

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Issue** | **Description** | **Authors** |
| 02/03/2020 | Log In | The log in system is working with normal java authentication. It still is not working with spring security. Needs more time to figure it out. | Daniel, Kalieb, Mihreteab, Sibtain |
|  |  |  |  |

Table of Contents

[1. Introduction 4](file:///A:\Mihre%20MUM%20Courses\CS545-WAA(Web%20Application%20Architecture)\Project%20Documentation\Final%20project\BLUE%20BANK%20SPECIFICATION%20v2.doc#_Toc291443302)

[1.1 Purpose 4](file:///A:\Mihre%20MUM%20Courses\CS545-WAA(Web%20Application%20Architecture)\Project%20Documentation\Final%20project\BLUE%20BANK%20SPECIFICATION%20v2.doc#_Toc291443303)

[1.2 System Description………………………………….4](file:///A:\Mihre%20MUM%20Courses\CS545-WAA(Web%20Application%20Architecture)\Project%20Documentation\Final%20project\BLUE%20BANK%20SPECIFICATION%20v2.doc#_Toc291443303)

1.3 Accessing the design nsystem……………………......5

1.4 Hotel online booking features…………………….....5

[2. Requirements - Use-Case – Usage Scenarios 6](file:///A:\Mihre%20MUM%20Courses\CS545-WAA(Web%20Application%20Architecture)\Project%20Documentation\Final%20project\BLUE%20BANK%20SPECIFICATION%20v2.doc#_Toc291443304)

[2.1 System users 6](file:///A:\Mihre%20MUM%20Courses\CS545-WAA(Web%20Application%20Architecture)\Project%20Documentation\Final%20project\BLUE%20BANK%20SPECIFICATION%20v2.doc#_Toc291443304)

[2.2 Requirements 6](file:///A:\Mihre%20MUM%20Courses\CS545-WAA(Web%20Application%20Architecture)\Project%20Documentation\Final%20project\BLUE%20BANK%20SPECIFICATION%20v2.doc#_Toc291443304)

[2.3 Funcrional requirements 7](file:///A:\Mihre%20MUM%20Courses\CS545-WAA(Web%20Application%20Architecture)\Project%20Documentation\Final%20project\BLUE%20BANK%20SPECIFICATION%20v2.doc#_Toc291443304)

[3. High Level Design 9](file:///A:\Mihre%20MUM%20Courses\CS545-WAA(Web%20Application%20Architecture)\Project%20Documentation\Final%20project\BLUE%20BANK%20SPECIFICATION%20v2.doc#_Toc291443305)

[4. Detailed Design 11](file:///A:\Mihre%20MUM%20Courses\CS545-WAA(Web%20Application%20Architecture)\Project%20Documentation\Final%20project\BLUE%20BANK%20SPECIFICATION%20v2.doc#_Toc291443306)

[4.1 Class diagram 12](file:///A:\Mihre%20MUM%20Courses\CS545-WAA(Web%20Application%20Architecture)\Project%20Documentation\Final%20project\BLUE%20BANK%20SPECIFICATION%20v2.doc#_Toc291443307)

[4.2 Interaction Diagram 12](file:///A:\Mihre%20MUM%20Courses\CS545-WAA(Web%20Application%20Architecture)\Project%20Documentation\Final%20project\BLUE%20BANK%20SPECIFICATION%20v2.doc#_Toc291443308)

[5. Issues, Risk and Dependencies 14](file:///A:\Mihre%20MUM%20Courses\CS545-WAA(Web%20Application%20Architecture)\Project%20Documentation\Final%20project\BLUE%20BANK%20SPECIFICATION%20v2.doc#_Toc291443309)

[6. Future Consideration 15](file:///A:\Mihre%20MUM%20Courses\CS545-WAA(Web%20Application%20Architecture)\Project%20Documentation\Final%20project\BLUE%20BANK%20SPECIFICATION%20v2.doc#_Toc291443310)

1. **Introduction**

The Hotel Reservation System is a Web Based Application designed to serve major use cases for a single location hotel that has the need to provide it’s booking services online. The system’s aim is to provide better business management and customer satisfaction by providing the following main services:

* Providing an automated working platform for the employees of the Hotel who will be administering this application with special ROLE based access to manage data resources
* Create an online access for customers so that they can create an account and make(manage) reservations online
* Secure and readily available data storage for all parties with controlled data access

**1.1 Purpose**

The best way to serve customers online is to be able to set visualization of what they want to buy before they do their decision. In our hotel reservation system project, customers will be able to check visually all the services offered while enjoying at home. The convenience of booking a hotel room online without in person checking or telephone calling has a lot of satisfaction and save time and money.

