Journal of Artificial Intelligence and Engineering Applications

Website: https://ioinformatic.org/

15th October 2025. Vol. 5. No. 1; e-ISSN: 2808-4519

E-Parking Monitoring Using Microcontroller-Based IOT Technology

Muhammad Fachri[1]*, Imran Lubis[2]

[1, 2] Informatic Engineering, Universitas Harapan Medan, Indonesia
mhd.fachrii30@gmail.com[1], imran.loebis.medan@gmail.com[2*]

Abstract

The rapid growth of urban areas has increased the demand for efficient parking systems, particularly in congested public spaces. This study presents the design and implementation of an IoT-based e-parking monitoring system utilizing NodeMCU ESP8266 and infrared (IR) sensors to detect parking slot availability in real time. A web-based interface enables users to monitor and reserve parking slots remotely with secure authentication. Testing results demonstrated that the system provides fast response times (2–21 ms) and accurate real-time updates on the web dashboard. Additional feedback mechanisms, such as buzzer alerts for unauthorized use, further enhance operational efficiency and parking discipline. This research shows that IoT integration significantly improves parking management and offers scalability and potential integration for future urban smart parking systems.

***Keywords**: E-Parking, Internet of Things, NodeMCU ESP8266, Parking Monitoring, Real-time Booking*

1. Introduction

The rapid advancement of technological innovation has ushered society into an era where technology dominates numerous aspects of human life and labor. One of the most transformative developments is the Internet of Things (IoT), a paradigm shift that enables physical devices to interconnect and be controlled remotely via the internet [1]. Although the historical progression of IoT spans several decades, recent technological breakthroughs have unlocked new possibilities for addressing multifaceted challenges across various sectors, including parking management systems [2], [3].

Accelerated urbanization and the surge in private vehicle ownership, particularly in cities such as Medan, have precipitated complex infrastructural issues, most notably traffic congestion and parking scarcity [4]. In high traffic public zones including commercial centers, office complexes, airports, hospitals, and other municipal facilities, the demand for parking spaces has reached critical levels [5]. Inadequate parking capacity or inefficient management can disrupt operations in these key areas [6].

Traditional parking systems, which typically lack real-time occupancy updates for drivers, further aggravate these inefficiencies [7]. Motorists are often compelled to circle parking areas in search of vacant slots, resulting in time wastage, excessive fuel consumption, and vehicular pileups [8]. While modern parking gates employ automated ticketing mechanisms, the absence of real-time slot availability data forces drivers to manually inspect parking spaces, rendering the current model highly inefficient [9].

A study by Pulungan et al, proposed an automated parking system that displays occupancy status via an LCD screen at the entrance. The system utilizes binary indicators "0" for vacant and "1" for occupied and notifies drivers with a "SORRY PARKING FULL" alert when capacity is reached, subsequently restricting gate access. This design aims to minimize entrance queues and mitigate congestion. However, a critical limitation persists: drivers only receive parking availability information upon arrival [10]. Pre-trip parking data accessibility remains unaddressed, despite its significance in optimizing travel efficiency and user convenience.

To bridge this gap, this study introduces an IoT-based microcontroller e-parking monitoring system capable of providing real-time parking availability updates accessible via mobile platforms. The proposed framework integrates sensor networks to detect occupancy status and relays data to a centralized server. End users can access this information through dedicated web or mobile applications, while parking administrators gain enhanced monitoring capabilities. Furthermore, the system can be expanded to incorporate pre-booking functionalities, allowing users to reserve parking slots in advance by specifying duration a feature explored in this research.

2. Research Methods

2.1. Type of Research

This study uses a Research and Development (R&D) approach aimed at developing a new system, namely a prototype parking monitoring system based on Internet of Things (IoT) technology. The R&D method was chosen because it allows for a systematic development process, which includes the stages of problem identification, system design, implementation and integration, testing, and system evaluation and revision. In addition, an experimental approach is used to evaluate the effectiveness of the developed prototype, both in terms of the functionality of the sensors used and the reliability of the IoT system when applied in real operational conditions.

