Irish Interdisciplinary Journal of Science & Research (IIJSR)
Volume 9, Issue 2, Pages 45-55, April-June 2025

Smart Hotel Booking and Management System Using Java MVC Architecture

R. Dhinesh[1*], Dr. T. Ananth kumar[2], Dr. P. Kanimozhi[3] & Sunday Adeola AJAGBE[4]

[1-3]Department of Computer Science and Engineering, IFET College of Engineering, Tamilnadu, India. [4]Department of Computer Engineering, Abiola Ajimobi Technical University, Ibadan, Nigeria. Corresponding Author (R. Dhinesh) Email: dhineshcrazy105@gmail.com[*]

DOI: https://doi.org/10.46759/IIJSR.2025.9207

Article Received: 27 February 2025 *Article Accepted: 29 April 2025* *Article Published: 20 May 2025*

ABSTRACT

In recent times, the hotel industry has become reliant on digital channels in order to improve their operational efficiency, while providing their customers with the best possible experience. To accomplish this, a web-based hotel room search and reservation application to facilitate interaction between the user, the administrator and hotel management was created using Java and Spring Boot. The application provides a way for users to search for hotels, see room details, and book with real-time persistence to the database. The backend of the application is built with the Spring MVC framework and uses integrations with Bootstrap, Thymeleaf, Leaflet and Nominatim APIs in the front-end. Nominatim and Leaflet provided the option for users to search for hotels based on their locations mixed in with a map view as well. MySQL was used for the database to store all user, hotel and booking information. The way the application is structured it somewhat follows proper Model-View-Controller for the web application and has user-role based access to limit and restrict the types of operations users can perform. The solution was designed to help simplify operational efficiency and provide hotel managers with an easy to use tool to manage hotel reservations.

Keywords: Hotel Room Booking; Java Web Application; Spring Boot; Spring MVC; Thymeleaf; MySQL; Leaflet; Nominatim API; HTML; CSS; Web Development; MVC Architecture.

1. Introduction

The hospitality industry is a significant portion of the world economy and the demand for hotel accommodations appears to be constantly rising as travel and tourism increase. As the hospitality industry grows, so does the need for efficient methods to facilitate hotel booking processes for hotel stakeholders that ensure smooth operation and provide consumer satisfaction [1]. Unfortunately, the accommodation segment of the industry continues to operate using the manual reservation method, with distinct disadvantages inherent to the method. Issues such as double booking, lost bookings, and administrative mistakes are indicative of the various complications facing hotels regarding room reservations. The absence of centralized systems to manage hotel data and customer interactions often lead hotels to depend on spreadsheets and manual systems. These real-life challenges create demand for automated hotel booking and reservations systems that overcome these issues while providing a streamlined, user-friendly approach.

A web-based hotel room booking and reservation system solves the challenges for both the user and the hotel administrator to make the process simple to search for hotels, view room availability, designate room types, and complete secure bookings [2]. Using automation of key booking process such as checking availability, confirming the booking, and validating payment eliminates manual processes and human error and improves operational efficiency, and assumes a fast pace. From the user's perspective, the hotel room booking and reservation system provides an easy platform. The hotel manager and administrator benefit from a user-friendly interface that allows them to manage hotel listings, room inventory, guest information, and booking details on one convenient platform. Automating these processes streamlines hotel activities and improves the accuracy of room availability, booking information and data, improving the overall experience for customers [3].

The system described in this dissertation was developed using Java and Spring Boot. Java and Spring Boot are extensively used to develop reliable, flexible and scalable enterprise-class applications. The application architecture also follows Model-View-Controller (MVC) patterns by separating concerns between the data, business logic and user interface, making it easier to develop modular and maintainable applications. The backend of the developed system leverages Spring Boot's ability to provide ease of development through out-of-the-box tools and configurations. Spring MVC acts as the layer that accepts requests and produces responses by routing requests to appropriate controllers, allowing a seamless interface accomplished through the combination of front-end and back-end components.

