# Cryptography: Implementation of Hashing function for Presto Hotel Booking System

Muhammad Muaazin Bin Abdullah
*Faculty of Computer Science and Information Technology*
*University Tun Husseein Onn Malaysia*
Batu Pahat, Johor, Malaysia
ci220133@student.uthm.edu.my

*Faculty of Computer Science and Information Technology*
*University Tun Husseein Onn Malaysia*
Batu Pahat, Johor, Malaysia
ci220099@student.uthm.edu.my

Nur Fatin Nazifa Binti Mohd Nazlan
*Faculty of Computer Science and Information Technology*
*University Tun Husseein Onn Malaysia*
Batu Pahat, Johor, Malaysia
ci220011@student.uthm.edu.m

Nurul Ummi Shahirah Binti Mohd Saifuddin

Nurul Nadia Binti Abd Wahab
*Faculty of Computer Science and Information Technology*
*University Tun Husseein Onn Malaysia*
Batu Pahat, Johor, Malaysia
ci220060@student.uthm.edu.my

***Abstract*— In response to the increasing demand for a more sophisticated and secure hotel booking experience, this project aims to develop a cutting-edge Hotel Booking System. Our focus lies in creating a user-friendly interface that enhances the booking process for travelers while streamlining operations for hotel administrators. Key features include robust data management using MySQL and stringent security measures such as SHA256 hashed passwords to protect guest information. By prioritizing efficiency and reliability, we seek to redefine industry standards and provide a seamless booking experience that meets the expectations of modern travelers and hospitality professionals alike. This abstract encapsulates our commitment to innovation and excellence in hospitality technology. (***Abstract**)***

***Keywords—SHA256, system, decrypt, 2FA** (key words)*

## I. Introduction

Welcome to a new era in hotel booking, where sophistication meets user-friendliness and security reigns supreme. Our Hotel Booking System project emerges as a response to the evolving needs of travelers and the hospitality industry alike. With a vision to revolutionize the booking experience, we prioritize creating a seamless interface that caters to modern travelers' expectations while simplifying operations for hotel administrators behind the scenes. Utilizing robust technologies like MySQL for data management and SHA256 hashed passwords for security, we ensure reliability and safeguard guests' sensitive information. This project is not merely about developing a booking system; it's a transformative journey aimed at setting new standards in hospitality, where efficiency, security, and user-centric design converge to redefine the entire booking process. Welcome to the future of hotel booking, where simplicity meets sophistication to elevate your experience as a guest or hotel administrator.

### A. Project Background

In response to the evolving landscape of travel and hospitality, where traditional booking methods no longer suffice, we embark on a transformative journey with our Hotel Booking System. Designed to meet the sophisticated demands of modern travelers and streamline operations for hotel administrators, our system prioritizes both user experience and data security. Utilizing MySQL for robust data management and employing SHA256 hashed passwords to safeguard guest information, we ensure reliability and security at every interaction. Beyond functionality, our vision extends to redefining the booking experience, setting new standards of excellence in the industry. Welcome to the future of hotel booking, where simplicity meets sophistication to create an immersive journey for guests and hotel owners alike.

### B. Problem Statement

The current landscape of hotel booking systems is plagued by outdated methods and inefficiencies that hinder both guests and administrators. Relying on cumbersome paper forms consumes excessive time and resources, leading to errors and frustrations for all parties involved. Existing systems often suffer from complexity, lacking user-friendly interfaces that discourage potential customers from completing bookings smoothly. Moreover, inadequate administrative tools make it challenging for hotel staff to manage reservations effectively and provide personalized service, ultimately compromising the guest experience. Addressing these issues is crucial for ushering in a new era of streamlined, user-centric hotel booking solutions.

### C. Objective

The objective of our Hotel Booking System project is centered on three key pillars: designing a User-Friendly interface, developing an Efficient and Secure System, and rigorously testing its Security protocols. With a commitment to enhancing user experience, we prioritize intuitive design that simplifies booking processes. Security is paramount, evidenced by our implementation of SHA256 hashed passwords to protect guest data from unauthorized access. By focusing on efficiency, we aim to streamline operations for hotel administrators, ensuring seamless management of bookings and guest interactions. Through comprehensive security testing, we verify the robustness of our system, safeguarding sensitive information and delivering peace of mind to both guests and hotel owners.