2.2. Tools and Materials

A. Computer Hardware Specifications
1. Processor (CPU): Intel Core i3-7100 @ 3.9GHz (4 Threads)
2. Monitor: 1366x768 Resolution
3. GPU (Graphics): NVIDIA GeForce GT 630 (2GB VRAM + 4GB Shared Memory)
4. RAM: 8GB DDR4
5. Storage: 223GB SSD (Corsair Force LE) + 931GB HDD (WD Blue)

B. Hardware for Microcontrollers and IoT
1. NodeMCU ESP8266: Used as the main microcontroller, which also supports Wi-Fi connectivity. It processes data from sensors and sends it to the IoT platform.
2. Infrared (IR) Proximity Sensor FC-51: Detects the presence of vehicles in parking spaces based on infrared light reflection. This sensor is effective over short distances and is quite stable in indoor environments.
3. Buzzer: Used as an audio output device to provide an audible warning. For example, it sounds when a vehicle approaches a sensor, or when all parking spaces are full.
4. Adapter: Used as a power source (usually 5V) to continuously support the system without relying on batteries.
5. Breadboard: Used to temporarily assemble circuits without soldering, very useful in the testing and prototype development phases of a system.

C. Software
1. Arduino IDE: To program the microcontroller.
2. IoT Platform: Firebase (for web-based monitoring).
3. Programming Language: C++ for the microcontroller, JavaScript/PHP for the web-based dashboard.
4. Database: MySQL.

2.3. Research Procedure

A. Needs Analysis

A preliminary study was conducted to identify manual parking issues, such as:
1. No real-time information.
2. Users have difficulty finding empty slots.

Determining system features:
1. Automatic detection of parking slot status.
2. Real-time information via application/website.
3. Parking space usage statistics.

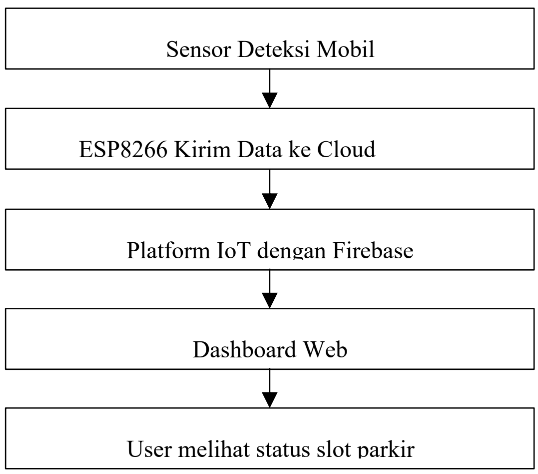


Fig. 1: Flowchart of IoT-Based Parking Slot Monitoring System

The figure illustrates the workflow of an Internet of Things (IoT)-based parking slot monitoring system. The system initiates with vehicle detection using ultrasonic or infrared sensors installed in each parking slot to identify occupancy status (vacant/occupied). The sensor data is then transmitted to an ESP8266 microcontroller module serving as the communication interface.

Subsequently, the detection data is wirelessly transmitted via HTTP protocol to cloud services utilizing Wi-Fi connection to ensure reliable and efficient data delivery. On the server side, the data is processed and stored in Firebase Realtime Database, enabling instant information synchronization along with user authentication management and parking status updates.

The parking availability information is then visualized through a responsive web-based dashboard accessible to end-users via web browsers with cross-platform compatibility support. Thus, this system provides real-time parking occupancy information, facilitating more efficient decision-making while optimizing parking space utilization and reducing vehicle search time.