The main purpose of this project is to provide customers an online hotel booking and create a convenient platform for employees or hotel admins to perform management (CRUD) operations.

* 1. **System Description**

The system is a web-based application and users (customer and administrators) will be able to access ROLE based services. The online banking system allows a user to login by checking the credentials for correctness. When credentials are approved, users can perform all operations they intend to do in the application platform. Aa admin has an additional separate platform for operations on top of a customer because an admin has a dual role, the second being a customer.

* 1. **Accessing the design system**
* All web browsers and machine operating systems are supported.
* MYSQL server is used as a database platform.
  1. **Hotel online booking features**

**Customer:**

* Available Rooms access
* Room type access
* Room view access
* Book room

**Admin:**

Manage bookings

* Create room
* View all rooms
* Update all rooms
* Delete rooms

1. **Requirements - Use-Case – Usage Scenarios**

Customers can perform various services provided by the hotel including room reservation, gym, car garage etc. When a customer requests a resource, the system checks for that availability of the resource and adds that resource to the list of available services. At the end of the booking, when all the service are added, the system automatically calculates the cost of the booking. The Managers or admins are concerned with the overall functioning of the system. He/she has the provision to add new services or resources.

**2.1 System Users**

This project is concerned with developing a flexible, open-ended design for hotel management information system that incorporates well-known design patterns and a lucid design style using UML diagrams. Our Hotel Management System consists of two users:

* Administrator (employee)
* Customer (individual)

**Administrator**: Is a system user responsible for managing the online reservation system by applying the CRUD operations.

**Customer**: Is an individual who can perform online booking including Room type selection, Room view (outdoor scene) selection and other additional services like gym and car garage for extra cost.

**2.2 Requirements**

A functional requirement describes what a software system should do. It describes the behavior of the system as it relates to the system's functionality.

while non-functional requirements place constraints on how the system will do so. It elaborates a performance characteristic of the system

In our system design the following are the non-functional requirements

1. **UI (User Interface).** Consistent User interface including the color theme of the hotel.
2. **Security.** The system shall be secure to protect both the hotel and customer information. In this project, we have employed best practice security measures provided by Spring framework.
3. **Internationalization.** The customer base language of the hotel incorporates a global space. English, Amharic, Tigrigna and Urdu are supported by the system application.
4. **Extensibility and Scalability.** Adding more functionality and capacity to accommodate more customers shall be the systems future development.

**2.3 Functional Requirements**

Use cases sign-up, login, book, review, search and payment can be performed by a customer. Use cases such as sign-up, login, book, search and CRUD operations can also be performed by a system administrator or an employee of the hotel.

A close up of text on a white background

Description automatically generated

Figure 1: System Use Case Diagram

A sample use case ***search*** scenario and the flow of events is presented below

|  |  |
| --- | --- |
| **Use Case Name** | **Search** |
| **Intent** | A User searches a hotel room for intending to reserve/book a room |
| **Description** | The system allows customers with accounts to search for available rooms |
| **Actor** | Customer |
| **Preconditions** | A User is logged in and credentials are verified. |
| **Postcondition** | Display Room booking confirmation. |
| **Triggers** | The user selects a room to book for |
| **Main success confirmation scenario** | 1. The user searches rooms using check-in and check-out dates 2. The system populates and presents available rooms 3. The user specifies the filters/requirements 4. The system filters the available rooms that match the filter criteria 5. The user selects a room and asks the system to book it 6. The system calculates the price of stay 7. The system asks the user to enter personal information 8. The user enters personal information 9. The system verifies the personal information 10. The system asks the user to enter credit card information 11. The user enters credit card information 12. The system verifies the credit card information 13. The user asks the system to proceed processing 14. The system persists the booking entry 15. The system populates booking confirmation |
| **Alternative scenario** | **2 Rooms unavailable**  2.1 The system tells the user there are no rooms available  **4 No available rooms**  4.1 The system tells the user there are no rooms available  **9. Personal information is invalid**  9.1. The system notified user of the validation errors  9.2. Resume at step 8  **12. Personal information is invalid**  12.1. The system notified user of the validation errors  12.2. Resume at step **11** |

Table 1: Search Use Case Description

1. High Level Design

The Project follows a Three-Tier architecture:

**Presentation:** This is the user interface of the application that presents the application’s features and data to the user. Presentation layer used in our project is Spring MVC Library.

**Service:** This layer contains the business logic that drives the application’s core functionalities. Like making decisions, calculations, evaluations, and processing the data passing between the other two layers. Business logic and data management encapsulation is performed in this layer which is designed to be presentation layer agnostic.