### D. Scope

The scope of our Hotel Booking System project is defined to ensure comprehensive functionality and ease of use for both administrators and users alike. Within the System Scope, functionalities include User Registration, facilitating seamless account creation with required username and SHA256 hashed password entry to uphold security standards. Administrators are empowered with capabilities for Booking Modification, enabling adjustments or cancellations as necessary, ensuring flexibility and customer satisfaction. Under the User Scope, the focus remains on intuitive interactions, allowing users to easily Register an Account adhering to specified password complexities, and Make a Booking effortlessly by providing necessary details like dates and room preferences. This scope underscores our commitment to delivering a robust, secure, and user-friendly platform that redefines the hotel booking experience.

### E. *Flow and Organization*

The flow and organization of our Hotel Booking System are meticulously structured to optimize user experience and operational efficiency. Beginning with User Registration, the process ensures that users can swiftly create accounts with secure SHA256 hashed passwords, adhering to specified complexity requirements. Once registered, users seamlessly navigate to the Booking interface, where they input essential details such as check-in/check-out dates and room preferences in an intuitive, user-friendly manner. Administrators, equipped with robust permissions, oversee Booking Modification, enabling them to manage reservations effectively, whether it's cancelling bookings or making necessary changes. This structured approach not only enhances user satisfaction by simplifying the booking process but also empowers administrators with tools to streamline operations, ensuring a seamless flow from registration to reservation management within our system.

## II. Literature Review

*This section examines current booking systems that are comparable to the suggested safe reservation system. The Eatigo.com restaurant booking system, the Golden Screen Cinema (GSC) ticket booking system, and the Airasia airline booking systems are the system that were examines.*

### A. *The Eatigo.com Restaurant Booking System*

Many restaurants in Malaysia utilize the well-known restaurant reservation platform Eatigo.com. By providing their name, phone number, and email address, customers may book seats. They provide the number of guests in their group as well as their preferred eating hour. The consumer then receives an email with reservation details. Customers must sign up and log in to the website in order to enjoy exclusive offers. To protect user credentials, Eatigo.com employs SHA-1 hashing.

### B. *AirAsia Flight Booking System*

Users of AirAsia's flight booking system can log in with their Google, Facebook, or WeChat accounts, or they can use their email address and password. When a user logs in, the system asks for their phone number as verification. It then sends them an SMS with a One-Time Password (OTP) to authenticate.

### C. *Golden Screen Cinema (GSC) Ticket Booking System*

Customers use their phone number and password to log in if they already have an account. In order to register, new users need to provide their name, email address, password, TAC (Transaction Authorization Code) number, and phone number. In order to authenticate, users must provide both a password and a One-Time Password (OTP), which is usually delivered via email. The system uses Two-Factor Authentication Comparison and Proposed Secure Booking System

*Comparison and Proposed Secure Booking System*

Current booking systems show several methods for protecting user identity and confidential information. They use techniques including Two-Factor Authentication (GSC), email confirmation and SHA-1 hashing (Eatigo.com), and social network login integration (AirAsia). These solutions give consumers ease and flexibility while guaranteeing safe access to their services.

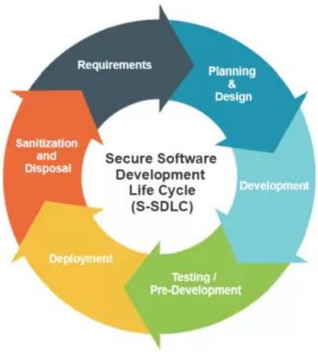
These current systems provide models for the proposed secure booking system, providing guidance for the implementation of comparable security features. Important things to think about would be:

A. User authentication: Putting strong authentication techniques into practice, such hashing passwords (like MD5 or SHA-256), integrating social media for instant logins, and using Two-Factor Authentication for more protection.

B. Data Encryption: Making sure that private user data, such as passwords and personal information, is encrypted both during storage and transmission to prevent unwanted access.

C. User Experience: By providing easy-to-use booking procedures and login choices, we strike a balance between security and user comfort.

## III. Methodology


Requirements
Planning & Design
Development
Testing / Pre-Development
Deployment
Sanitization and Disposal
Secure Software Development Life Cycle (S-SDLC)

For the Hotel Booking System project, a suitable Secure Software Development Lifecycle (SDLC) would involve integrating security considerations seamlessly throughout the development process. Here is how each stage of the Secure SDLC aligns with the project:

### A. *Planning*

Secure Inputs, collate security inputs from stakeholders, ensuring security definitions are comprehensive and integrated alongside functional and non-functional requirements. This includes considerations for securing user data, implementing authentication mechanisms, and protecting against common threats like SQL injection and XSS.