On the frontend component, [4] the system leverages Thymeleaf, a modern templating engine that works smoothly with Spring Boot, to dynamically generate HTML content. It uses Bootstrap, a popular front-end framework, to enable the system to be responsive and mobile-friendly, providing an effective layout for different types of displays when booking hotels over mobile, tablets, and desktops. Users can access the booking platform making decisions on the front-end platform when using a computer, tablet, and/or smart phone without losing quality of user experience. The system incorporates Leaflet, an open-source JavaScript library to deliver maps showing the location of hotels in an interactive manner [5]. Users are enabled to see where hotels are located using the maps as well as by using the Nominatim API to provide geographical searches to find hotels based on their location preferences.

This web-based hotel booking system aims to maximize the efficiency and user experience for hotel bookings. The entire hotel booking process is completed electronically, meaning that hotels will not have to enter information manually and will save hours in human resources, errors will be minimal, and efficiency will be maximized [6]. Additionally, booking users will have a more simplified and intuitive experience when booking their hotel, and instead of worrying about forms and paper work, they will make quick and secure bookings directly from their own device [7]. As online booking continues to expand in property segment, the development of online booking systems will become a key driver of digital transformation in the hospitality industry.

With the right technology and understanding the importance of user-centered design, this system was developed to satisfy the needs of hotel management and hotels as it continues to evolve. This system, built from Java, Spring boot, Thymeleaf, and MySQL, has taken several steps toward modernizing hotel reservation systems and improving the overall guest experience.

1.1. Study Objectives

The study is conducted with the following objectives:

a) To eliminate the manual booking process for hotels to reduce error and lag time associated with this process.

b) To develop a user, hotel manager, and admin interface that is easy to use and portable.

c) To implement role-based authentication and access control on various user levels.

d) To allow hotel managers to maintain all hotel information, room availability, and to keep booking history.

e) To allow customers to explore, filter, sort and complete a hotel room booking in a secure way.

f) To remain consistent, scalable, and maintain modularity, by using modern tools that facilitates full-stack development.

2. Related Works

2.1. Traditional Hotel Booking Systems

Historically, hotel booking systems were primarily manual, with clients needing to call, fax, or visit an establishment to secure a reservation. As a result, the opportunity for human error was quite prevalent [8]. Most of the time, the process would be cumbersome, and there was no central data management process to reference room availability across various physical locations. Double-bookings, missed reservations, and other administrative mishaps were legitimate threats. Additionally, most hotel booking systems were regionalized, only serving a handful of hotels or destinations, rarely able to expand or keep pace with and/or take advantage of rising demand, and rarely offering a seamless user experience [9]. While the traditional booking system served a purpose, they were inefficient, tedious, and simply did not meet the demands of the hospitality market.

2.2. Online Hotel Booking Systems

Following the rise of the internet in the 1990s, the hospitality industry embraced these new opportunities and access to technology, launching online hotel booking systems. Travel booking platforms like Expedia and Booking.com quickly changed how we booked hotels, allowing searches, checking of room availability and booking rooms all online. These systems automated the reservation process, meaning no need for phone or in-person discussions. However, early online hotel booking systems were basic [10]. Hotels could post static availability for example. There was very little filtering, and while you could book in real time, not all hotel systems allowed for this.

Barua and Kaiser [11] propose in their study a proposal with more sophisticated real time database capabilities and directly linked to booking status as well. They structured the database system to assess room availability and booking status automatically. While this eliminates many booking disputes - some earlier linked very basic and rudimentary interface - really has no or unacceptable filtering or mapping functionality (input real time maps), two prominent characteristics of today's well designed user experiences.

2.3. Integration of Interactive Maps in Hotel Booking Systems

Interactive maps are now an important option in the hotel booking process. Lee et al. (2024) [12] were the first to see the potential of Google Maps as a hotel booking medium, allowing customers to look at hotels on a map and filter results based on proximity to other locations [13]. The mapping feature improved user experience by allowing them to search hotels geographically. The Google Maps were nice, but their usage raised privacy issues with respect to users being tracked, etc. [14].

