

Smart Parking IoT Based: Design and Prototype

Direstu Amalia¹, Rangga Hadiansyah¹, Virma Septiani¹

¹Politeknik Penerbangan Palembang

Corresponding Author E-mail: direstu@poltekbangplg.ac.id

Received 24 June 2021; Revised 28 November 2021; Accepted 5 January 2022

Abstract:

The Arduino Mega 2560 microcontroller is utilized as the processor in this study to construct an IoT-based smart parking prototype. By using research and development methods, system and infrastructure prototypes using sensors, processors, and web-based reservation apps are now the techniques used. Users may select parking slots and view slots available using HMI and an online reservation application designed using App Inventor. This display communicates with the system in real time over the internet. The hardware is environmentally beneficial because it is energy efficient and long-lasting. The smart parking system design has been extensively tested in the past. However, in this study, the payment system is examined, where payments will only be calculated when the user enters and exits the designated parking slot. Through this design, it is hoped that there will be product contributions that can be implemented into society and become a new practical model for learning microcontrollers.

Keyword: Internet of Thing, Microcontroller, Arduino Mega, Smart Parking, MIT Inventor

1. Introduction

The internet has infiltrated many facets of our lives, including personal needs, education, and even the economy. The availability of an internet connection has immensely aided us in a variety of ways, from simply watching television to visiting websites that give various services and entertainment, even download web-based

programs onto our devices. The Internet of Things (IoT) is a term used to describe a physical object (or a group of such objects) that is equipped with sensors, processing capabilities, software, and other technologies and connects and exchanges data with other devices and systems over the Internet or other communication networks. Developing a generic architecture for the Internet of Things is thus a difficult endeavour, owing to the enormous number of devices, link layer technologies, and services that could be involved in such a system (Zanella, Bui, Castellani, Vangelista, & Zorzi, 2014). The Internet of Things (IoT) has received a lot of research attention over the year. The Internet of Things (IoT) is a component of the Internet of the Future, and it will consist of billions of intelligently communicating 'things.'

The Internet of the future will be made up of heterogeneously connected gadgets that will expand the world's borders with physical and virtual things (Li, Xu, & Zhao, 2014). Data collection and real-time analysis, workload management, manufacturing process optimization, prediction in determining maintenance plans, and automation are all possible with the Internet of Things. Robotics and automation, predictive maintenance, and problem diagnosis are all aided by IoT-enabled manufacturing systems. It's a work utility with an intelligent system that creates a stable and efficient network, as well as control and monitoring capabilities and digital communication. The Internet of Things has been used in production systems for oil and gas companies, logistic transportation systems, and smart buildings. To put it another way, this technology creates a new experience, improves security and monitoring, enhances resource efficiency, and reduces cost immediately.

2. Literature Review

Several previous studies have used a range of options to construct the system (Ali et al., n.d.)(... & 2016, n.d.)(Chatzigiannakis, Vitaletti, Communications, & 2016, n.d.)(Ganesh Gopal, Asha Jerlin, & Abirami, 2019)(in & 2017, 2017) (Wang & He, 2011). According to the LIPI Institute, the current parking problem has an impact on regional income and air pollution, as well as the fact that parking spots in major cities are scarce and automobile mobility is high. As a result, the best solution is to implement a smart parking system that allows drivers to order parking online and properly search for vehicle parking site (Haryono, 2019). Researchers

conducted numerous literature research to make sure that the system design is improved from the previous version, which served as a prototype for this design.

The current smart parking design is generally to help find a parking slot immediately or choose the desired parking slot. Generally, the vehicle is considered parked and charged a parking fee when entering the parking gate, and the parking fee continues to be calculated until the vehicle exits the parking exit gate, even though in reality, maybe the car left the parking slot minutes ago, but because of the queue, they just got through the exit gate and have charge more. A classic complaint is that users have to pay more for multiples of minutes to queue out of the parking lot. The existing smart parking system still has the opportunity to be optimized, especially in large and dense parking lots. It is necessary to rethink a system that is not only fast in identifying vacant parking slots but also provides fairness in payments, considering the large and dense parking lot, just getting out of the parking lot can take several minutes. So, in this study, the system that is improved is a smart parking system with an online reservation facility that processes payments when the car enters the parking slot, and once the car leaves the parking slot, the system automatically calculates the total payment even before the vehicle passes through the parking exit gate.

