is the user “customer”, the second is “the manager” and the third is “the receptionist”.

▪ The customer will have access to many features, such as booking a room after selecting its type, in addition to checking the prices and completing or cancelling the reservation based on his needs.

▪ The receptionist will have less authority than the manager. He is responsible for checking in and out. He will also be able to check the availability of rooms, receive customer complaints.

▪ The manager will have full control over the website, including the ability to access employee and customer data stored in the database, delete a user or employee, and modify or change things. He will also have a full view of the room details and reservations.

2.4. Assumptions & Dependencies

▪ It is assumed that users will open the website from any source or device.

▪ It is assumed that all users of this system have internet to access the site.

▪ This system depends on a reliable database, to store information about reservations, users, and staff.

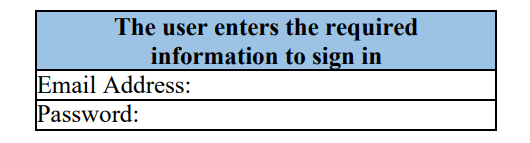
3. Functional Requirements

3.1. User Class 1 – The User

1Title: User Sign Up. Description: The system shall allow the user to create an account so that the user can log in to the website.

2Title: User Sign Up. Description: The system shall allow the user to create an account so that the user can log in to the website.

Description: The system shall allow the user to enter a valid email address and password to log in to the website.



3-Title: change password.

Description: The system shall allow the user to enter the old password , new password and New password confirmation to change the password.

A close-up of a form

Description automatically generated

4-Title: Check Availability.

Description: The system shall allow the user to enter the appropriate dates for him in addition to the type of room he wants with the features he wants, and the number of Childs he has and the available rooms with these additions are shown to him, so he chooses the room he wants with the appropriate price for him.

A blue and white card with black text

Description automatically generated

Title: Discover & Reserve.-5

Description: The system shall allow the user to explore the available room types, in addition to the features of each room and the view that the room overlooks, then he books the room he likes.

6-Title: My Booking & Profile.

Description: The system should allow the user to cancel and view his reservation, as well as write any complaints.

3.2. User Class 2 – Receptionist

Title: Receptionist log in.

Description: System shall allow the registered receptionist to login by providing a valid username and valid password.

Description: System shall display from the receptionist page the receptionist's functionalities, such as viewing the Rooms available, check in, check out.

Title: Check in/out review.

Description: System should allow the receptionist to be able to access the check-in/out for the customers

3.3. User Class 3 – The Hotel Manger

Title: Manger log in.

Description: System shall allow the manager to login by providing a valid user name and valid password.

Title: View staff report.

Description: The system shall allow the manager to check the reports of the staff and all the information that relates to them.

Title: View customer’s information.

Description: The system shall allow the manager to access customers' data, starting from their personal data to the reservations data they made, such as appointments, privileges, and anything else.

Title: View booking information.

Description: The system shall allow the manager to view all reservation data, including dates and prices and all its data.

4. Non-Functional Requirements

A nonfunctional requirement is a characteristic that determines how a system operates. It improves the efficiency of the software and demonstrates the system's performance. Based on the testing performed on our system we were able to analyze and quantify some quality attributes of the major non-functional requirements that must be highlighted.

Performance:

▪ The time it takes to navigate from one page to another should not be more than 3 seconds.

Availability:

▪ The system should be capable of operating 24 hours a day, seven days a week.

Recoverability:

▪ The system should recover after a failure within 5 hours.

Maintainability:

▪The system is easy to maintain as we followed the best methods for keeping clean code and there are no unused functions to make the system as maintainable and understandable as possible.

Reliability:

▪ The rate of failures should be no more than five times per year.

Useability:

▪ A ten-minute tutorial should be enough for a user to know all about the system’s features and be able to use it. It was determined after showing people who were unfamiliar with our system and who know how to browse on the internet, all agreed that it would not take more than ten minutes to become familiar with the system.

Security:

▪ To ensure system reliability and privacy and data security, we will utilize the following methodologies such as not everyone is allowed to control who can create, see, copy, change, or delete information, and no one is allowed to log in unless they have the correct password and username or email address, and the system should validate the password and username.

5. Diagrams

Use Case Diagram5.1

A diagram of a network

Description automatically generated

5.2. Use Case Scenarios

**1. Sign In**

**Actor**: User, Receptionist, Manager

**Description**: Allows actors to log into the system by entering valid credentials.

**Pre-conditions**:

The actor must have a registered account.

The system must be operational.

**Post-conditions**:

The actor is granted access to their respective dashboard.

Invalid login attempts trigger error messages.

**2. View Profile**

**Actor**: User

**Description**: Enables the user to view their profile information.

**Pre-conditions**:

The user must be logged into the system.