As a result of privacy concerns, many hotel booking systems now use Leaflet and Nominatim API instead of Google Maps. These mapping interfaces are open-source, and developers have more control over their map data while also ensuring the privacy of their users. Leaflet, for example, is lightweight, and users can tailor their maps to the specific needs of their platform [15]. Location-based services by using interactive maps has now become a standard in the hotel booking space, allowing for a more natural and customized search for hotels by users.

2.4. Limitations of Existing Work

The security of online payments and user data [16]. Even with SSL encryption and payment gateway integration being standard practice today, many hotel booking platforms are still vulnerable to data breaches as well as fraud. Spring Security has made strides in protecting these types of systems, however, even when platforms use Spring Security, not all of them stay up to date with the latest security protocols [17]. In addition to this, with the digital transformation of the industry towards more mobile technology, ensuring that booking systems have full mobile optimization and secure mobile payment options, has also been a challenge [18].

Lastly, scalability is an ongoing challenge. When a hotel expands its presence globally, the system is required to provide the capability of an increasing number of rooms, users, and bookings [19]. Some of the platform systems in place today are unable to scale appropriately and efficiently, and larger systems specifically that leverage an older technology stack or legacy database, can obviously not scale. Additionally, some of these systems are not suited for dealing with a large number of rooms, users, or bookings, especially when rapid, real time data transfer is required.

The [20] limitations mentioned above suggest that there is a need for a more advanced and scalable solution that also has security built in, and provide real time synchronization, personalized experience, and increased user engagement. The current work reported on in this paper seeks to fill these gaps, by implementing the latest technology while working to create a robust, secure, user centric platform which caters to the rapidly changing needs of the hospitality industry.

3. Proposed methodology

The hotel room booking and reservation system we propose is a systematic web application focusing on usability, performance, and scalability. The user can easily conduct a search for hotels, view available rooms and make reservations through a clear interface. The system architecture employs a Model-View-Controller (MVC) approach, which provides methods for component-based development.

3.1. System Architecture

The System Architecture follows the Model-View-Controller (MVC) architectural pattern to help reduce coding and easier maintenance:

- Model: The data and business logic representing the application as a whole. This may include the entities such as Users, Hotels, Rooms and Bookings - which will be transcribed to the underlying relational database via object relational mapping.

- View: The typical front-end user interface; which may include globally dynamic web pages (using server-side templating). This layer accounts for all visual presentation and user interaction.

- Controller: The controller drives incoming requests and processes the request, then selects the right views. It handles various potential states, ensuring underlying business rules are being followed and data is being processed correctly.