App Inventor and Firebase

App Inventor is a web-based tool that lets users design Android and iOS apps right from their browser. This website enables users to connect to all the tools they need to learn how to build their own apps. This is accessible at MIT app inventor website. To do it, click the orange "Create Apps!" option from any page on this website. Instructions for setting up Set up a live testing environment on your smartphone or tablet (or, if you do not have a mobile device, run the emulator). 2) Get a feel for the Designer and Blocks Editor by going about the App Inventor environment. Last, this apps let you create a package for your app.

The aim of Firebase's Realtime Database technology is to enable data to be shared between different clients, where a "client" can be an app running on an Android or iOS device or JavaScript running on a web server. The system's major function is to provide a secure, trustworthy, and efficient way to synchronize data

with minimal coding work on the developer's behalf. The database system has been built to support millions of users, and the database is termed "realtime" because the pace at which data is synced between clients is probably as close to realtime as is currently possible (given the physical limitations of transmitting data over the internet and wireless connections). The time taken for a data update on one client to propagate to another is visually undetectable, as will be shown in later chapters. The Realtime Database also supports data persistence by storing data locally, allowing data to be accessed even when a device is not connected to the internet. The local data is automatically synchronized and combined with the remote data when connectivity is restored.

Mikrokontroler Arduino Mega 2560

The Arduino Mega 2560 is an open-source hardware platform that provides the ATmega 2560 microcontroller and is an 8-bit board. This board features a lot of I/O pins, including 54 digital inputs and outputs, 15 of which are PWM outputs, 16 analog inputs, and 4 UARTs (hardware serial ports). A 16 MHz crystal oscillator is also included in the Arduino Mega 2560. Simply connect electricity to the PC/laptop by USB or via DC Jack utilizing a 7–12 V DC converter for a simple setup.

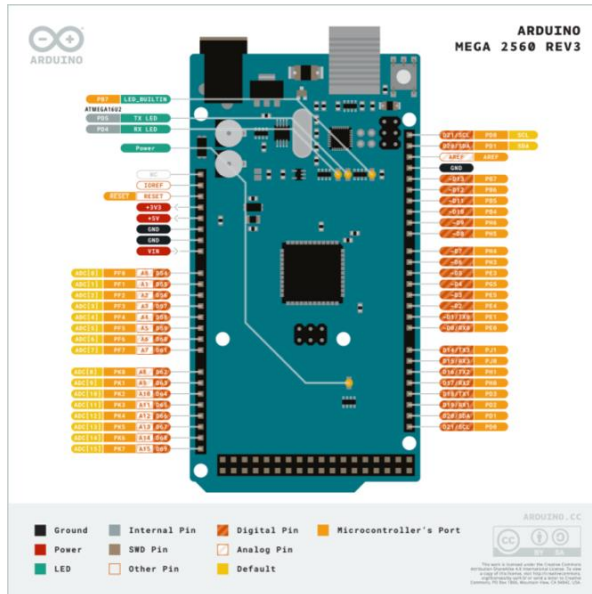


Figure 1. Arduino Mega 2560 Rev 3 Specification

2. Methods

The methodology is to build and develop a model that explains the process or strategy that's used to design, develop, or plan the smart parking project. This article describes the software and hardware that will be used further to implement this project.

3. Results and Discussion

Block Diagram

This section shows the general system design with a diagram, which shows the input, process, and output of the system.

