

Sistem Pengendalian dan Monitoring Waktu Rental PlayStation Berbasis Internet of Things Menggunakan ESP32 dan RF433 MHz

Ahmad Alfian Ertiansyah^{1*}, Mohamad Irwan Afandi²

¹ Program Studi Sistem Informasi, Fakultas Ilmu Komputer, UPN “Veteran” Jawa Timur

² Program Studi Sistem Informasi, Fakultas Ilmu Komputer, UPN “Veteran” Jawa Timur

*E-mail: 21082010054@student.upnjatim.ac.id

Abstrak

Rental PlayStation umumnya masih menjalankan proses billing secara manual, di mana operator menyalakan televisi, mengatur timer, dan mencatat durasi bermain secara manual sehingga rawan kesalahan pencatatan dan kecurangan. Penelitian ini bertujuan merancang dan mengimplementasikan sistem pengendalian dan monitoring waktu rental PlayStation berbasis Internet of Things (IoT) yang mampu mengendalikan perangkat secara otomatis dan mencatat aktivitas rental secara akurat. Sistem dikembangkan menggunakan metode Research and Development (R&D) dengan mikrokontroler ESP32-C3 sebagai pengendali utama, modul RF433 MHz sebagai media transmisi nirkabel, relay wireless 220V sebagai aktuator, serta Firebase Realtime Database dan aplikasi web sebagai antarmuka monitoring. ESP32 membaca status meja dari Firebase melalui mekanisme polling setiap 3 detik dan menerjemahkannya menjadi perintah RF433 MHz untuk menyalakan atau mematikan televisi. Hasil pengujian menunjukkan komunikasi RF433 MHz berhasil 100% pada jarak 15 meter, rata-rata waktu respons sistem 2,59 detik, dan rata-rata selisih ketepatan timer 2,72 detik. Pengujian fungsional black-box terhadap seluruh fitur berhasil, pengujian multi-unit tidak menimbulkan interferensi, dan sistem stabil selama pengujian 120 menit. Sistem ini terbukti dapat meningkatkan efisiensi operasional dan mengurangi risiko kesalahan pencatatan pada rental PlayStation.

Kata kunci : ESP32, Firebase, Internet of Things, Rental PlayStation, RF433 MHz

Abstract

PlayStation rental businesses generally still perform billing manually, where operators turn on the television, set the timer, and record the playing duration by hand, making the process prone to recording errors and fraud. This study aims to design and implement an Internet of Things (IoT) based control and monitoring system for PlayStation rental time that can control devices automatically and record rental activities accurately. The system was developed using the Research and Development (R&D) method with an ESP32-C3 microcontroller as the main controller, an RF433 MHz module as the wireless transmission medium, a 220V wireless relay as the actuator, and Firebase Realtime Database together with a web application as the monitoring interface. The ESP32 reads the table status from Firebase through a polling mechanism every 3 seconds and translates it into RF433 MHz commands to switch the television on or off. The test results show that RF433 MHz communication succeeded 100% at a distance of 15 meters, with an average system response time of 2.59 seconds and an average timer accuracy deviation of 2.72 seconds. Black-box functional testing of all features was successful, multi-unit testing produced no interference, and the system remained stable during a 120-minute test. This system is proven to improve operational efficiency and reduce the risk of recording errors in PlayStation rentals.

Keywords: ESP32, Firebase, Internet of Things, PlayStation Rental, RF433 MHz

PENDAHULUAN

Perkembangan teknologi *Internet of Things* (IoT) telah mendorong transformasi berbagai sektor usaha dari proses manual menuju otomatisasi. Salah satu bidang usaha jasa yang berkembang pesat adalah rental PlayStation (PS), yang memiliki dua proses bisnis utama

yaitu penyewaan konsol berdasarkan durasi bermain serta penjualan konsumsi. Namun, sebagian besar rental PS masih menjalankan proses billing secara manual, baik dalam mengatur durasi bermain maupun mencatat transaksi [1].

Hal serupa terjadi pada Mapian *PlayStation*, sebuah rental PS di Ponorogo yang menjadi objek penelitian ini. Proses billing masih mengandalkan pencatatan manual, di mana operator menyalakan televisi, mengatur timer, serta mencatat durasi bermain ke buku atau *notepad* ponsel. Cara manual ini membutuhkan waktu, rawan kesalahan manusia ketika rental ramai, serta berpotensi menimbulkan kecurangan karena tidak ada pencatatan otomatis yang memvalidasi transaksi secara *real-time* [8], [9].