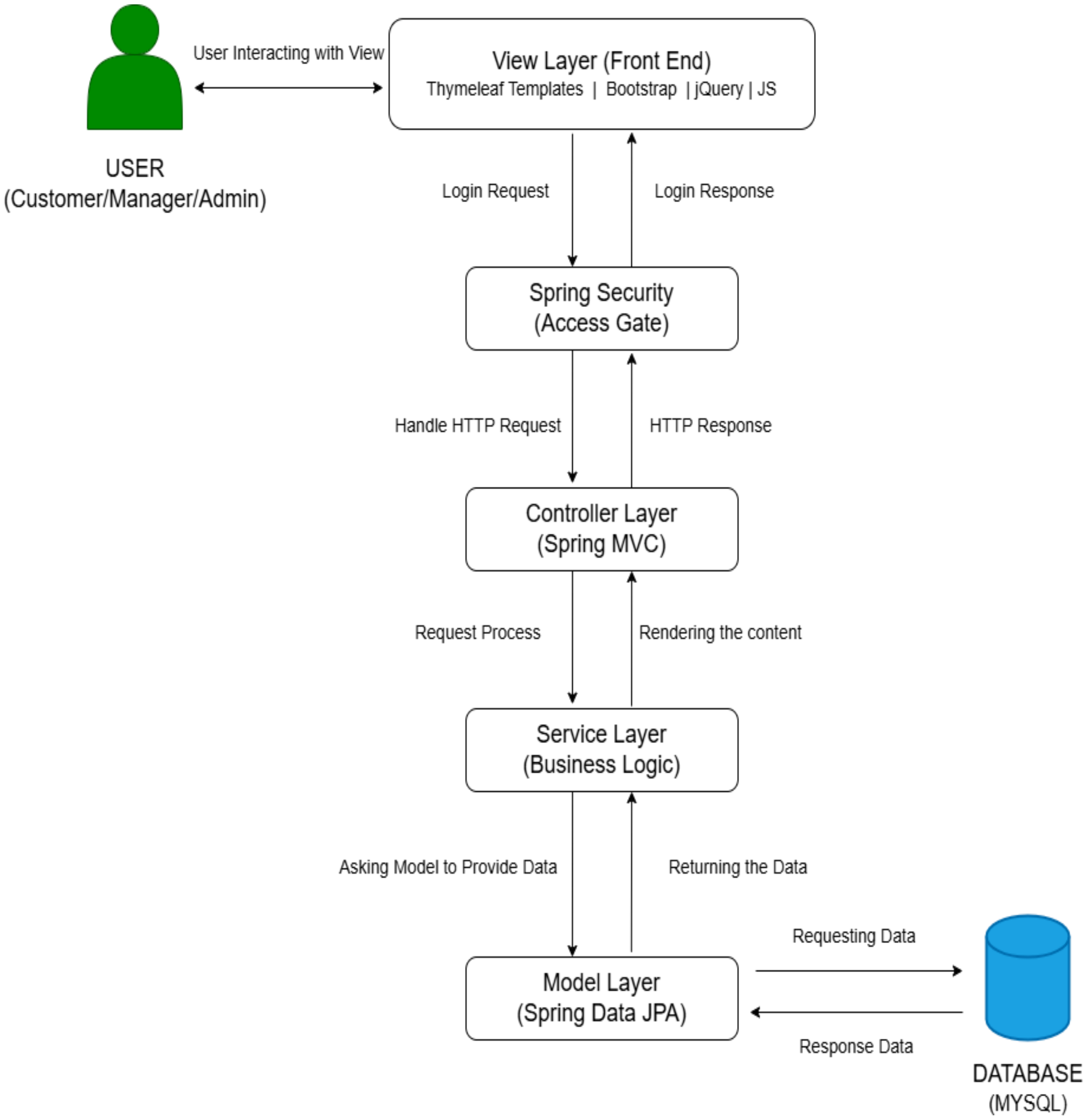


Figure 1. MVC Architecture

Figure 1 shows the Architecture diagram of Smart Hotel Booking and Management System. The application also follows an organization of unforeseen three layers:

- Presentation Layer: The layer that user interacts with, which allows them to capture input, and retrieve back their results.

- Application Layer: The layer that holds the core business logic and service classes that define the operations of the system.

- Persistence Layer: The layer that holds all the actual interaction with the sqlite database via repository interfaces and object-relational mapping.

This architecture provides a way to modularize and therefore achieve testibility and therefore scalability. All necessary to continue to future enhancements.

3.2. Key Modules and Functionalities

The key modules and functionalities are as follows:

- User Management: The system allows management of user registrations, logins, and a basic role-based access system. There are two types of users; regular users who can make bookings and administrators who can manage hotels and bookings.

- Hotel and Room Management: Administrators will have an accessible dashboard where hotel entries can be created, modified, and deleted, in addition to room type and rates. The dashboard has form validations in place to restrict incomplete or incorrect data inputs.

- Room Booking Process: Users can search hotels, filter based on the user's choice of hotels from multiple choices, and choose your room choices. The booking process requires user input in the form of check-in/check-out dates, and once the user confirms the booking it changes the room's availability in real time.

- Interactive Map Integration with hotel listings: Hotel listings will include an interactive geographic map which will give users a spatial representation of the hotels, while allowing the users to choose their accommodations based on landmarks or areas of interest.

- Generated Booking History and Management: Users are able to view past and current bookings in their profile, whereas administrators would be able to view all user bookings and manage cancellations or modifications anytime.

3.3. Security and Validation

The system ensures secure operations through several mechanisms:

- Encrypted password storage.

- Role-based access restrictions to safeguard administrative functionalities.