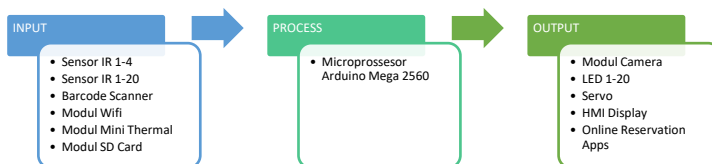


Figure 2. Control System Diagram

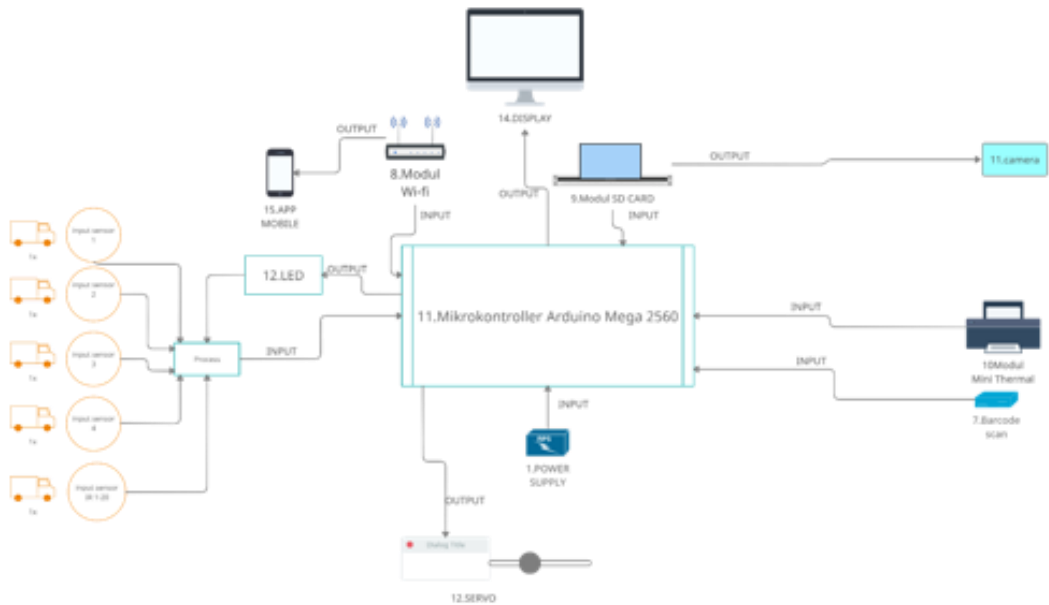


Figure 3. Diagram Block of Smart Parking System

Table 1. Details of the hardware used to build the system

No	Hardware Types	Information
1	Power supply	The power supply functions as input power to run the "tool name" system.
2	Sensor IR 1 (INFRA RED 1)	Functions as an input sensor when the driver is about to take a ticket, IR sensor 1 replaces the pushbutton function to take parking tickets.
3	SENSOR IR 2 (INFRA RED 2)	Functions as an input sensor to close the parking gate when the vehicle has passed the gate.
4	Sensor IR 3 (INFRA RED 3)	Functions as an input sensor when the driver is about to take a ticket, the IR 3 sensor replaces the pushbutton function to take a parking ticket.
5	SENSOR IR 4 (INFRA RED 4)	Functions as an input sensor to close the parking gate when it has passed the gate.
6	SENSOR IR 1-20 (INFRA RED 1-20)	Functions as an input sensor to indicate the presence or absence of vehicles in the parking area.
7	BARCODE SCANNER	Functions to scan barcode ticket for payments at the exit gate.
8	Modul WIFI	Wemos D1 R1 WIFI module serves to connect hardware to the internet.
9	Modul SD CARD	The SD CARD module functions to store ticket data and parking customer images.
10	Modul Mini Thermal	Modul Mini Thermal for printing ticket

11	Modul camera	To take pictures of customers when infrared 1 (IR 1) detects a parking customer.
12	LED	Functions as an indicator light on the parking lot
13	Servo	Servo serves to open and close the gate.
14	HMI Display	The HMI display is useful for selecting a parking slot through the parking gate.
15	Reservation Apps	Web based apps that connected to microcontroller

Circuit Design

This section shows in detail the circuit design made according to the block diagram above.