### B. *Development*

Secure by Design, utilize security best practices during product development to ensure that the code is inherently secure by design. This involves implementing secure coding principles and incorporating OWASP secure coding guidelines. Static Code Review, conduct static code reviews in parallel with development to identify and address security vulnerabilities early in the development lifecycle.

### C. *Build*

Monitor Compilation Processes, implement processes to monitor software compilation, ensuring security is assured at this stage too. This involves validating the build environment and configurations to prevent unintended vulnerabilities.

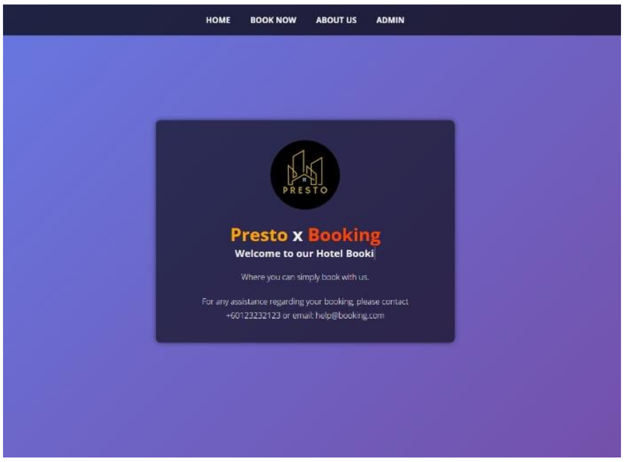
### D. *Testing*

Integrate security testing throughout the development lifecycle, including dynamic application security testing (DAST), static application security testing (SAST), and interactive application security testing (IAST). Automation and Continuous Integration, utilize test automation and continuous integration tools to automate security testing, enabling regular and consistent evaluation of security controls.

### E. *Monitoring and Scanning*

Bolster the release and deploy stages with additional monitoring and scanning tools to ensure the integrity of the software product across different environments.

## IV. Result



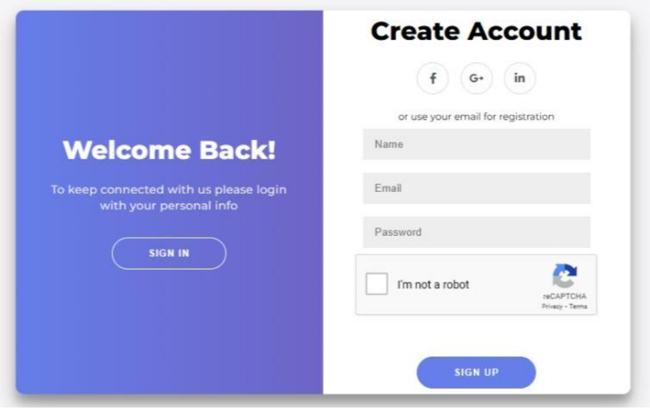
Presto x Booking

Welcome to our Hotel Booki

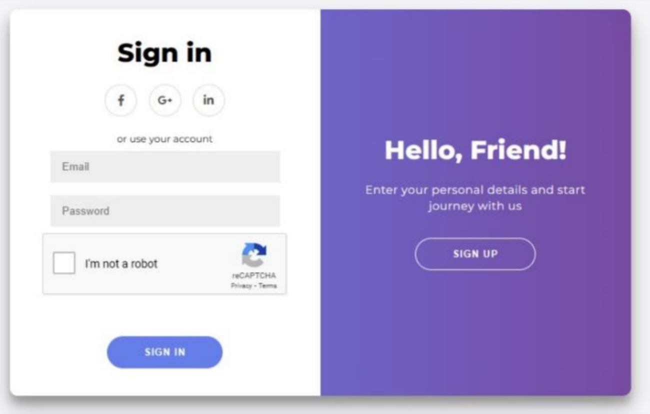
Where you can simply book with us.