**Persistence/Repository:** This layer is responsible for interacting with databases to save and restore application data. Hibernate was used to communicate with the relational database.

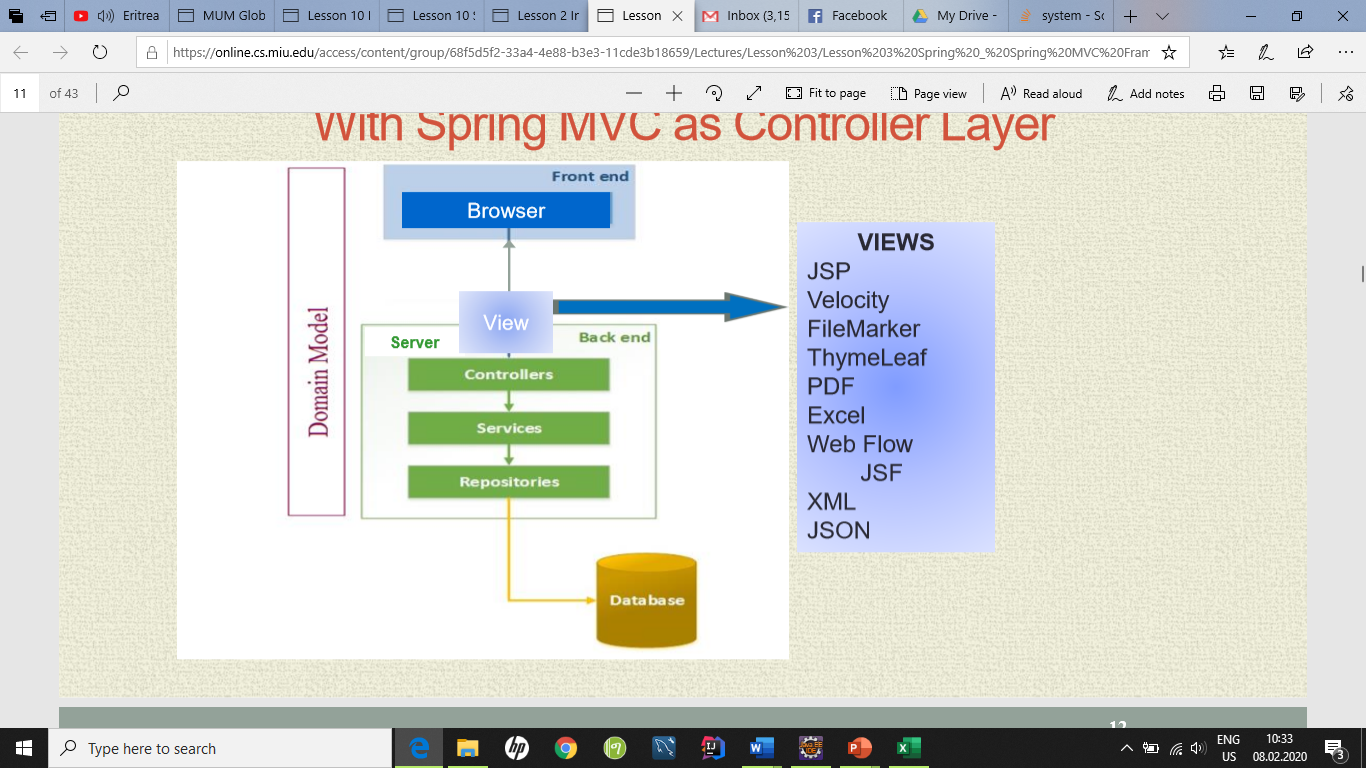


Figure 2: 3-Tier Architecture

**Entities in the project**

|  |  |  |  |
| --- | --- | --- | --- |
| **Entity** | **purpose** | **Role** | **Type** |
| User | Holds user id and names | User info | Entity |
| Address | Holds user address info including user email | User address including email | Entity |
| UserCredientials | Holds passwords and usernames | User verification | Entity |
| Role | Holds Customer or admin | Normal customer or administrator roles | Entity |
| Authority | Holds username and authority | Authority as customer or admin | Entity |
| Room | Holds room id and price | Rooms with their respective price | Entity |
| RoomType | Holds room types | Single, double, deluxe, etc.. | Entity |
| ViewType | Holds view name and price | Outdoor view options | Entity |
| Booking | Holds check-in, check-out, room, price | Room stay dates and extra options for customers like gym and car parking | Entity |
| Review | Holds reviewer info | Reviewers comments, loads image files (room images) | Entity |
| Payment | Holds payment info | All credits card or debit card info including billing and billing address | Entity |

**Table 2 entity description**

1. **Detailed design**

Class diagrams support the speciﬁcation of the concept of classes known from object-oriented programming. The classes below are the major driving classes in our project design.