Perkembangan perangkat IoT membuka peluang otomatisasi operasional rental PS. Mikrokontroler ESP32 memiliki konektivitas *WiFi* bawaan yang memungkinkan perangkat menerima perintah dari sistem web, sedangkan modul RF433 MHz dapat mengirimkan kode perintah ke relay *wireless* sebagai saklar untuk menyalakan atau mematikan televisi pada tiap meja. Beberapa penelitian terdahulu telah memanfaatkan ESP32 dan komunikasi nirkabel untuk pengendalian perangkat [4], [6], [14], [16], namun sebagian besar masih berupa prototipe dan belum menerapkan mekanisme pencatatan durasi serta sistem billing untuk layanan berbasis sewa.

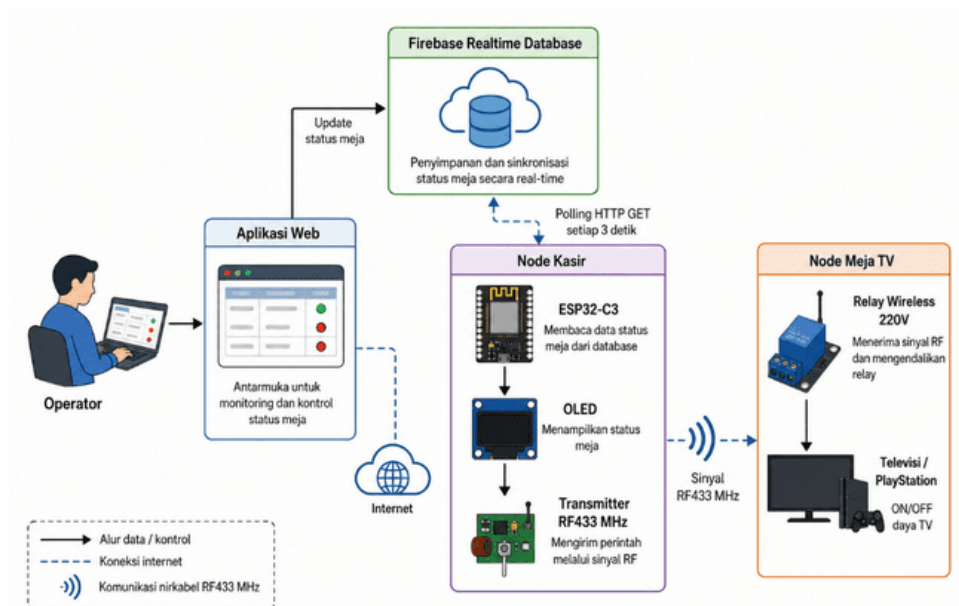
Berdasarkan permasalahan tersebut, penelitian ini bertujuan merancang dan mengimplementasikan sistem pengendalian dan monitoring waktu rental *PlayStation* berbasis IoT menggunakan ESP32 dan RF433 MHz yang dapat mengendalikan perangkat secara otomatis dan mencatat aktivitas rental secara akurat untuk mendukung rekapitulasi transaksi yang lebih efisien.

METODE PENELITIAN

Penelitian ini menggunakan metode *Research and Development* (R&D), yaitu metode penelitian yang digunakan untuk menghasilkan produk tertentu dan menguji kelayakan produk tersebut [10], [11]. Tahapan penelitian terdiri dari analisis kebutuhan, perancangan sistem, implementasi (pengembangan perangkat keras dan perangkat lunak), pengujian sistem, serta penarikan kesimpulan.

