

Development of mobile application for keyless smart lodging

S.F. Pratama^{1,*}, S.K. Subramaniam², M. Esro², A.K. Muda¹, R. Abd. Samad³

¹Center for Advanced Computing Technology, Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia

²Centre for Telecommunication Research & Innovation, Fakulti Kejuruteraan Elektronik dan Kejuruteraan Komputer, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia

³Centre of Technopreneurship Development, Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia

*Corresponding author's email: satrya@utem.edu.my

ABSTRACT: Tourism is one of the economic sectors severely affected by the enforcement of various Movement Control Orders by the Government of Malaysia, especially in the states that rely on it, such as Melaka. Nevertheless, Melaka State Government remains committed to the Smart Melaka Initiative, especially in the tourism sector, such as accommodation and lodging industry. To support the Smart Melaka Initiative and at the same time adhering to the health guideline to minimize unnecessary contacts, this research proposes to develop an integrated system for the accommodation and lodging premises, which is aimed to replace existing key and/or access card to open the guest room. This paper focuses on the development of the mobile application as one of the components of the proposed integrated system, which is used to control the door lock of the guest room by employing various sensors commonly available in the recent mobile devices.

Keywords: *mobile application; keyless entry; smart lodging*

1. INTRODUCTION

Prior to the enforcement of various Movement Control Orders (MCOs) by the Government of Malaysia, Melaka State Government has enacted the Melaka Smart City Policy 2030 (Dasar Smart City Melaka 2030) through what is known as Smart Melaka Initiative, which consists of five main strategies: smart mobility, smart environment and smart energy, smart living, smart governance, and smart economy [1]. The tourism industries have been recognized as one of the main contributors to the state economy, and thus it has become one of the main target audiences of Smart Melaka Initiative.

The use of ICT in tourism sectors breeds the new domain called smart tourism [2]. One of the subdomains of smart tourism is called Tourism 4.0, which exploits the Industrial Revolution 4.0 involving interconnectivity, automation, machine learning, and real-time data. Internet of Things (IoT) has been extensively used in Tourism 4.0 [3], and thus it is recommended to implement the IoT-powered systems and applications to support the Smart Melaka Initiative.

Based on the increasing demand for the accommodation and lodging by tourists, while at the same supporting the Smart Melaka Initiative, this research proposes to develop an integrated system for smart accommodation and lodging, specifically for the guest room door access. Instead of using key or access card, which is prone to loss, this research proposes to employ the mobile application to access their room.

Furthermore, the use of mobile application for accessing their room also reduces the number of exchanging objects between the receptionist and guests, and thus minimize the risks of spreading infectious diseases such as Covid-19. The remainder of this paper is structured as follows. The methodology of the proposed application and its results and discussion are presented in Sections 2 and 3, respectively, while Section 4 discusses the conclusion and future works.

2. METHODOLOGY

The proposed system consists of mobile application, which will be used by the guests, IoT-capable guest room microcontrollers, and a cloud-based web platform. This paper, however, will only focus on the development of the mobile application. The proposed mobile application is developed using Object-Oriented methodology and implemented using Java programming language on Android Studio for the Android and Swift programming language on Xcode for the iOS. However, due to limited space, the UI designs of the mobile application cannot be presented in this paper.

The mobile application utilizes the camera and Bluetooth sensors commonly available on recent mobile devices. These sensors are accessed using AndroidX Jetpack's `CameraX` and built-in Bluetooth Low Energy (BLE) libraries for the Android, and built-in `AVFoundation` and `CoreLocation` frameworks for the iOS. Both mobile application versions also utilize the Google ML Kit library for QR code recognition.

3. RESULTS AND DISCUSSION

This section discusses the results of the mobile application development. The mobile application consists of several main modules, namely login, guest check-in, and room entry modules. The architectural design of the system is illustrated in Figure 1.