B. System Design

1. Block Diagram

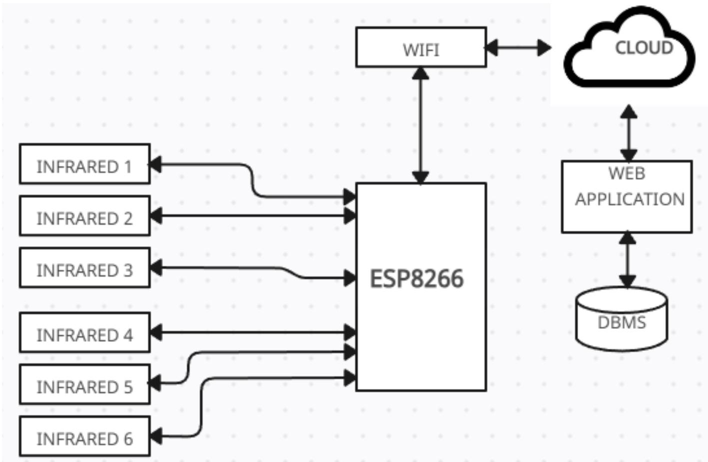


Fig. 2: Block Diagram

The image above shows a block diagram of an automatic parking monitoring system using the ESP8266 microcontroller and infrared sensors. This system is designed to detect the availability of parking slots in real-time and display the information through a web application.

System Components and Workflow:

a. Infrared Sensors (Infrared 1 – 6):
There are six infrared sensors installed in each parking slot. These sensors function to detect the presence of a vehicle. When a car is parked, the sensor sends a signal to the microcontroller.
b. ESP8266:
The ESP8266 module acts as the central unit for processing and transmitting data. It receives inputs from all infrared sensors, processes them, and sends the information to the cloud via a WiFi connection.
c. WiFi and Cloud:
The ESP8266 connects to a WiFi network to transmit data to the cloud. The cloud serves as temporary storage or as a bridge to the web application.
d. DBMS (Database Management System):
The data sent from the cloud is stored in the database management system. The DBMS keeps track of the status of each parking slot (occupied or vacant).
e. Web Application:
The web application retrieves data from the database and displays it in a user interface (UI). Users can monitor the availability of parking slots online through this application.

This system is highly suitable for use in campus parking areas, shopping centers, or other public spaces to help users efficiently and in real-time find available parking slots.

2. Pseucode Program

a. ESP8266 Pseudocode Program to Read Sensors and Send Data via Wi-Fi

```
Begin
SET pinSensor = D2
Set Status Slot = "Available"
Initialize Wi-Fi with SSID and Password
Connect to Wi-Fi network

Loop Forever:
Read valueSensor from pinSensor
IF valueSensor indicates vehicle Detected Then
```

```
Status Slot = "Full"
Turn On buzzer
Else
Status Slot = "Available"
Turn Off buzzer
End If

Create JSON payload containing statusSlot
Send data to IoT server (via HTTP)

Wait 2 seconds
End Loop
End
```

b. Pseudocode for a Dashboard Application/Web to Display Slot Status

```
Begin
Connection to a database or IoT server

Loop Every 3 seconds:
Fetch StatusSlot from database or API

If StatusSlot == "Full" THEN
Display a Red slot box with the text "FULL"
Else
Display a Green slot box with the text "AVAILABLE"
End If
End Loop
End
```