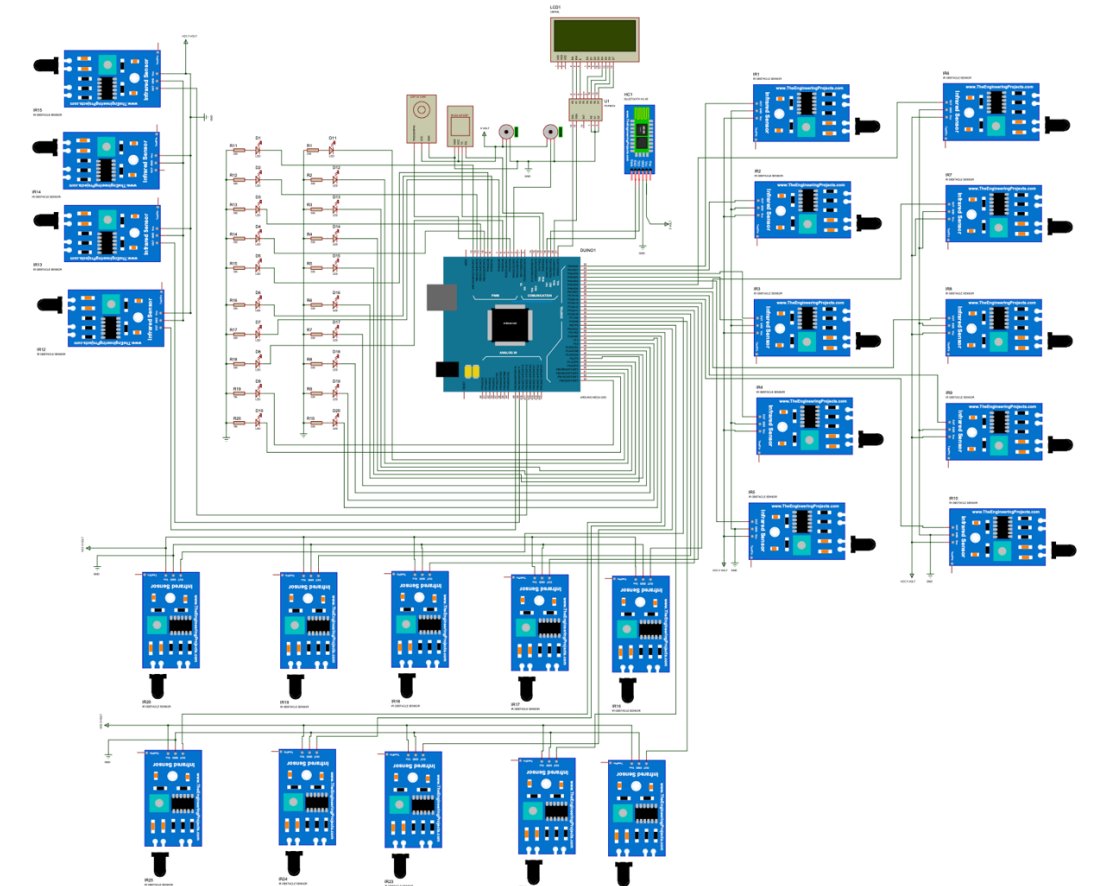


Figure 4. Circuit Design of Smart Parking System

Prototype Design

The hardware design for the proposed smart parking prototype is shown in this section. A sensor will be installed in each of the 20 slots in the parking lot design. All the components will be organized and hidden in a box at the table's bottom. Only the servo, camera, and thermal print will be displayed. The remaining components, including the Multiprocessor device, will be stored below the table. This is set up for the display to appear precise and organized, as though it's a real parking lot. This smart parking prototype design comprises of an iron-legged table with an acrylic box of the dimensions described in the figure. The anti-corrosion paint on the iron legs ensures that they will be safe and can be used for a long period. (Utara & 2004, n.d.)(Teknologi & 2015, 2015).