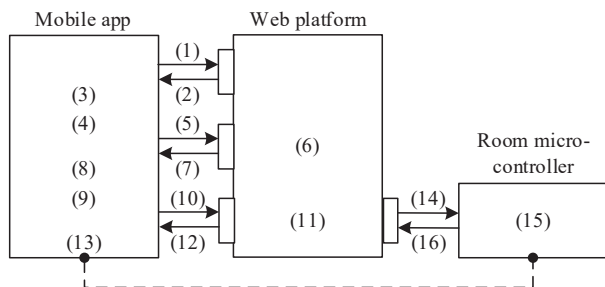


Figure 1 Architectural design of the proposed system

The flow of the system as shown in Fig. 1 is as follows. The login module is used by the guest by submitting their username and password to the web platform's login Representational State Transfer Application Programming Interface (REST API) endpoint via Secure Hypertext Transfer Protocol (HTTPS) protocol (1). Upon successful authentication, the web platform will return a unique guest identifier (ID) (2), which will be stored in the `SharedPreferences` and `UserDefaults` storage engine in the Android and iOS mobile applications, respectively (3), and the guest will be directed to the main menu, where the guest can access the guest check-in and room entry modules. However, room entry module is hidden by default until the guest has successfully checked-in in the lodging premise. Furthermore, subsequent usages of the mobile application will not prompt the guest to login again, unless they decide to logout from the mobile application.

The check-in module is used by the guest upon arrival in the premise, where the guest will use the mobile application to scan and retrieve a unique QR code representing the guest room ID displayed by the receptionist (4), which will be sent along with their unique guest ID to the web platform's guest check-in REST API endpoint via HTTPS protocol (5). The web platform will bind the guest ID to the guest room ID until the designated check-out date and time (6). Upon successful binding, the web platform will return a success flag to the mobile application (7), which will store the guest room ID in the storage engine and temporarily show the room entry module until check-out time (8).

The guest will then use the room entry module to access their assigned room. Upon activation of this module, the mobile application will send the guest's unique ID and a randomly generated universally unique ID string, which is assigned as key string (9), to the web platform's room entry REST API endpoint via HTTPS protocol (10). The web platform will look up the guest's assigned room, and upon successful validation, will simultaneously send a Remote Procedure Call (RPC) command to the room's microcontroller via Message Queuing Telemetry Transport (MQTT) protocol with the key string as its parameter and return a success flag to the mobile application (11).

When the mobile application receives the success flag from the web platform (12), the mobile application will activate the Bluetooth sensor of the mobile device, set them as a BLE beacon, and broadcast the key string using the BLE beacon (13). After 30 seconds, the mobile application will stop the BLE beacon broadcast and

deactivate the Bluetooth sensor to minimize the mobile device's battery consumption.

Simultaneously, when the guest room microcontroller receives the RPC command with the key string embedded (14), the microcontroller will use its Bluetooth sensor to scan the BLE beacon that is broadcasting the key string, essentially validating the proximity of the guest to the room (15). Once the guest's mobile device is detected, it will unlock the assigned guest room's door. After the mobile device is detected or after 30 seconds has elapsed without detection, the guest room microcontroller will send the response for the RPC command to the web platform via MQTT protocol (16). Despite the slightly longer response time compared to the existing commercial products [4, 5], the proposed system is more secure because the responsibility to verify the guest unlock request is handled by the web platform instead of directly by the microcontrollers.

4. CONCLUSION

The proposed mobile application for keyless smart lodging has been proposed and successfully implemented for Android and iOS platforms in this paper. The proposed mobile application is one of the components of an IoT-based integrated system. It consists of three main modules, which is the login, guest check-in, and room entry modules, and utilizes the camera and Bluetooth sensors of the mobile device. Additional modules to the proposed mobile application, as well as the discussion of the IoT-capable guest room microcontroller and cloud-based web platform designs and implementations will be thoroughly elaborated in the future works.

ACKNOWLEDGEMENT

Authors are grateful to Universiti Teknikal Malaysia Melaka for the financial support through Short Term Research Grant (PJP/2020/FTMK/PP/S01796).

REFERENCES

- [1] Smart City Advisory Council. "Smart City Melaka." Kerajaan Negeri Melaka. <https://smartcitymelaka.com/> (accessed 20 September 2020).
- [2] U. Gretzel, M. Sigala, Z. Xiang, and C. Koo, "Smart tourism: foundations and developments," *Electronic Markets*, vol. 25, no. 3, pp. 179-188, 2015.
- [3] A. Verma and V. Shukla, "Analyzing the Influence of IoT in Tourism Industry," in *International Conference on Sustainable Computing in Science, Technology and Management*, Jaipur - India, 2019: Elsevier.
- [4] OmniTec Systems. "Electronic Hotel Lock with Bluetooth." <https://www.omnitecsystems.com/products/electronic-hotel-lock/electronic-lock-hotel-bluetooth-evo> (accessed 20 September 2020).
- [5] OpenKey. "OpenKey Locks." <https://www.openkey.co/openkey-locks/> (accessed 20 September 2020).