For any assistance regarding your booking, please contact +60123232123 or email: help@booking.com

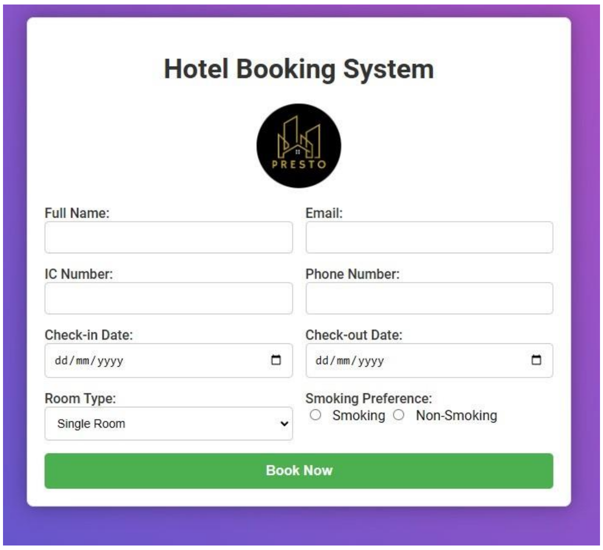
Index page for Presto Hotel Booking System



Create an account interface of Presto Hotel Booking System with reCAPTCHA



User authentication through sign-in interface with reCAPTCHA



Full Name:

Email:

IC Number:

Phone Number:

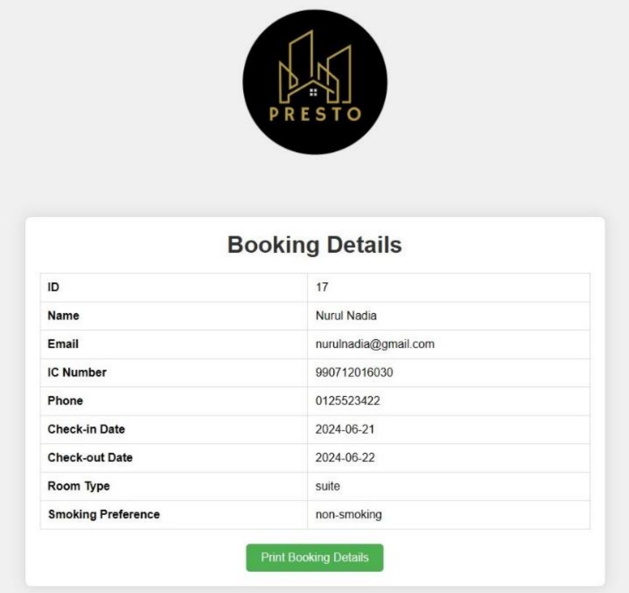
Check-in Date: dd/mm/yyyy

Check-out Date: dd/mm/yyyy

Room Type: Single Room

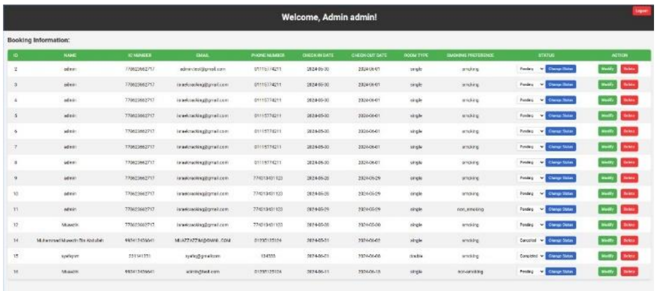
Smoking Preference: ○ Smoking ○ Non-Smoking

Entering details for Presto Hotel Booking System



| | |
|---|---|
| ID | 17 |
| Name | Nurul Nadia |
| Email | nurulnadia@gmail.com |
| IC Number | 990712016030 |
| Phone | 0125523422 |
| Check-in Date | 2024-06-21 |
| Check-out Date | 2024-06-22 |
| Room Type | suite |
| Smoking Preference | non-smoking |

Displaying detailed information after booking

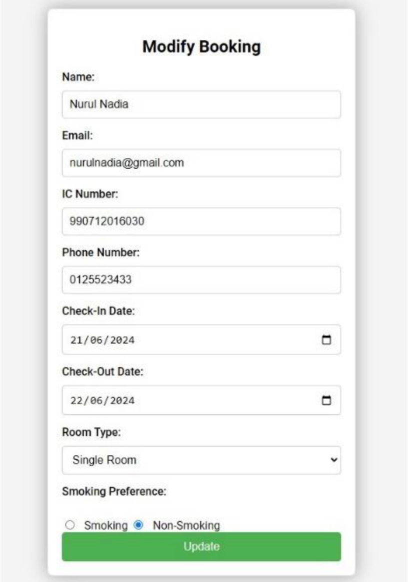


Welcome, Admin admin!