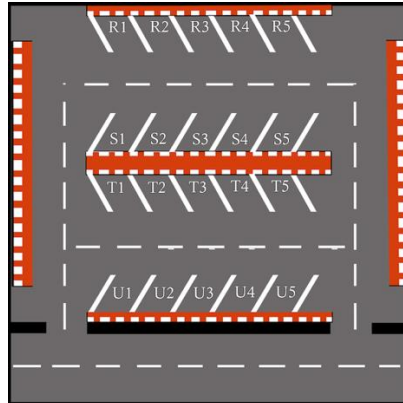


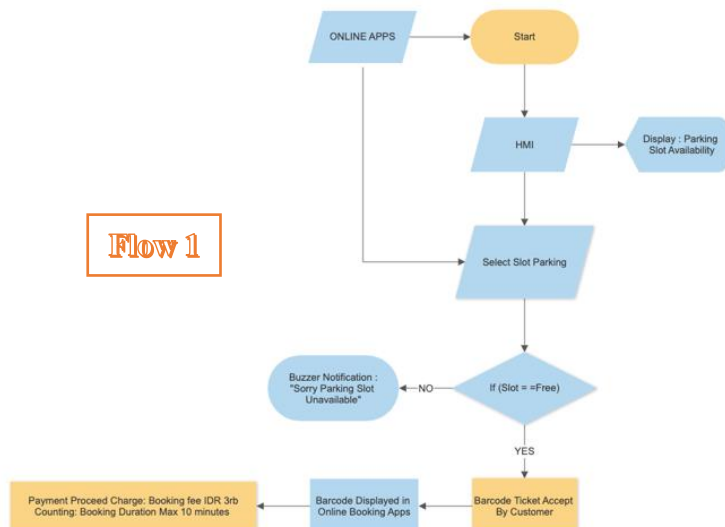
Figure 5. The Parking Lot Design



Figure 6. The Tabel Design

Flowchart System

This section shows in detail the mechanism in the smart parking system that is made. One of the most significant aspects of system design is determining the system's path and describing it in a flowchart. This is to address the required design easier to implement. Each flow chart is assigned to a specific process or system (Xinogalos, 2013)(Ensmenger, 2016). Flow charts are a useful tool for streamlining procedures. They assist project teams in identifying the various aspects of a process and understanding the interrelationships among the various steps by giving a graphical picture.