Arsitektur Sistem

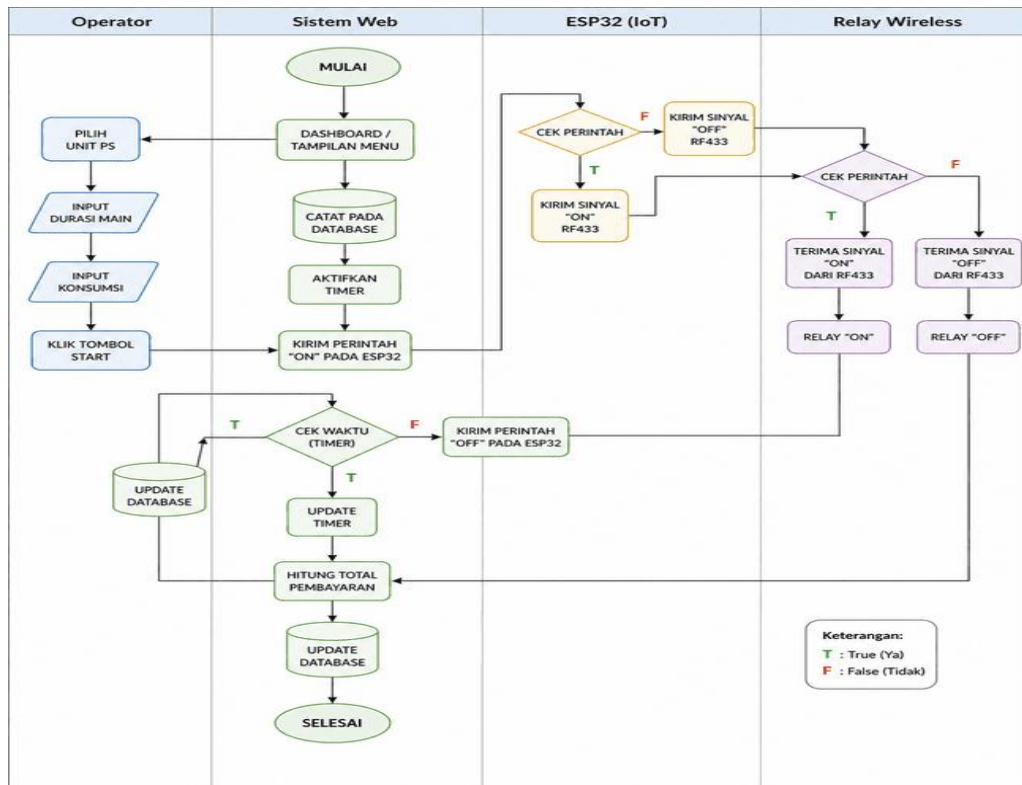
Sistem terdiri dari dua node yang berkomunikasi secara nirkabel. Node kasir terdiri dari mikrokontroler ESP32-C3 yang terhubung dengan modul transmitter RF433 MHz dan layar OLED. Node meja TV terdiri dari relay *wireless* 220V yang berfungsi sebagai saklar daya televisi. Operator mengendalikan sistem melalui aplikasi web yang terhubung dengan *Firestore Realtime Database*. Ketika status meja pada *Firestore* berubah, ESP32 membacanya melalui mekanisme *polling* setiap 3 detik melalui *HTTP GET*, kemudian mengirimkan kode RF433 MHz ke relay untuk menyalakan atau mematikan televisi. Arsitektur sistem secara keseluruhan ditunjukkan pada Gambar 1.



Gambar 1. Arsitektur sistem IoT rental PlayStation

Alur Kerja Sistem

Alur kerja sistem dimulai ketika operator memilih meja dan paket durasi pada aplikasi web, lalu menekan tombol Start Billing. Aplikasi web mengubah status meja menjadi aktif serta menyimpan waktu mulai dan waktu selesai ke *Firestore Realtime Database*. ESP32 yang melakukan *polling* setiap 3 detik membaca perubahan status tersebut dan mengirimkan kode RF433 MHz untuk menyalakan relay sehingga televisi menyala. Selama sesi berjalan, aplikasi web menampilkan hitung mundur durasi. Ketika waktu habis atau pembayaran dilakukan, status meja diubah menjadi tidak aktif, ESP32 mengirimkan perintah RF untuk mematikan relay, dan data transaksi disimpan ke dalam riwayat. Keseluruhan alur kerja sistem ditunjukkan pada Gambar 2.



Gambar 2. Flowchart alur kerja sistem billing rental PlayStation

Perangkat Keras dan Perangkat Lunak

Perangkat keras yang digunakan meliputi ESP32-C3 sebagai pengendali utama, modul RF433 MHz STX882 sebagai transmitter nirkabel dengan konfigurasi 24-bit protokol 1 dan *pulse length* 350 mikrodetik, relay *wireless* 220V sebagai aktuator, serta OLED 0,96 inci sebagai penampil status. Perangkat lunak yang digunakan meliputi Arduino IDE untuk pemrograman ESP32, *library* RCSwitch dan Adafruit SH110X, Firebase *Realtime Database* sebagai basis data, serta aplikasi web berbasis HTML, CSS Bootstrap, dan JavaScript sebagai antarmuka monitoring.

Metode Pengujian

Pengujian dilakukan melalui beberapa skenario, yaitu pengujian jarak komunikasi RF433 MHz, pengujian waktu respons sistem, pengujian ketepatan timer billing, pengujian pengendalian beberapa unit secara bersamaan (*simultaneous*), pengujian tampilan OLED, pengujian fungsional aplikasi web menggunakan metode *black-box testing*, serta pengujian stabilitas sistem. Pengujian fungsional menggunakan pendekatan *black-box* yang berfokus pada kesesuaian keluaran sistem terhadap masukan yang diberikan.