Booking Information:

| ID | NAME | IC NUMBER | EMAIL | PHONE NUMBER | CHECK IN DATE | CHECK OUT DATE | ROOM TYPE | SMOKING PREFERENCE | STATUS | ACTION |
|---|---|---|---|---|---|---|---|---|---|---|
| 2 | admin | 770623662717 | admin.test@gmail.com | 01115774211 | 2024-05-30 | 2024-06-01 | single | smoking | Pending Change Status | Modify Delete |
| 3 | admin | 770623662717 | israelcracking@gmail.com | 01115774211 | 2024-05-30 | 2024-06-01 | single | smoking | Pending Change Status | Modify Delete |
| 4 | admin | 770623662717 | israelcracking@gmail.com | 01115774211 | 2024-05-30 | 2024-06-01 | single | smoking | Pending Change Status | Modify Delete |
| 5 | admin | 770623662717 | israelcracking@gmail.com | 01115774211 | 2024-05-30 | 2024-06-01 | single | smoking | Pending Change Status | Modify Delete |
| 6 | admin | 770623662717 | israelcracking@gmail.com | 01115774211 | 2024-05-30 | 2024-06-01 | single | smoking | Pending Change Status | Modify Delete |
| 7 | admin | 770623662717 | israelcracking@gmail.com | 01115774211 | 2024-05-30 | 2024-06-01 | single | smoking | Pending Change Status | Modify Delete |
| 8 | admin | 770623662717 | israelcracking@gmail.com | 01115774211 | 2024-05-30 | 2024-06-01 | single | smoking | Pending Change Status | Modify Delete |
| 9 | admin | 770623662717 | israelcracking@gmail.com | 774313431123 | 2024-05-28 | 2024-05-29 | single | smoking | Pending Change Status | Modify Delete |
| 10 | admin | 770623662717 | israelcracking@gmail.com | 774313431123 | 2024-05-28 | 2024-05-29 | single | smoking | Pending Change Status | Modify Delete |
| 11 | admin | 770623662717 | israelcracking@gmail.com | 774313431123 | 2024-05-29 | 2024-05-29 | single | non_smoking | Pending Change Status | Modify Delete |
| 12 | Muaezin | 770623662715 | israelcracking@gmail.com | 774313431123 | 2024-05-28 | 2024-05-30 | single | smoking | Pending Change Status | Modify Delete |
| 14 | Muhammad Muaezin Bin Abdullah | 990412456641 | MUAZZAZZIM@GMAIL.COM | 01235125124 | 2024-05-31 | 2024-06-02 | single | smoking | Cancelled Change Status | Modify Delete |
| 15 | syafiqym | 231141231 | syafiq@gmail.com | 134333 | 2024-06-01 | 2024-06-06 | double | smoking | Completed Change Status | Modify Delete |
| 16 | Muaezin | 990412456641 | admin@test.com | 01235125124 | 2024-06-11 | 2024-06-13 | single | non-smoking | Pending Change Status | Modify Delete |

Admin dashboard (Booking information)



Modify Booking

Name: Nurul Nadia

Email: nurulnadia@gmail.com

IC Number: 990712016030

Phone Number: 0125523433

Check-In Date: 21/06/2024

Check-Out Date: 22/06/2024

Room Type: Single Room

Smoking Preference: ○ Smoking ● Non-Smoking

Modify booking details

## V. Discussion

The implementation of the SHA256 hashing function in the Presto Hotel Booking System is a critical measure to ensure data security, particularly in safeguarding guest information. This section analyzes the results of this cryptographic implementation through appropriate theoretical foundations, comparisons with similar systems, and a high level of analytical and critical discussion.

### A. *Theoretical Foundations of SHA256 Hashing*

Hashing functions, such as SHA256 (Secure Hash Algorithm 256-bit), are fundamental to modern cryptographic security. The primary purpose of a hashing function is to convert input data into a fixed-size string of characters, which is typically a hash code. The key properties of a secure hash function include:

i. **Deterministic**: The same input will always produce the same output.
ii. **Fast Computation**: The hash function should be capable of returning the hash value quickly.
iii. **Pre-image Resistance**: It should be computationally infeasible to reverse the hash function to obtain the original input.
iv. **Small Changes in Input Produce Unpredictable Changes in Output**: Known as the avalanche effect.
v. **Collision Resistance**: It should be difficult to find two different inputs that hash to the same output.

The SHA256 algorithm adheres to these principles, making it a robust choice for securing sensitive information within the Presto Hotel Booking System.

### B. *Comparison with Similar Systems*

To understand the efficacy of our implementation, we compare the Presto system with similar hotel booking systems that employ hashing functions for data security.

I. **Booking.com**: Utilizes advanced encryption standards and hashing techniques similar to SHA256 for password storage. This ensures that even if the database is compromised, the actual passwords remain secure.