3. Design Hardware

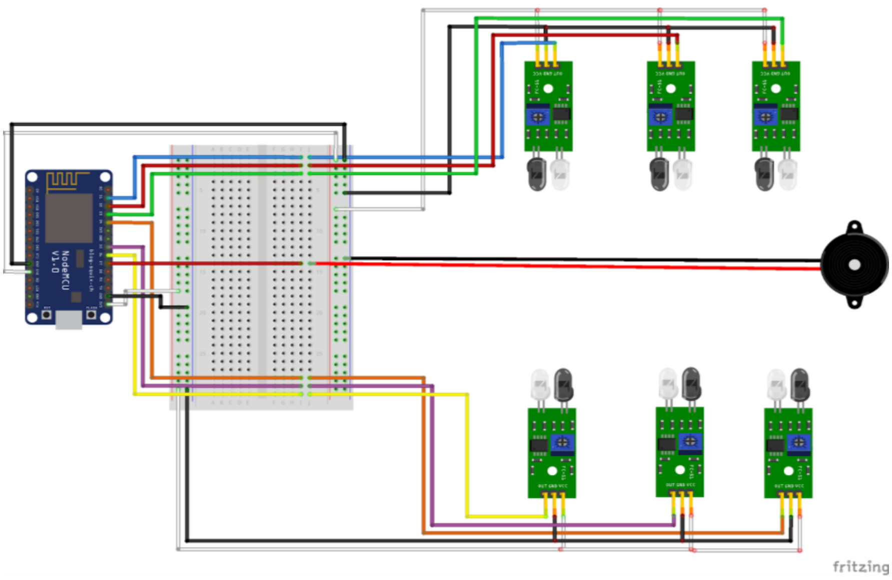


Fig. 3: Design Hardware

4. Design Database

Table 1 : Data Booking

Field Name	Data Type	Description
Id	int(11)	Primary key, auto increment
user_id	int(11)	Foreign key ke tabel users(id)
slot_id	int(11)	Foreign key ke tabel slots(id)
booking_time	datetime	Order time
Code	varchar(10)	Booking verification code
status	enum	Booking status: booked, checked_in, cancelled

Table 2. Parking Block

Field Name	Data Type	Description
Id	int(11)	Primary key, auto increment
slot_number	varchar(10)	Parking slot name/number (misal: A1, B2)
status	enum	Slot status: available, booked, occupied

Table 3. Data User

Field Name	Data Type	Description
Id	int(11)	Primary key, auto increment
username	varchar(50)	Username unique
password	varchar(255)	Password (di hash)

3. Result and Discussion

The IoT-based parking monitoring system was successfully implemented by integrating NodeMCU ESP8266, FC-51 infrared sensors, and a PHP and MySQL-based server. This implementation aims to provide real-time parking space availability information through an interactive and user-friendly web interface.

On the hardware side, IR sensors were installed on each parking block to detect the presence of vehicles. The detected data is sent via ESP8266 to the server using a WiFi connection and HTTP requests. The software supports monitoring display, login, registration, user dashboard, parking spot booking feature, and booking list.

3.1. Booking Feature Implementatio

The developed parking management system applies a secure user authentication mechanism, in which registration and login are mandatory prerequisites before users can access the parking slot booking feature. This two-step verification aims to improve user accountability and prevent unauthorized access to the system.

The main feature of this system is the parking slot booking module, which is directly integrated with the MySQL database and synchronized in real time with the hardware components in the field. Booking is done through an intuitive dropdown menu interface, allowing users to easily select available parking slots. Once selected, the user confirms the reservation through a transaction process validated by the system.