- Form validations to prevent input-based vulnerabilities.

- Session control to prevent unauthorized access to protected areas.

All user and admin actions are carefully regulated to maintain data integrity and ensure reliable operations.

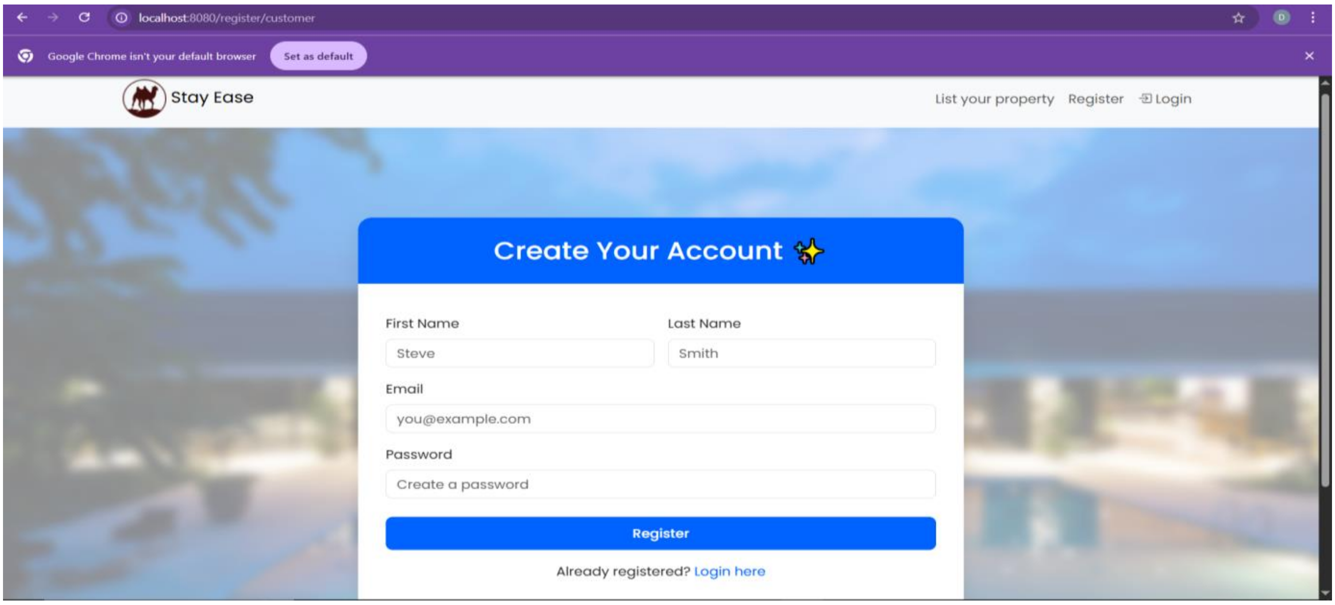
░4. Results and Discussion

The designed hotel room booking and reservation system was successfully deployed and the multiple modules - user management, hotel listing, booking management, and administrator control - were tested with the intention of ensuring comprehensive functionality, intuitive navigation, secured access, and concurrency. The system also fulfilled its functional and non-functional requirements with only slight latency and a high level of accuracy in regards to data processing.

➢ System Functionality Evaluation

The application was rolled out in a local situation using normal web browsers. Testing was carried out on all core functionality of the application either in the role of user or one of the roles mentioned above, these include:

- User registration
- Adding hotels
- Make room bookings
- Views of booking history


localhost:8080/register/customer
Google Chrome isn't your default browser
Set as default
Stay Ease
List your property
Register
Login

Create Your Account ✨

First Name: Steve

Last Name: Smith

Email: you@example.com

Password: Create a password

Register
Already registered? Login here

Figure 2. User Registration

Figure 2 output was consistently produced correctly and invalid & incomplete submissions were appropriately accepted or rejected at all times.

Once the application was released the existing users could register, login and access appropriate features based on role. When bookings were made these reflected immediately in the database, the user on the registering end had their hotel choices available to them and the bookings made at that time. The map-based interface improved the user decision-making process, given they are spatially aware of the hotel location in relation to others listed.