**Post-conditions**:

The profile information is displayed.

Changes (if edited) are saved successfully.

**3. Reserve Room**

**Actor**: User

**Description**: Allows the user to book a room.

**Pre-conditions**:

The user must be logged in.

Rooms must be available for the selected criteria.

**Post-conditions**:

A new booking is created and confirmed.

Payment is processed if required.

**4. Check Availability**

**Actor**: User

**Description**: The user can check available rooms by specifying dates, type, and features.

**Pre-conditions**:

The user must be logged in.

Rooms must exist in the system.

**Post-conditions**:

A list of available rooms is displayed to the user.

**5. Forget Password**

**Actor**: User

**Description**: Provides a mechanism to recover or reset the password in case the user forgets it.

**Pre-conditions**:

The user must provide a valid email or username for account recovery.

**Post-conditions**:

The password reset instructions are sent to the user.

**6. View Booking**

**Actor**: Use

**Description**: The user can view their previous and current reservations.

**Pre-conditions**:

The user must be logged into the system.

The user must have existing bookings.

**Post-conditions**:

All relevant booking information is displayed.

**7. Check In & Out**

**Actor**: Receptionist

**Description**: The receptionist can manage check-ins and check-outs for customers.

**Pre-conditions**:

The receptionist must be logged in.

The booking must exist in the system.

**Post-conditions**:

The booking status is updated to "Checked In" or "Checked Out."

**8. View Users**

**Actor**: Manager

**Description**: Enables the manager to view all users registered in the system.

**Pre-conditions**:

The manager must be logged in.

**Post-conditions**:

A list of registered users is displayed.

**9. View Booking Info**

**Actor**: Manager

**Description**: Allows the manager to view detailed information about reservations, including prices, dates, and customer details.

**Pre-conditions**:

The manager must be logged in.

Bookings must exist in the system.

**Post-conditions**:

Booking details are displayed.

**10. Add Rooms**

**Actor**: Manager

**Description**: Permits the manager to add new rooms to the system.

**Pre-conditions**:

The manager must be logged in.

**Post-conditions**:

The new room details are saved in the system.

**11. Add Hotels**

**Actor**: Manager

**Description**: Allows the manager to add new hotels to the system.

**Pre-conditions**:

The manager must be logged in.

**Post-conditions**:

The new hotel details are saved in the system.

**12. View Profiles**

**Actor**: Managers

**Description**: Enables the manager to view customer and staff profiles for administrative purposes.

**Pre-conditions**:

The manager must be logged in.

Profiles must exist in the system.

**Post-conditions**:

The profile details are displayed.

5.3.1 Class Diagram

A diagram of a computer

Description automatically generated with medium confidence

5.4.1. Sequence Diagram

for Booking a Reservation

A diagram of a book

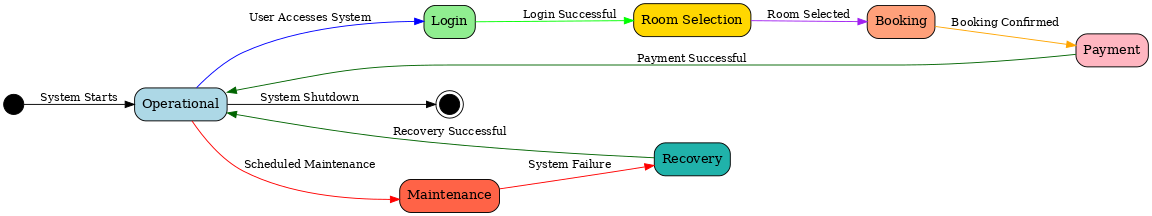
Description automatically generated

5.4.2. Sequence Diagram for Remove Booking

A diagram of a system

Description automatically generated

State diagram 5.5

The state diagram illustrates the flow of the Hotel Management System from initialization to completion. It starts with the System Initialization, transitions through states like Login, Room Selection, Booking, and Payment, and ends with the user logging out. Non-functional states like Maintenance and Recovery ensure system reliability during failures. This structured flow ensures smooth transitions, secure interactions, and system continuity.

Activity diagram 5.6

The activity diagram showcases the workflow of the Hotel Management System, beginning with user login, followed by Room Selection, Booking Confirmation, and Payment. Users can also manage their profiles, while non-functional activities like Scheduled Maintenance and System Recovery ensure operational reliability. The flow ensures a seamless user experience and robust system performance.

A diagram of a system recovery

Description automatically generated

Context model 5.7

The context diagram depicts the Hotel Management System as the central component interacting with external entities such as Users, Receptionists, Managers, a Payment Gateway, and the Database. Users perform tasks like logging in and booking rooms through the Frontend, while the Backend manages logic, data storage, and service integration. The Maintenance Team ensures system reliability and recovery during downtime, creating a cohesive and efficient ecosystem.