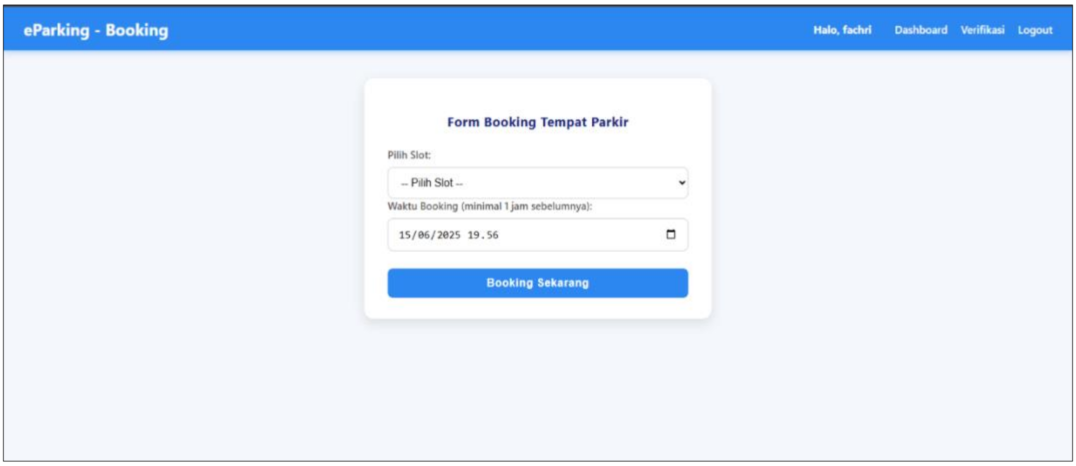


Fig. 4 : Parking Slot Booking Page Display

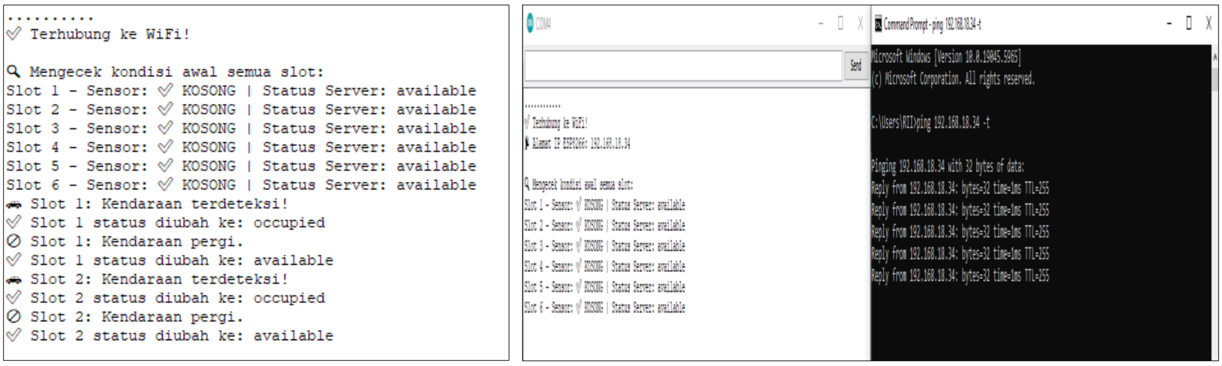
Once the booking is successful, the system automatically performs several key functions, including:

a. Updating the slot status to "BOOKED" in the database,
b. Activating hardware indicators such as LEDs or information boards at the physical parking location,
c. Enforcing a locking mechanism to prevent the same slot from being booked again by other users,
d. Generating a transaction log with a timestamp as part of system auditing and tracking.

Real-time data synchronization ensures that any change in slot status is immediately forwarded to all system components. In addition, hardware integration enables physical verification of reservations that have been confirmed digitally. This approach creates a dual-layer confirmation system (digital and physical) that effectively prevents double booking or improper slot allocation.

3.2. Sensor and Connectivity Testing

Comprehensive testing was conducted on both hardware and software. IR sensors were tested using miniature car objects to verify detection accuracy. The results showed that the sensors were able to detect vehicle presence in real time with a fast response time, as evidenced in sections figure 5 a and b.



a b

Fig. 5 : a. Testing All Sensors, b. ESP8266 Connection Test Results

The system also demonstrated connection stability, where the ESP8266 successfully connected to the WiFi network and continuously transmitted data to the server, as shown in Figure 6. Additionally, two-way communication testing was performed: from ESP8266 to the website and vice versa. The system was able to provide feedback on booking status by activating a buzzer when there was a usage violation.