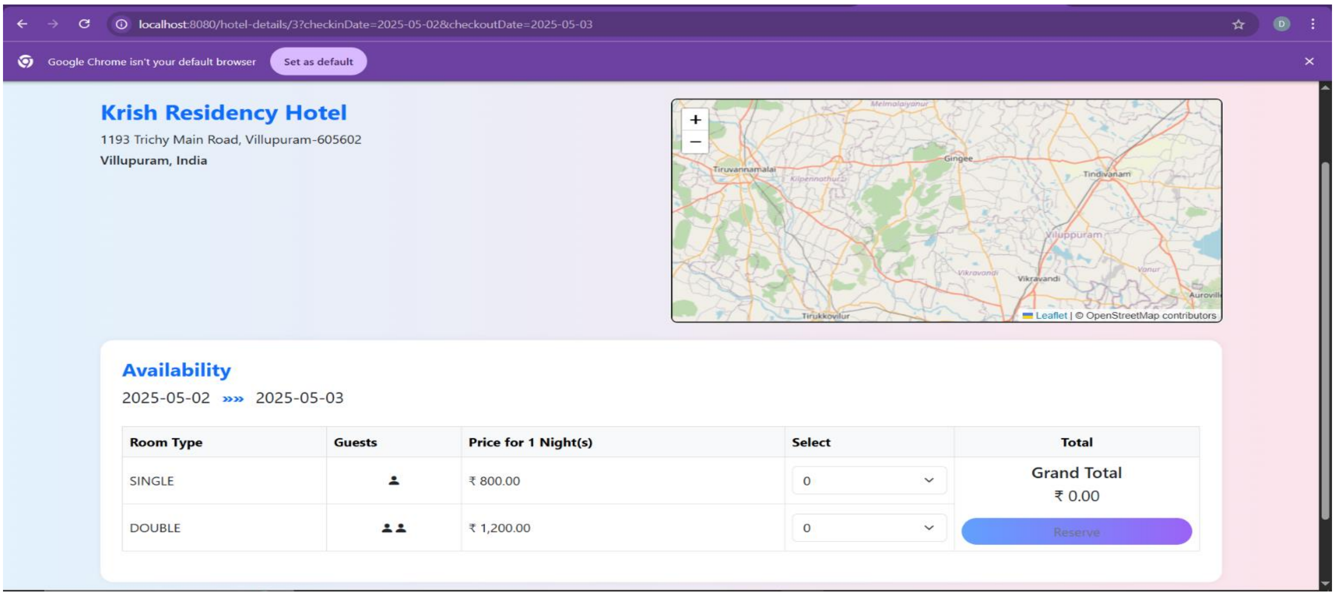
➢ User Experience and Interface Feedback

The frontend design was assessed against certain responsivity and usability criteria. The system was capable of managing various form factors, with pages adapting fluidly from desktop and laptop to tablet and mobile devices. The product's card-like presentation format, combined with an interactive map, made it easier for users to browse for hotels.

The user feedback, obtained through informal usability testing with several users, suggested that the system was intuitive and also easy to use. Users also commented positively on the relatively simple path to a booking that presented users with clear data and also looked refreshing. Administrative users noted the ability for easy management of hotels and bookings through the backend dashboard.

➢ Performance and Reliability

The system response time remained stable under normal use cases. You were able to examine the database calls and the page load times, and the functions were performing within an acceptable range for normal booking actions and simultaneous users. In the back-end services were performing practical tasks to support booking confirmations, information retrieval, and form validations.


localhost:8080/hotel-details/3?checkinDate=2025-05-02&checkoutDate=2025-05-03
Google Chrome isn't your default browser
Set as default
Melmalaiyanur
Tiruvannamalai
Kilpennathur
Gingee
Tindivanam
Viluppuram
Vikravandi
Vikravandi
Vanur
Auroville
Tirukkovilur
Leaflet | © OpenStreetMap contributors

Krish Residency Hotel
1193 Trichy Main Road, Villupuram-605602
Villupuram, India

Availability
2025-05-02 »»» 2025-05-03

Room Type	Guests	Price for 1 Night(s)	Select	Total
SINGLE	👤	₹ 800.00	0	Grand Total ₹ 0.00
DOUBLE	👤👤	₹ 1,200.00	0	Reserve

Figure 3. User Hotel Booking Page

Figure 3 is system maintained stability, with no crashes to the service or data loss. To help avoid over-booking, the system checked availability in real time in advance of confirming bookings, which showed that the data handling component performed as requested.