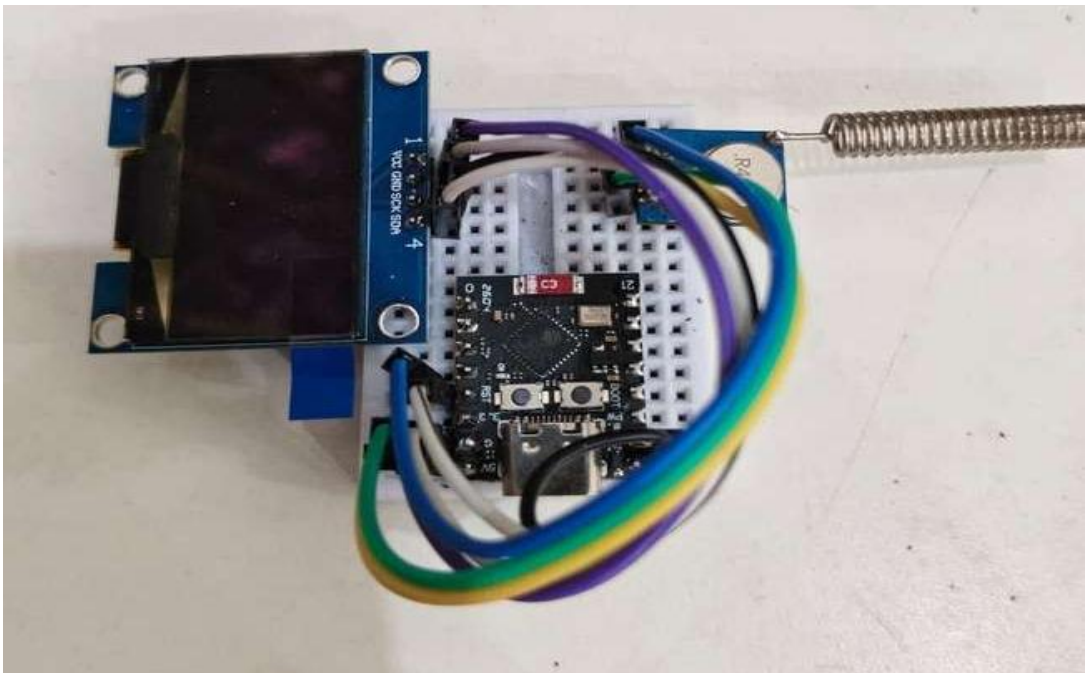
HASIL DAN PEMBAHASAN

Bagian ini menyajikan hasil implementasi sistem, baik perangkat keras maupun perangkat lunak, serta hasil pengujian terhadap setiap aspek fungsional dan kinerja sistem. Pembahasan diakhiri dengan perbandingan antara sistem manual dan sistem yang dikembangkan.

Hasil Implementasi Perangkat Keras

Sistem terdiri dari dua node yang berkomunikasi secara nirkabel. Node kasir terdiri dari mikrokontroler ESP32-C3, modul transmitter RF433 MHz STX882, dan layar OLED 0,96 inci

yang berfungsi mengirimkan perintah serta menampilkan status sistem. Hasil implementasi node kasir ditunjukkan pada Gambar 3.



Gambar 3. Hasil implementasi node kasir

Rangkaian node kasir menghubungkan ESP32-C3 dengan modul RF433 MHz dan OLED melalui konfigurasi pin sebagaimana ditunjukkan pada Tabel 1. Komunikasi dengan OLED menggunakan antarmuka I2C, sedangkan modul RF menggunakan satu jalur data untuk transmisi sinyal.

Tabel 1. Konfigurasi pin ESP32-C3

No	Pin ESP32-C3	Terhubung ke	Komponen	Fungsi
1	GPIO 4	SDA	OLED SH1106	Jalur data I2C OLED
2	GPIO 5	SCL	OLED SH1106	Jalur clock I2C OLED
3	GPIO 3	DATA	RF433 MHz TX	Transmisi sinyal RF
4	5V	VCC	OLED + RF TX	Sumber tegangan 5V
5	GND	GND	OLED + RF TX	Ground bersama

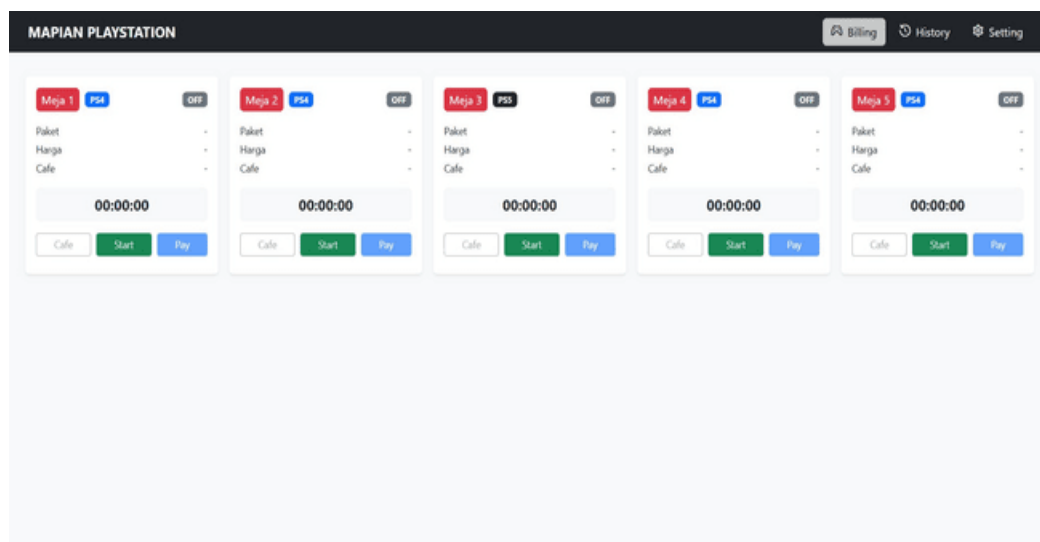
Node meja TV terdiri dari relay *wireless* 220V yang dipasang pada jalur daya televisi dan berfungsi sebagai aktuator untuk menyalakan atau mematikan televisi sesuai perintah RF yang diterima. Hasil pemasangan relay ditunjukkan pada Gambar 4.