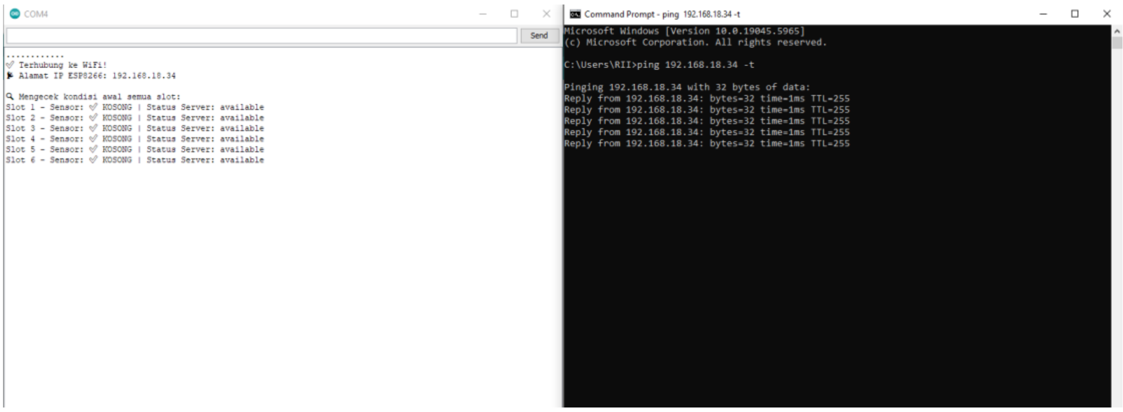


Fig. 6 : Test Result from Website to ESP8266

3.3. Evaluation Based on Test Table

Testing was carried out on all parking blocks, including blocks A1 to A3 and B1 to B3. The FC-51 IR sensor was able to detect vehicles with high speed and accuracy. Responses from the ESP8266 were sent via HTTP to the server using a WiFi connection, and statuses were updated in real-time on the web page. All results are recorded in Table 4.

Table 4 : Parking Block Testing

Parking Block	Response Time (ms)	Sensor Status	Website Condition	Buzzer Status
A1	2	OCCUPIED	RED (OCCUPIED)	INACTIVE
A2	4	OCCUPIED	RED (OCCUPIED)	INACTIVE
A3	3	OCCUPIED	RED (OCCUPIED)	INACTIVE
B1	14	OCCUPIED	YELLOW (BOOKED)	ACTIVE
B2	21	OCCUPIED	RED (OCCUPIED)	INACTIVE
A1	2	OCCUPIED	RED (OCCUPIED)	INACTIVE

From these test results:

a. Response times ranged from 2 ms to 21 ms, indicating the system operates very quickly.
b. All sensors showed "OCCUPIED" status when an object was placed, and this was accurately displayed on the website (RED if occupied, YELLOW if booked).
c. Buzzer activation only occurred if a vehicle was parked in a block that had been booked but not yet verified, as seen in Block B1.

Based on the test tables, it can be concluded that the integration of sensor systems, microcontrollers, and web interfaces functions harmoniously. The system not only detects actual conditions but also provides responsive alerts when parking rule violations occur.

3.4 System Functionality Analysis

The system demonstrated good performance in three main aspects:

1. **Real-Time Monitoring**
   Sensor data is sent and displayed directly, allowing users to know the parking status without delay.

2. **Online Booking**
   The booking feature shown in Figure 4 allows users to book parking slots through a user-friendly interface.
3. **Automatic Feedback**
   As shown in Figure 6, the system responds to field conditions by activating a buzzer if a vehicle parks without prior verification, supporting security and order.

3.5 Implementation Result

In the implementation phase, the IoT-based E-Parking system was tested using a miniature parking lot consisting of six parking slots divided into two rows, namely row A (A1, A2, A3) and row B (B1, B2, B3). Each slot was equipped with an infrared (IR) FC-51 sensor installed at the center of the slot to detect the presence of vehicles. These sensors were connected to the NodeMCU ESP8266, which functioned as the main controller as well as the data transmitter to the Firebase-based server in real time.

The first figure illustrates the initial condition of the system when all parking slots are empty. Each sensor detected a "no vehicle" condition, and the web dashboard displayed all slots with a green (AVAILABLE) status. The second figure shows the condition when all slots are fully occupied by miniature cars. All sensors successfully detected vehicle presence, transmitted data to the NodeMCU ESP8266, and updated the database in real time. The web interface displayed a red (OCCUPIED) status for all slots, while the buzzer was activated for any slot that was reserved but used without verification.

This implementation demonstrated that the system can detect parking slot status quickly (response time of 2–21 ms) and update information accurately on the web dashboard. The integration of hardware (sensors + NodeMCU) and software (web dashboard + MySQL database) worked harmoniously, supporting real-time monitoring, slot booking, and violation alerts.