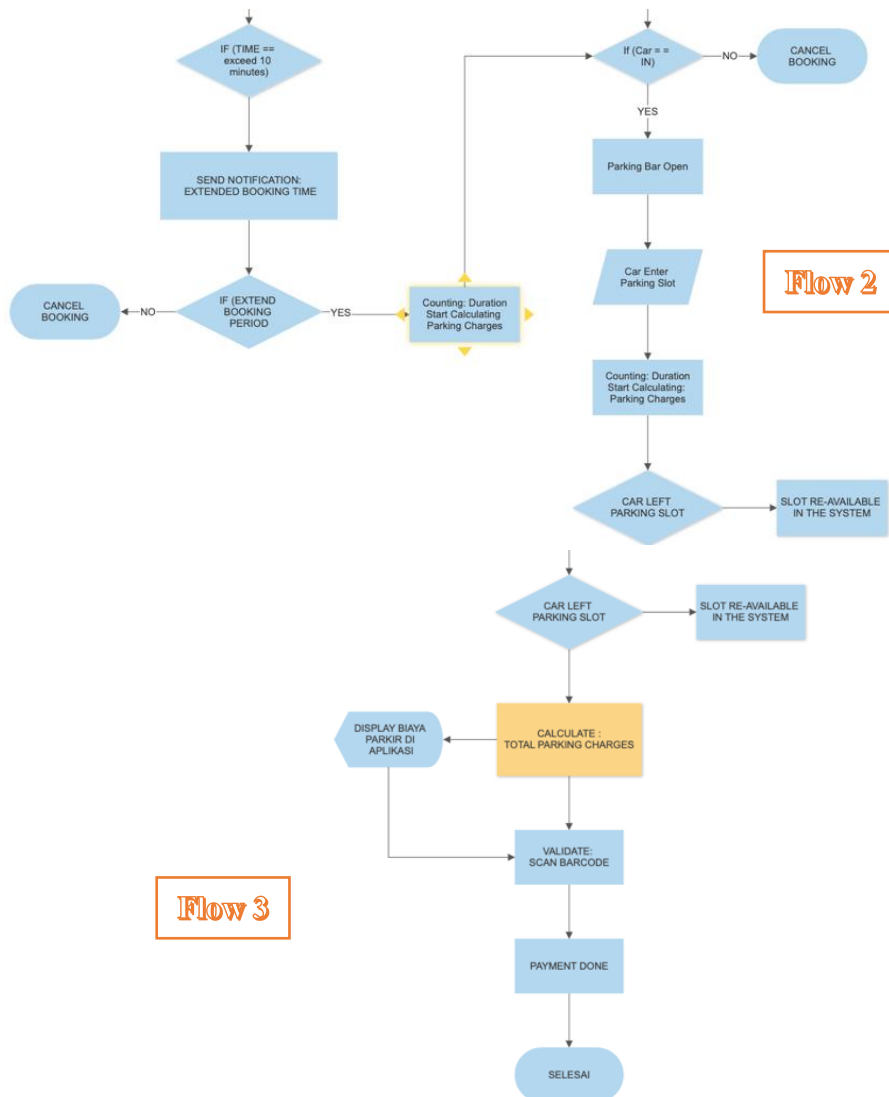


Figure 7. Flowchart of the system

Sensors have been used as inputs in most parking systems to detect the presence of users at the entrance and issue tickets. Image processing captures the driver's license plate. The information is stored in a database after a camera captures

the driver's license plate. This is to minimize car theft and unauthorized users. In the use of Online Reservation Apps, before utilizing the app, user must first register, and the user's information is saved in a database that serves as part of the security system. Users who use the online application can book a slot for a price of 3,000 rupiah, which is immediately withdrawn by a reservation application that is linked to the online payment system, as well as the e-ticket appears in the application. The system has set a booking time limit of 10 minutes, and if the vehicle does not arrive within that time, the system will calculate the regular parking time and charge with the user's permission.

Every user can cancel the booking immediately if the request to compute the charge is not approved, though the booking fee of 3 thousand rupiah is non-refundable. When the user completes the online reservation and arrives at the destination, everything he must do is scan the e-ticket to confirm that he is the one who booked the reservation. Unlike users who drive at the parking location without first making an online reservation, there is an HMI display in front of the parking entrance gate that presents the status of available slots and enables the driver to choose the slot to be addressed. The barcode ticket will then be printed and scanned at the entrance by the driver. The updated payment criteria in this research are that the driver will be charged when the car is detected entering the parking slot, and the total parking charge will be calculated and processed immediately after the vehicle leaves the parking slot. Each parking spot has an IR sensor that detects the presence or absence of parked vehicles, so that when the vehicle leaves the slot, the system receives the signal that the slot is available for other users to order.

Smart Parking Mechanism

1. Power Supply: Power Supply is used to provide the required electrical voltage source to the tool.
2. Input System: The input system in this research is the sensor used, namely infrared 1, infrared 2, infrared sensors 3. The function of infrared 1 is as a trigger to replace the button to issue a ticket, infrared 2 to close the gate that is open when the parking user passes. Infrared 3 as a vehicle detection sensor in the parking lot that is already filled.

3. System process: The system process is where all inputs and outputs are combined and processed. In this study, the researchers used the Arduino Mega as artificial intelligence to manage the tools in this study, where the Arduino Mega that the researchers used included a wifi system as a liaison to the Internet of Things (IoT) system. This process is also equipped with a mini thermal module to print tickets.
4. Output system: the system output here is the result that is issued in the process, namely in the form of a servo motor for parking gates, parking slot indicator led lights, led matrix to see the number of parking slots.
5. Mobile applications: With the app inventor, researchers can do computer programming to create software applications with an Android-based operating system. This app inventor is based on visual block programming because it allows users to use, view, compose and drag and drop blocks which are command symbols and event handler functions to create an application that can run on the android system.
6. Database: Firebase is a database management system based on NoSQL. Firebase is a service from Google to provide convenience and even make it easier for application developers to develop their applications. Firebase aka BaaS (Backend as a Service) is a solution offered by Google to speed up developer work. By using Firebase, app developers can focus on developing applications without putting a lot of effort into the backend.