II. **Expedia**: Implements SHA256 along with additional security protocols such as multi-factor authentication and encrypted data storage to enhance security.
III. **Airbnb**: Uses a combination of SHA256 hashing and bcrypt, a more computationally intensive hashing function, to ensure high levels of security for user data.

The comparison indicates that SHA256 is a widely accepted and robust solution in the industry. However, systems like Airbnb take additional steps by using bcrypt, which, while slower than nowadays system, provides an extra layer of security through its adaptive nature.

### C. *Analytical and Critical Examination*

The adoption of SHA256 within the Presto Hotel Booking System offers several advantages:

i. **Data Integrity and Security**: The hashed passwords ensure that even if the database is breached, the attackers cannot easily revert the hashes to plaintext passwords.
ii. **Performance**: SHA256 is designed to be fast, allowing for quick verification without noticeable delay to the user experience.
iii. **Compliance**: Using SHA256 helps in complying with industry standards and regulations such as GDPR, which mandate robust data protection mechanisms.

Despite SHA256's collision resistance, theoretical vulnerabilities could emerge with increased computational power. It is important to stay updated with cryptographic advancements. SHA256 is designed for speed, but its computational requirements may still pose challenges under high load conditions. Efficient resource management is necessary to maintain optimal performance.

### D. *Recommendations*

To further enhance the security and efficiency of the Presto Hotel Booking System, we recommend the following:

i. **Periodic Security Audits**: Regularly auditing the cryptographic protocols ensures that the system remains secure against emerging threats.
ii. **Consideration of Alternative Hashing Algorithms**: Exploring algorithms like bcrypt or Argon2 for password hashing could provide

additional security benefits due to their adaptive nature and resistance to brute-force attacks.

The implementation of SHA256 hashing in the Presto Hotel Booking System represents a significant step towards ensuring data security and integrity. Through a thorough theoretical understanding, comparison with industry standards, and critical analysis, we affirm that this approach meets the sophisticated demands of modern travelers and administrators. Continuous improvement and adherence to best practices will further solidify the system's security, positioning Presto at the forefront of secure and efficient hotel booking solutions.

## VI. Conclusion

In summary, this research has achieved all its goals and shown how password hashing can be used in real-world cryptographic applications. We have strengthened the integrity of the Hotel Booking System by incorporating strong security mechanisms that comply with the Open Web Application Security Project (OWASP) guidelines. This has ensured increased security and system efficiency.

The results of this study make a substantial contribution to the subject of cybersecurity by providing insightful analysis and useful applications that can be applied by others in related fields. The implementation of OWASP standards and protocols emphasizes how crucial it is to uphold strict security procedures in digital systems, which improves the general dependability and credibility of web-based apps.

Looking ahead, suggestions for more research and development might be made to improve the system. First off, adding multi-factor authentication (MFA) to the security measures would add another degree of defense. Furthermore, the quick identification and mitigation of possible threats could be facilitated by the implementation of continuous security monitoring and regular vulnerability assessments. Finally, investigating innovative cryptographic methods like homomorphic encryption could strengthen systemic data security and privacy even more. The Hotel Booking System can keep developing and hold its leading position in safe and effective digital solutions by exploring these possibilities.

## References

[1] S. Colcol, “Everything You Need to Know about Hotel Reservation Systems,” *SiteMinder*, Mar. 17, 2023. https://www.siteminder.com/r/hotel-reservation-system/

[2] A. Kumar, “Hotel Booking Management System Project | Hotel Booking Management Project,” *PHPGurukul*, Apr. 27, 2020. https://phpgurukul.com/hotel-booking-management-system-using-php-and-mysql/

[3] S. Bondar, “PHP and Password Hashing: Securely Storing and Verifying Passwords | Reintech media,” *reintech.io*, Apr. 28, 2023. https://reintech.io/blog/php-password-hashing-securely-storing-verifying-passwords

[4] Owasp, “Input Validation · OWASP Cheat Sheet Series,” *Owasp.org*, 2019. https://cheatsheetseries.owasp.org/cheatsheets/Input_Validation_Cheat_Sheet.html

[5] “OWASP Secure Coding Practices - Quick Reference Guide | Secure Coding Practices | OWASP Foundation,” *owasp.org*. https://owasp.org/www-project-secure-coding-practices-quick-reference-guide/stable-en/01-introduction/05-introduction.html

[6] “XAMPP tutorial: installation and first steps,” *IONOS Digital guide*. https://www.ionos.com/digitalguide/server/tools/xampp-tutorial-create-your-own-local-test-server/