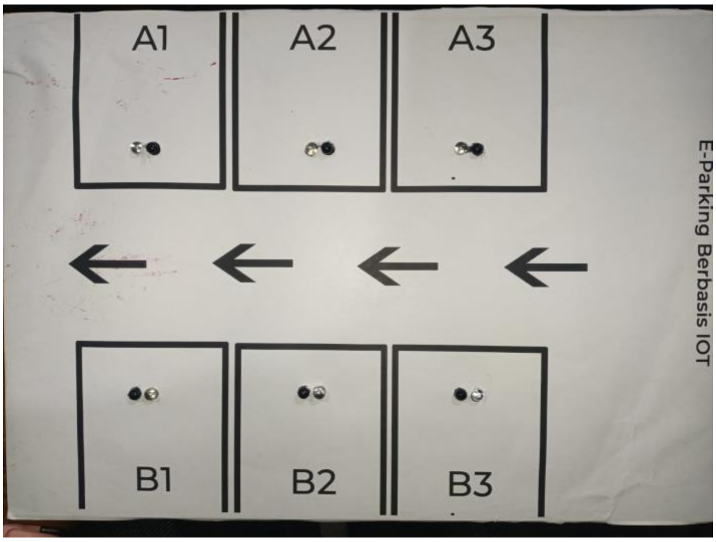


Fig. 7: Initial condition of the system when all slots are empty.

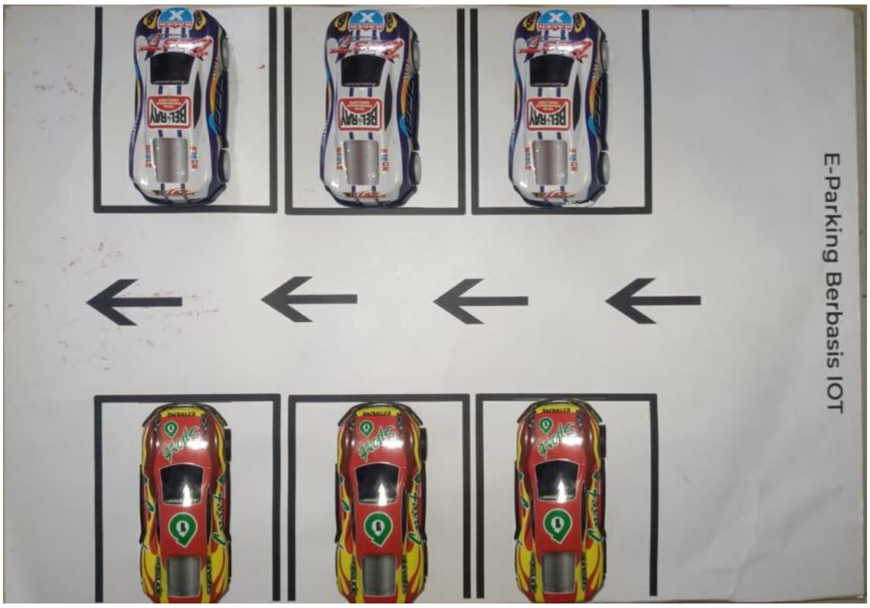


Fig. 8: Condition of the system when all slots are occupied.

4. Conclusion

This research successfully developed and implemented an IoT-based e-parking monitoring system using the NodeMCU ESP8266 microcontroller and FC-51 infrared sensors, which integrates with a web-based interface powered by PHP and MySQL. The system is capable of real-time vehicle detection, data transmission, and visualization through a cloud-connected web dashboard.

The most notable achievement is the real-time synchronization between hardware and software, enabling users to monitor and book parking slots efficiently through a user-friendly interface. The system supports secure authentication, booking validation, and physical feedback via buzzer activation if violations occur, such as unauthorized parking on a booked slot.