Online Reservation System

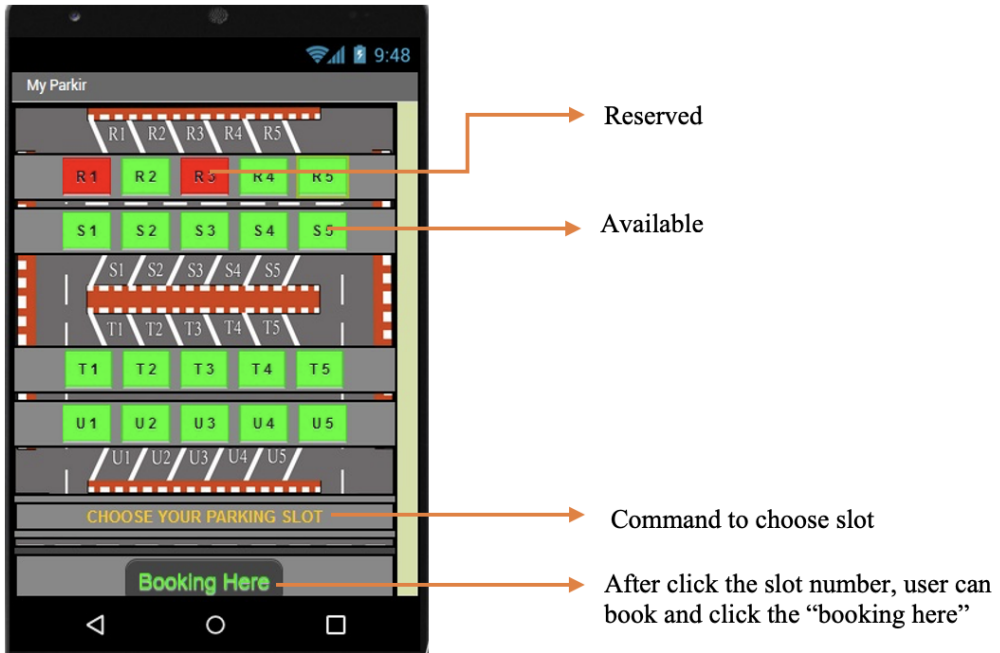


Figure 8. Online Reservation Display

When user initiates activation of a smart parking system via an application, the process flows as follows. Important information such as a user's name and vehicle plate number must be entered. The user is then moved to the next window, which displays the current slot availability. A filled slot is indicated by red, whereas an empty spot is indicated by green, and the user can opt to reserve it. The e-ticket will then show up on the application page. This e-ticket distinguishes between online reservation users and non-reservation users. Offline users click via the HMI Display, which has the same display with an online apps, at the parking entrance gate having go through the regular ticketing process. The system begins to calculate the duration and parking price as soon as the car enters the specified parking slot. Online reservations are different in that there is a booking cost and usual service charges will be charged if the car does not arrive within the given period. Furthermore, when the parking reservation time is due for 10 minutes, the user receives a notification as a reminder, with the option to extend the duration of their

reservation. Since this is not the case, the user has the option to cancel. The system will display the whole charge for online users, and they will be able to make payments. Inside the system, slots will be reallocated.

2. Conclusion

This system follows the flowchart specified, and all hardware can function as expected. Some research related to smart parking online reservations has similarities in hardware and system (in & 2017, 2017) (... & 2016, n.d.) (Chatzigiannakis et al., n.d.)(Ganesh Gopal et al., 2019). The flowcharts and payment systems used, on the other hand, are significantly different. Because this system is currently a prototype of smart parking, researchers hope it can be implemented in the real parking system to verify the level of satisfaction of parking management and parking service users.