Gambar 4. Hasil implementasi relay wireless pada node meja TV

Hasil Implementasi Perangkat Lunak

Perangkat lunak terdiri dari aplikasi web monitoring dan basis data Firebase *Realtime Database*. Aplikasi web menyediakan antarmuka bagi operator untuk memulai sesi, menambah waktu, mencatat pesanan, dan melakukan pembayaran. Tampilan halaman *billing* aplikasi web ditunjukkan pada Gambar 5.



Gambar 5. Halaman billing aplikasi web monitoring

Data sistem disimpan dalam Firebase *Realtime Database* yang terdiri dari beberapa node utama. Node *billing* menyimpan status dan data sesi tiap meja, dan menjadi acuan utama ESP32 dalam menentukan perintah RF. Struktur node Firebase ditunjukkan pada Tabel 2.

Tabel 2. Struktur node Firebase Realtime Database

Node	Field Utama	Fungsi
billing	status, startTime, endTime, tarifNama, grandTotal	Menyimpan status dan data sesi tiap meja; status menjadi pemicu perintah RF
tarif	nama, durasi, harga	Menyimpan daftar paket tarif rental
cafe	nama, harga, stok	Menyimpan daftar menu konsumsi
history	meja, billingTotal, cafeTotal, grandTotal, tanggal	Menyimpan riwayat transaksi yang telah dibayar
esp32	ip, status	Menyimpan alamat IP dan status koneksi ESP32

Setiap meja memiliki kode RF unik untuk perintah ON dan OFF dengan panjang 24-bit, sehingga perintah hanya memicu relay pada meja yang dituju dan tidak terjadi interferensi. Pemetaan kode RF tiap meja ditunjukkan pada Tabel 3.

Tabel 3. Pemetaan kode RF433 MHz tiap meja

ID Meja	Kode RF ON	Kode RF OFF
meja1	1001001	1001000
meja2	1002001	1002000
meja3	1003001	1003000
meja4	1004001	1004000
meja5	1005001	1005000

Implementasi Logika Program ESP32

ESP32 membaca data dari Firebase melalui mekanisme *polling* setiap 3 detik menggunakan *HTTP GET*. Sinyal RF hanya dikirim ketika terjadi perubahan status meja, sehingga menghindari pengiriman berulang yang tidak perlu. Cuplikan logika inti deteksi perubahan status dan pengiriman sinyal RF ditunjukkan sebagai berikut.

```
for (JsonPair meja : root) {
    const char* id = meja.key().c_str();
    bool st = meja.value()["status"] | false;
    if (prevStatus[idx] != st) { // status berubah
        sendRF(id, st);         // kirim RF ON/OFF
        prevStatus[idx] = st;
    }
    idx++;
}
```

Fungsi *sendRF()* memilih kode RF yang sesuai dengan meja dan status yang diminta, kemudian mengirimkannya melalui modul RF433 MHz dengan panjang 24-bit.

```
void sendRF(const char* id, bool status) {
    for (int i = 0; i < RF_COUNT; i++) {
        if (strcmp(rf[i].id, id) == 0) {
            unsigned long c = status ? rf[i].on : rf[i].off;
            rfSwitch.send(c, 24);
            return;
        }
    }
}
```

Hasil Pengujian Jarak Komunikasi RF433 MHz

Pengujian jarak dilakukan pada jarak 15 meter (jarak terjauh lingkungan pengujian) dalam kondisi *Line of Sight* (LOS) dan *Non-Line of Sight* (NLOS), masing-masing lima kali untuk perintah ON dan OFF. Hasilnya ditunjukkan pada Tabel 4.