4.1 **Class Diagram**

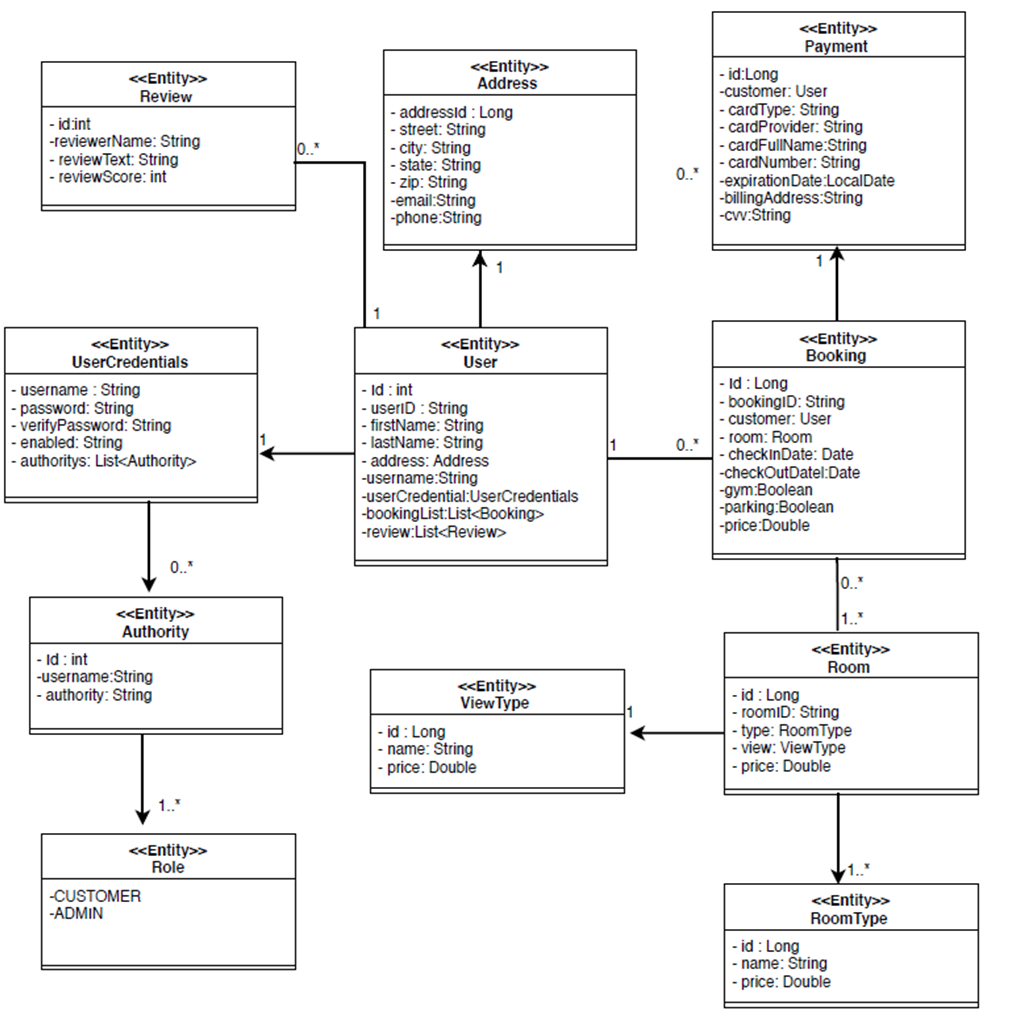


Fig 3. High level design diagram

4.2 **Interaction Diagram**

A sequence diagram simply depicts interaction between objects in a sequential order i.e. the order in which these interactions take place. Dynamic behavior of the system has been modelled for the system using interaction diagrams; specifically sequence diagrams. Figure below depicts the sequence diagram for the use case ***Searching for booking***.

A screenshot of a social media post

Description automatically generated

Figure 4: Sequence Diagram- Search room

A screenshot of a social media post

Description automatically generated

Figure 5: Sequence Diagram- Book Room

1. **Issues and risks**

The main issues that happened during the development of the project were the login and signup. Spring security is still not working properly. It works good in one pc but not on the other due to integration problem. We had challenges in making the customer to manage booking, so far, the challenge to delete their booking was successfully done.

At some point during the project, the team faced some issues related to integrationin which westruggled in doing things again.

1. **Future considerations**

The current system does not allow customers to manage their reservations. One such future consideration would be more convenient for customers to manipulate booking.

Another point that can be added is the functionality of adding pictures in reviews so that reviewers can post pictures if they like.