➢ Security and Data Integrity

The security features were evaluated through functional validation of authentication/authorization, along with a manual examination of access controls. Access to admin features by unintended users was successfully prevented using role-based access controls. Passwords were securely stored, and critical operations, including removing hotels and altering bookings, were restricted to users who authenticated as an administrator.

Form fields were tested for injection vulnerabilities, and the input sanitization mechanisms functioned as expected. Data integrity was ensured; for example, through controlling duplicate bookings and validating date fields before booking confirmation.

➢ Future Scope

While the current version meets the basic functionality of a hotel booking system, growth in further functionality remains an option for development in the future. Significant enhancements include a secure online payment gateway to allow users to complete their transactions using digital payments directly within the system. Moreover, incorporating a dynamic pricing model to account for seasonal trends, room availability, and other user demand considerations will improve revenue management for the hotel.

A mobile application version of the system would improve accessibility for users to be able to book hotels using their mobile phones. Allowing for multiple language support can also allow for greater reach of the platform to foreign, and widespread user accessibility. Finally, the addition of a customer product review and star rating system can further aide in transparency and allowing users to better decide using the past experiences of guests.

The future improvements discussed would optimize the system to a fully featured commercial-grade hotel reservation system that would be applicable for wide scale deployment in the hospitality industry.

░5. Conclusion and Future Work

The System provides a holistic solution to the challenges of outdated booking systems by facilitating a clear user-friendly, efficient, and secure method of booking hotels. The application was developed using the Model-View-Controller design, which promotes separation of concerns, thus simplifying development and maintenance. The application allows users to search and view hotels, verify room availability, create reservations, and view their previous bookings, while administrators can view and manipulate hotel data and keep track of user activity. The application executes instantaneous changes, validates user inputs, and maintains role-based access control mechanisms to ensure data integrity and security regardless of user engagement levels. The interactive map-based hotel listings and responsive frontend enhance the usability of the system, facilitating easy navigation and selection across all devices and screen sizes. The system has shown to be highly reliable under different test scenarios and was consistently performing throughout initial testing of normal usage. Multiple layers of user role management, secure login mechanisms, and a structured database design demonstrate stability and robustness of the application. The system achieves objective status of digitizing hotel room booking processes in a scalable and user-focused manner.

While the system's functionality is completed and it is fully functional, there are possibilities to enhance its capabilities in future versions. For example: adding real-time payment gateways, giving hotel ratings and reviews, sending emails, enabling a mobile app and many more features would greatly enhance a user's experience and engagement with the system. Including multi-language support, dynamic pricing and analytics dashboards could make the system easily deployable commercially. These future improvements would help maximize the reach and capability of the system, allowing transformation from an academic project to a full-featured, production-ready hotel booking platform.

Declarations

Source of Funding

This study did not receive any grant from funding agencies in the public, commercial, or not-for-profit sectors.

Competing Interests Statement

The authors declare that they have no conflict of interest.

Consent for publication

The authors declare that they consented to the publication of this study.

Authors' contributions

All the authors took part in literature review, analysis, and manuscript writing equally.

References

[1] Barua, B., & Kaiser, M.S. (2024). Microservices-Based Framework for Predictive Analytics and Real-time Performance Enhancement in Travel Reservation Systems. ArXiv Preprint ArXiv: 2412.15616. https://arxiv.org/abs/2412.15616.

[2] Hotel and Hospitality Management Software Market Size & Share by 2033, Straits Research (2025). https://straitsresearch.com/report/hotel-and-hospitality-management-software-market.