3. Acknowledgment

We thank you for the ideas and input from colleagues, and all who have supported this research to completion. Thank you to my family for always supporting this research journey. I am blessed.

4. References

- Ali, G., Ali, T., Irfan, M., Draz, U., Sohail, M., Glowacz, A., ... Martis, C. (n.d.). IoT based smart parking system using deep long short memory network. *Mdpi.Com*. <https://doi.org/10.3390/electronics9101696>
- Chatzigiannakis, I., Vitaletti, A., Communications, A. P.-C., & 2016, undefined. (n.d.). A privacy-preserving smart parking system using an IoT elliptic curve based security platform. *Elsevier*. Retrieved from <https://www.sciencedirect.com/science/article/pii/S014036641630072X>
- Ensmenger, N. (2016). The Multiple Meanings of a Flowchart. *Http://Utpress.Utexas.Edu/Index.Php/Genders*, 51(3), 321–351. <https://doi.org/10.7560/IC51302>
- Ganesh Gopal, D., Asha Jerlin, M., & Abirami, M. (2019). A smart parking system using IoT. *World Review of Entrepreneurship, Management and Sustainable Development*, 15(3), 335–345.

<https://doi.org/10.1504/WREMSD.2019.099409>

- Haryono, A. (2019). Solusi Lahan Parkir yang Kian Terbatas [The Solution for Parking Spaces That Are Increasingly Limited]. Lembaga Ilmu Pengetahuan Indonesia. Retrieved November 20, 2021, from LIPI website: <http://lipi.go.id/lipimedia/solusi-lahan-parkir-yang-kian-terbatas/21535>
- in, P. S.-2017 I. conference on advances, & 2017, undefined. (2017). An IoT-based E-parking system for smart cities. *Ieeexplore.Ieee.Org*. <https://doi.org/10.1109/ICACCI.2017.8125982>
- Li, S., Xu, L. Da, & Zhao, S. (2014). The internet of things: a survey. *Information Systems Frontiers* 2014 17:2, 17(2), 243–259. <https://doi.org/10.1007/S10796-014-9492-7>
- P. R.-I. C. on R. T. in, & 2016, undefined. (n.d.). IoT smart parking system for reducing green house gas emission. *Ieeexplore.Ieee.Org*. Retrieved from <https://ieeexplore.ieee.org/abstract/document/7569513/>
- Teknologi, S. U.-J., & 2015, undefined. (2015). Pengaruh Konsentrasi Larutan NaNO₂ sebagai Inhibitor terhadap Laju Korosi Besi dalam Media Air Laut [Effect of Concentration of NaNO₂ Solution as Inhibitor on Iron Corrosion Rate in Seawater Media]. *Jurnal.Umj.Ac.Id*, 7(2). Retrieved from <https://jurnal.umj.ac.id/index.php/jurtek/article/view/400>
- Utara, I. D.-U. S., & 2004, undefined. (n.d.). Kimia dari inhibitor korosi [The chemistry of corrosion inhibitor]. *Academia.Edu*. Retrieved from <https://www.academia.edu/download/33877492/Korosi.pdf>
- Wang, H., & He, W. (2011). A reservation-based smart parking system. *2011 IEEE Conference on Computer Communications Workshops, INFOCOM WKSHPs 2011*, 690–695. <https://doi.org/10.1109/INFCOMW.2011.5928901>
- Xinogalos, S. (2013). Using flowchart-based programming environments for simplifying programming and software engineering processes. *IEEE Global Engineering Education Conference, EDUCON*, 1313–1322. <https://doi.org/10.1109/EDUCON.2013.6530276>
- Zanella, A., Bui, N., Castellani, A., Vangelista, L., & Zorzi, M. (2014). Internet of things for smart cities. *IEEE Internet of Things Journal*, 1(1), 22–32. <https://doi.org/10.1109/JIOT.2014.2306328>