Tabel 4. Hasil pengujian jarak komunikasi RF433 MHz

No	Jarak	Kondisi	Percobaan ON	Percobaan OFF	Status
1	15 m	LOS	5/5 berhasil	5/5 berhasil	Berhasil
2	15 m	NLOS	5/5 berhasil	5/5 berhasil	Berhasil

Berdasarkan Tabel 4, komunikasi RF433 MHz berhasil mengendalikan relay dengan tingkat keberhasilan 100% pada jarak 15 meter, baik pada kondisi LOS maupun NLOS. Hal

ini menunjukkan sinyal masih dapat menembus halangan fisik seperti dinding pada jarak operasional rental.

Hasil Pengujian Waktu Respons Sistem

Pengujian waktu respons mengukur lama waktu sejak operator menekan perintah pada aplikasi web hingga relay merespons secara fisik. Pengujian dilakukan 10 kali untuk perintah ON dan 10 kali untuk OFF. Hasil pengujian ditunjukkan pada Tabel 5.

Tabel 5. Hasil pengujian waktu respons sistem

No	Respons ON (detik)	Respons OFF (detik)
1	1,53	1,33
2	3,28	1,71
3	2,36	1,63
4	1,88	2,01
5	2,18	3,41
6	7,18	0,98
7	1,72	1,99
8	2,84	1,41
9	6,86	2,31
10	3,76	1,51

Tabel 6. Rekapitulasi waktu respons sistem

Perintah	Jumlah	Tercepat	Terlama	Rata-rata
ON	10	1,53 dtk	7,18 dtk	3,36 dtk
OFF	10	0,98 dtk	3,41 dtk	1,83 dtk
Keseluruhan	20	0,98 dtk	7,18 dtk	2,59 dtk

Berdasarkan Tabel 6, rata-rata waktu respons keseluruhan adalah 2,59 detik. Variasi waktu respons, termasuk nilai tertinggi 7,18 detik, dipengaruhi oleh interval *polling* Firebase 3 detik serta variasi latensi jaringan saat permintaan *HTTP*.

Hasil Pengujian Ketepatan Timer Billing

Pengujian ketepatan timer mengukur selisih antara target waktu selesai dengan waktu aktual relay padam pada empat variasi durasi, masing-masing dua kali percobaan. Batas toleransi ditetapkan maksimal 5 detik, mengacu pada interval *polling* 3 detik ditambah waktu *HTTP* dan transmisi RF. Hasil dan rekapitulasinya ditunjukkan pada Tabel 7 dan Tabel 8.

Tabel 7. Hasil pengujian ketepatan timer billing

No	Durasi	Target (dtk)	Aktual (dtk)	Selisih (dtk)	Status
1	5 mnt	300	301,52	1,52	Berhasil
2	5 mnt	300	302,44	2,44	Berhasil
3	10 mnt	600	604,55	4,55	Berhasil
4	10 mnt	600	602,89	2,89	Berhasil
5	15 mnt	900	903,16	3,16	Berhasil
6	15 mnt	900	902,32	2,32	Berhasil
7	30 mnt	1800	1803,48	3,48	Berhasil
8	30 mnt	1800	1801,38	1,38	Berhasil

Tabel 8. Rekapitulasi ketepatan timer billing

Durasi	Min (dtk)	Maks (dtk)	Rata-rata (dtk)
5 menit	1,52	2,44	1,98

Durasi	Min (dtk)	Maks (dtk)	Rata-rata (dtk)
10 menit	2,89	4,55	3,72
15 menit	2,32	3,16	2,74
30 menit	1,38	3,48	2,43
Keseluruhan	1,38	4,55	2,72

Seluruh percobaan menghasilkan selisih di bawah toleransi 5 detik dengan rata-rata keseluruhan 2,72 detik. Selisih ini tidak berdampak material terhadap perhitungan biaya karena satuan tarif rental menggunakan durasi per paket.

Hasil Pengujian Simultaneous Multi-Unit

Pengujian *simultaneous* dilakukan dengan menyalakan 2 hingga 5 meja secara bersamaan untuk memastikan tidak terjadi interferensi antar relay. Hasilnya ditunjukkan pada Tabel 9.

Tabel 9. Hasil pengujian simultaneous multi-unit

No	Jumlah Meja Aktif	Interferensi	Status
1	2 meja bersamaan	Tidak ada	Berhasil
2	3 meja bersamaan	Tidak ada	Berhasil
3	4 meja bersamaan	Tidak ada	Berhasil
4	5 meja bersamaan	Tidak ada	Berhasil

Seluruh pengujian berhasil tanpa interferensi karena setiap meja memiliki kode RF unik, sehingga perintah hanya memicu relay pada meja yang dituju.

Hasil Pengujian Tampilan OLED

Pengujian tampilan OLED dilakukan untuk memastikan informasi yang ditampilkan sesuai dengan kondisi kerja sistem. Hasilnya ditunjukkan pada Tabel 10.