A blue oval with black text

Description automatically generated

5.8 Architectural model

Customers, receptionists, and managers are all integrated into the Hotel Management System architecture through a responsive Frontend and a strong Backend that manages booking, payments, authentication, and room selection. With a Payment Gateway for transactions and an Authentication Service for access control, data is safely kept in the database. Within five hours, the Maintenance & Recovery System guarantees dependability, system uptime, and recoverability. This design successfully satisfies both functional and non-functional needs while guaranteeing seamless user engagement, safe operations, and round-the-clock availability.

A diagram of a software process

Description automatically generated

**6.Wireframes**

**Wireframes are designed for:**

**Sign-Up and log-In pages.-1**

**User profile, displaying the rest of the user info**A screenshot of a computer

Description automatically generated

**A screenshot of a hotel

Description automatically generated**

home page-3A screenshot of a computer

Description automatically generated

check Availability-5

BOOK NOW-6A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Payment -7

A screenshot of a computer

Description automatically generated

Booking History-8

Test Cases -7

A white sheet of paper with text

Description automatically generated

Test Case 1: User Signup (Positive Test)

Test Case ID: TC01

Title: User Signup

Description: Verify the system allows a user to successfully sign up using valid credentials.

Input: Valid email address and password

Expected Result: Account is created and user is redirected to login page

Test Case 2: User Login (Positive Test)

Test Case ID: TC02

Title: User Login

Description: Verify the system authenticates a user with correct login credentials.

Input: Registered email and correct password

Expected Result: User is redirected to the user dashboard

Test Case 3: Room Booking (Positive Test)

Test Case ID: TC03

Title: Room Booking

Description: Verify that a logged-in user can book a room.

Input: Check-in/out dates, room type, number of children

Expected Result: Room is reserved and confirmation message is shown

Test Case 4: View Booking History (Positive Test)

Test Case ID: TC04

Title: View Bookings

Description: Verify that a user can view their existing and past reservations.

Input: Logged-in user clicks “My Bookings”

Expected Result: Booking history is displayed

Test Case 5: Invalid Login Attempt (Negative Test)

Test Case ID: TC05

Title: Invalid Login

Description: Verify that an error is shown when logging in with incorrect credentials.

Input: Correct email, incorrect password

Expected Result: "Invalid login credentials" error message appears

**Observer Pattern**-8

1Class Diagram:

A diagram of a function

Description automatically generated with medium confidence

Sequence Diagram:-2

A diagram of a process

Description automatically generated

**Adapter Pattern**-9

Class Diagram:-1

A close-up of a computer

Description automatically generated

Sequence Diagram-2A diagram of a project

Description automatically generated

Singleton Pattern-10

Class Diagram:-1

A person with a name tag

Description automatically generated with medium confidence

Sequence Diagram:-2

A diagram of a software flowchart

Description automatically generated

11. Risk and Security Requirements

11.1 Risk Identification

| **Risk ID** | **Category** | **Description** |
| --- | --- | --- |
| R1 | Security | Unauthorized access to admin features |
| R2 | Technical | Server downtime due to high load |
| R3 | Data Integrity | Booking information gets lost/corrupted |
| R4 | Usability | User confusion due to unclear UI |
| R5 | Legal/Privacy | Exposure of user data without consent |

11.2 Risk Assessment

| **Risk ID** | **Severity** | **Likelihood** | **Classification** |
| --- | --- | --- | --- |
| R1 | High | Medium | Critical |
| R2 | Medium | High | Major |
| R3 | High | Low | Major |
| R4 | Medium | Medium | Moderate |
| R5 | High | Low | Critical |

11.3 Mitigation Techniques

| **Risk ID** | **Mitigation Strategy** |
| --- | --- |
| R1 | Use role-based authentication and strong encryption |
| R2 | Implement load balancing and auto-scaling to handle traffic |
| R3 | Regular database backups and data validation |
| R4 | Conduct usability testing and improve UI clarity |
| R5 | Apply GDPR-compliant practices and encrypt sensitive user data |

### 11.4 Requirement Refinement

#### 🔹 Revised Functional Requirements:

* Only the hotel manager can add or delete hotel rooms.
* Users must verify their email before logging in for the first time.

#### 🔹 Revised Non-Functional Requirements:

* All user data must be transmitted over HTTPS.
* Users should be automatically logged out after 10 minutes of inactivity.
* The system must log all login attempts for auditing and security review.

12.apandex

GitHub Link:-2

A screenshot of a computer

Description automatically generated<https://github.com/badran24>