[3] Aravani, L., Pintelas, E., Pierrakeas, C., & Pintelas, P. (2024). A Natural Language Processing Framework for Hotel Recommendation Based on Users' Text Reviews. ArXiv Preprint ArXiv: 2408.00716. https://arxiv.org/abs/2408.00716.

[4] Jishan, M.A., Singh, V., Ghosh, A.K., Alam, M.S., Mahmud, K.R., & Paul, B. (2024). Hotel Booking Cancellation Prediction Using Applied Bayesian Models. ArXiv Preprint ArXiv: 2410.16406. https://arxiv.org/abs/2410.16406.

[5] Valentine, M. (2025). Hospitality Software Trends Shaping the Industry in 2025. Sabre Hospitality. https://www.sabrehospitality.com/resources/article/hospitality-software-trends-shaping-the-industry-in-2025/.

[6] Hotel Management Software Market Share, Growth &Trends (2025). Polaris Market Research. https://www.polarismarketresearch.com/industry-analysis/hotel-management-software-market.

[7] "Hotel and Hospitality Management Software Market Report 2025, Analysis and Forecast," The Business Research Company (2025). https://www.thebusinessresearchcompany.com/report/hotel-and-hospitality-management-software-global-market-report.

[8] Mehataj, M., Kanimozhi, P., & Kumar, T.A. (2025). Social Media and Its Influence on Travel Decision-Making. In Decoding Tourist Behavior in the Digital Era: Insights for Effective Marketing, Pages 135–166, IGI Global.

[9] Exploration of Hotel Reservation through Mobile Online Travel Agencies (2024). International Journal of Tourism Research. https://onlinelibrary.wiley.com/doi/10.1002/jtr.2734.

[10] Hotel Management Software Market to Reach $5.9 Billion, Globally, by 2032 at 5.9% CAGR Globe Newswire (2024). https://www.globenewswire.com/en/news-release/2024/08/01/2922468/0/en/Hotel-Management-Software-Market-to-Reach-5-9-Billion-Globally-by-2032-at-5-9-CAGR-Allied-Market-Research.html.

[11] Barua, B., & Kaiser, M.S. (2024). Real-Time Performance Optimization of Travel Reservation System Using AI and Microservices. ArXiv Preprint ArXiv: 2412.06874 https://arxiv.org/abs/2412.06874.

[12] Lee, T., & Han, Y. (2024). An Intelligent Hotel Reservation System Using Machine Learning Algorithms for Personalized Services. Journal of Intelligent Systems, 29(6): 889–897.

[13] InnQuest (2025). The Impact of AI on Hotel Management. https://www.innquest.com/blog/the-impact-of-ai-on-hotel-management-what-to-expect-in-2025/.

[14] Condé Nast Traveler (2023). Bright Ideas in Travel 2023. https://www.cntraveler.com/story/bright-ideas-in-travel-2023.

[15] Kumar, R., & Sharma, A. (2024). AI-Driven Hotel Booking Systems: Future Trends and Impacts on the Hospitality Industry. International Journal of Hospitality and Tourism Management, 10(4): 211–220.

[16] Smart Hotel Technology Trends to Keep an Eye on for 2025. Hotel Management Network (2025). https://www.hotelmanagement-network.com/features/smart-hotel-technology-trends-to-keep-an-eye-on-for-2025.

[17] Wang, L., Verma, R.S., & Kumar, A. (2023). IoT-Based Hotel Management System: A Review and Future Directions. International Journal of Information Technology and Web Engineering, 19(2): 83–96.

[18] Wilson, A.G.B., & Patel, S.K. (2025). Implementation of Blockchain for Secure Hotel Reservation Systems. Journal of Computer Applications in Engineering, 40(1): 129–137.

[19] Taylor, J.M. (2025). Hotel booking system performance optimization through big data analytics. International Journal of Big Data and Cloud Computing, 13(1): 44–57.

[20] Gupta, R.S., Sharma, M.S., & Reddy, P.S.V.R. (2023). Integration of Cloud Computing and Artificial Intelligence in Hotel Management. Journal of Cloud Computing Applications, 7(2): 88–96.