Sensor testing demonstrated excellent response time (as low as 2 ms) and high detection accuracy. The system maintained reliable connectivity, with seamless data exchange between ESP8266 and the server. Evaluation results confirmed the robustness of the entire architecture in real operational settings.

The novelty of this work lies in the integration of real-time monitoring with slot reservation and violation alerting in a single IoT ecosystem. This research can serve as a foundation for the development of smart parking systems in urban areas, particularly in public and commercial facilities. Future enhancements may include mobile app integration, payment systems, and predictive analytics for parking demand forecasting.

.

References

[1] F. Nahdi and H. Dhika, "Analisis Dampak Internet of Things (IoT) Pada Perkembangan Teknologi di Masa Yang Akan Datang," *INTEGER: Journal of Information Technology*, vol. 6, no. 1, Jun. 2021, doi: 10.31284/j.integer.2021.v6i1.1423.

[2] M. B. Hartanto, A. S. Putra, T. M. Fawa, and T. M. Fawaati, "Analisis Dampak Implementasi Internet Of Things (IOT) Terhadap Efisiensi Operasional Di Industri Manufaktur."

[3] F. Al-Turjman and A. Malekloo, "Smart parking in IoT-enabled cities: A survey," *Sustain Cities Soc*, vol. 49, p. 101608, Aug. 2019, doi: 10.1016/j.scs.2019.101608.

[4] S. R. A. Manalu, E. Simamora, E. U. Br Hombing, J. Maulana, K. S. A. Siahaan, and M. E. Manalu, "Analisis Faktor Penyebab Kemacetan Lalu Lintas Di Simpang Empat Unimed Mmtc Medan Analysis Of Factors Caused By Traffic Construction At The Unimed Middle Mixation Of Mmtc Medan," *JIIC : Jurnal Intelek Insan Cendekia*, vol. 1, no. 8, pp. 3421–3429, Oct. 2024, [Online]. Available: https://jicnusantara.com/index.php/jiic

[5] Z. Al Mujaddidi, D. Fardila, and E. Kurniati, "Analisia Kebutuhan Lahan Parkir (Studi Kasus Rumah Sakit Umum Daerah Sumbawa Besar)," *J_CENTAL*, no. 1, pp. 10–14, Aug. 2023.

[6] T. H. Nainggolan, N. Sebayang, N. G. De, J. Henrique, and N. Sudiasa, "Analisis Karakteristik Parkir Kendaraan Pada Area Parkir Rsud Dr. Mohamad Saleh Kota Probolinggo," *SONDIR*, vol. 5, no. 2, pp. 40–47, 2021, [Online]. Available: https://ejournal.itn.ac.id/index.php/sondir

[7] E. D. Risqullah and H. A. Rosyid, "Optimalisasi Sistem Parkir Masyarakat Teknologi dan Strategi Pemasaran Inovatif Era 5.0," *Jurnal Inovasi Teknik dan Edukasi Teknologi*, vol. 2, no. 10, pp. 461–479, 2022, doi: 10.17977/um068v1i102022p461-479.

[8] R. R. Shalhan and E. Rachmawati, "Pengenalan Ruang Kosong pada Tempat Parkir Mobil Menggunakan Convolutional Neural Network (CNN)," *e-Procceding of Engineering*, vol. 7, no. 2, pp. 8234–8245, Aug. 2020.

[9] F. B. A. Putra and L. Hayat, "Rancang Bangun Miniatur Sistem Parkir Cerdas Bertingkat Berbasis Internet of Things Menggunakan ESP32," *Jurnal Riset Rekayasa Elektro*, vol. 3, no. 1, Jun. 2021, doi: 10.30595/jrre.v3i1.9643.

[10] A. I. Pulungan, S. Sumarno, I. Gunawan, H. S. Tambunan, and A. R. Damanik, "Rancang Bangun Sistem Parkir dan Ketersediaan Slot Parkir Otomatis Menggunakan Arduino," *Jurnal Ilmu Komputer dan Informatika*, vol. 2, no. 2, pp. 127–136, Jun. 2022, doi: 10.54082/jiki.33.