Tabel 10. Hasil pengujian tampilan OLED

No	Kondisi	Hasil	Status
1	ESP32 masuk AP Mode	Menampilkan mode konfigurasi Wi-Fi dan IP	Berhasil
2	ESP32 terhubung Wi-Fi	Menampilkan status koneksi dan IP ESP32	Berhasil
3	Meja aktif	Menampilkan status meja aktif	Berhasil
4	Meja tidak aktif	Menampilkan status meja tidak aktif	Berhasil

Hasil Pengujian Fungsional (Black-Box Testing)

Pengujian fungsional menggunakan metode *black-box testing* terhadap 27 skenario yang mencakup seluruh fitur aplikasi web. Ringkasan hasil per fitur ditunjukkan pada Tabel 11.

Tabel 11. Ringkasan hasil pengujian black-box

Fitur yang Diuji	Jumlah Skenario	Hasil
Start Billing	4	Berhasil
Tambah Waktu	4	Berhasil
Pesanan Cafe	4	Berhasil
Pembayaran	4	Berhasil
History	5	Berhasil
Setting	6	Berhasil
Total	27	100% Berhasil

Seluruh 27 skenario berhasil dijalankan dengan keluaran sesuai yang diharapkan, menunjukkan bahwa seluruh fitur fungsional sistem telah terimplementasi dengan benar.

Hasil Pengujian Stabilitas Sistem

Pengujian stabilitas dilakukan untuk mengetahui ketahanan sistem dalam operasional berkelanjutan pada durasi 30, 60, dan 120 menit. Hasilnya ditunjukkan pada Tabel 12.

Tabel 12. Hasil pengujian stabilitas sistem

No	Durasi	Kondisi	Catatan Observasi
1	30 menit	Stabil	Tidak restart, WiFi stabil, polling normal
2	60 menit	Stabil	Tidak restart, WiFi stabil, polling normal
3	120 menit	Stabil	Tidak restart, WiFi stabil, polling normal

Selama pengujian, sistem berjalan stabil tanpa *restart* maupun gangguan koneksi. Mekanisme *auto-reconnect* WiFi dan penyimpanan konfigurasi pada memori non-volatile menjaga sistem tetap terhubung secara otomatis.

Perbandingan Sistem Manual dan Sistem IoT

Untuk menilai kontribusi sistem terhadap operasional rental, dilakukan perbandingan antara proses manual sebelumnya dengan sistem IoT yang dikembangkan, sebagaimana ditunjukkan pada Tabel 13.

Tabel 13. Perbandingan sistem manual dan sistem IoT

Aspek	Sistem Manual	Sistem IoT
Penyalan TV	Manual via saklar	Otomatis via relay RF433 MHz
Pencatatan waktu	Buku/notepad	Otomatis (Unix timestamp)
Penghitungan durasi	Manual, rawan keliru	Otomatis, selisih $\pm 2,72$ dtk
Pemadaman waktu habis	Manual operator	Otomatis via timer sistem
Pesanan cafe	Dicatat terpisah	Terintegrasi satu transaksi
Rekap pendapatan	Manual	Otomatis (filter tanggal)
Potensi kecurangan	Tinggi	Rendah (tercatat di cloud)
Skalabilitas	Terbatas	Modular hingga 20 meja

Berdasarkan Tabel 13, sistem IoT mengotomatisasi proses yang sebelumnya manual, sehingga mengurangi beban kerja operator, menekan risiko kesalahan pencatatan, dan mengurangi potensi kecurangan karena seluruh transaksi tercatat di *cloud*. Pemutusan arus oleh relay juga dilakukan pada kondisi tegangan jaringan lokal yang stabil sehingga tidak berdampak signifikan terhadap perangkat televisi, sebagaimana terbukti dari operasional Mapian *PlayStation* selama kurang lebih dua tahun tanpa kerusakan perangkat.

SIMPULAN

Penelitian ini berhasil merancang dan mengimplementasikan sistem pengendalian dan monitoring waktu rental *PlayStation* berbasis *Internet of Things* menggunakan ESP32-C3 dan RF433 MHz. Sistem mampu mengendalikan televisi secara otomatis melalui relay *wireless* serta mencatat waktu bermain dan transaksi secara otomatis melalui integrasi *Firestore Realtime Database* dengan aplikasi web. Hasil pengujian menunjukkan komunikasi RF433 MHz berhasil 100% pada jarak 15 meter, rata-rata waktu respons sistem 2,59 detik, dan rata-rata selisih ketepatan timer 2,72 detik yang berada di bawah toleransi. Pengujian fungsional, multi-unit, dan stabilitas seluruhnya menunjukkan hasil yang baik. Dengan demikian, sistem terbukti dapat meningkatkan efisiensi operasional serta mengurangi risiko kesalahan pencatatan dan kecurangan pada rental *PlayStation*. Untuk pengembangan selanjutnya, disarankan penambahan mekanisme pengiriman ulang sinyal RF secara otomatis, penggunaan relay dua

arah atau sensor arus sebagai umpan balik, penerapan autentikasi pada aplikasi web, migrasi mekanisme *polling* menjadi *streaming listener* agar respons mendekati *real-time*, serta pengembangan aplikasi *mobile* sebagai antarmuka alternatif.

DAFTAR PUSTAKA

- [1] A. Zilham Fadillah dan R. Gunawan, "Potensi IoT dalam Industri 4.0," 2024.
- [2] H. Kurnia, "Implementasi IoT pada Sistem Monitoring Suhu dan Kelembaban Menggunakan ESP32, Firebase dan Kodular," 2025.
- [3] D. Feriyanto, V. M. Tandayu, Y. Afrida, N. Aminudin, dan R. Ratnasari, "Identifikasi dan Pemilihan Mikrokontroler untuk Project Internet of Things (IoT)," *Aisyah Journal of Informatics and Electrical Engineering*, vol. 7, no. 1, 2025.
- [4] A. H. Awami, M. Abdulrazigh, dan M. Khairalla, "Wireless Control System for Power Source Using RF433MHz Communication and AVR Microcontroller," *International Journal of Electrical Engineering and Sustainability (IJEES)*, vol. 3, no. 1, pp. 83-94, 2025.
- [5] M. M. Alifia Qodri, N. Rahaningsih, dan R. Danar Dana, "Sistem Pengendalian Lampu Rumah dan Kantor Berbasis Internet of Things Menggunakan Mikrokontroler NodeMCU ESP8266," 2024.
- [6] W. W. Anggoro dan I. R. Widiyari, "Perancangan dan Penerapan Kendali Lampu Ruangan Berbasis IoT (Internet of Things) Android," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 8, no. 3, 2021.
- [7] A. Z. Hasibuan dan M. S. Asih, "Smart Neighborhood Alert System Berbasis IoT dan Aplikasi Android untuk Meningkatkan Respons Warga terhadap Kejadian Darurat," *Jurnal Komputer Teknologi Informasi Sistem Informasi (JUKTISI)*, vol. 4, no. 2, pp. 1461-1468, 2025.
- [8] F. A. Rohim, F. M. Dewanto, dan N. D. Saputro, "Pengembangan Aplikasi Website Rental PlayStation dengan Vue JS Menggunakan Algoritma Data Aggregation," *Jurnal Informatika Teknologi dan Sains (JINTEKS)*, vol. 7, no. 3, pp. 1192-1201, 2025.
- [9] P. A. Dzikri, V. Atina, dan J. Maulindar, "Pemodelan Sistem Informasi Rental PlayStation Berbasis Web pada Sanjaya PlayStation Menggunakan Metode Waterfall," *JEKIN - Jurnal Teknik Informatika*, vol. 4, no. 3, pp. 688-697, 2024.
- [10] L. Judijanto et al., *Metodologi Research and Development (Teori dan Penerapan Metodologi RnD)*, 1st ed. Jambi: PT. Sonpedia Publishing Indonesia, 2024.
- [11] Sugiyono, *Metode Penelitian Kuantitatif Kualitatif dan R&D*. Bandung: ALFABETA, 2011.
- [12] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, dan M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications," *IEEE Communications Surveys and Tutorials*, vol. 17, no. 4, pp. 2347-2376, 2015.
- [13] P. Sethi dan S. R. Sarangi, "Internet of Things: Architectures, Protocols, and Applications," *Journal of Electrical and Computer Engineering*, 2017.
- [14] R. K. Bangun dan I. R. Widiyari, "Implementation of an IoT Based Billing System to Optimize Management of Billiard Table Usage," *INOVTEK Polbeng Journal - Informatics Series*, vol. 9, no. 2, 2024.
- [15] Y. Natasya dan H. Santoso, "Prototipe Aplikasi Smart Lighting untuk Mengontrol Lampu Jalan Berbasis Android Menggunakan ESP32," *SIBATIK Journal*, vol. 2, no. 8, 2023.
- [16] J. W. Jokanan, A. Widodo, N. Kholis, dan L. Rakhmawati, "Rancang Bangun Alat Monitoring Daya Listrik Berbasis IoT Menggunakan Firebase," *Jurnal Teknik Elektro*, vol. 11, no. 1, 2022.
- [17] W. Waryanto, M. Mursalin, T. Aprilia, dan M. Khozin, "Implementasi Ganda Modul ESP32 untuk Kontrol Pintu Otomatis dengan Monitoring Realtime dan Autolock Terjadwal melalui Web dan Chatbot," *Jurnal Riset Rekayasa Elektro*, vol. 7, no. 1, pp. 1-10, 2025.