



**REAL TIME SYSTEM AND INTERNET OF THINGS FINAL PROJECT REPORT
DEPARTMENT OF ELECTRICAL ENGINEERING
UNIVERSITAS INDONESIA**

PARK-IR: INFRARED-BASED SOLUTION FOR SEAMLESS PARKING

GROUP A3

ALTHAF NAFI ANWAR	2106634881
BERNANDA NAUTVAL R.I.W.	2106708463
MICHAEL GUNAWAN	2106731195
RAYHAN AKBAR ARRIZKY	2106632655

PREFACE

Puji dan syukur kami panjatkan ke hadirat Tuhan Yang Maha Esa, atas rahmat dan karunia-Nya, kami dapat menyelesaikan proyek akhir untuk mata kuliah Sistem Waktu Nyata dan IoT. Adapun judul dari proyek kami adalah “PARK-IR: INFRARED-BASED SOLUTION FOR SEAMLESS PARKING” yang pada kesempatan kali ini, kamu juga ingin mengucapkan terima kasih kepada dosen mata kuliah kami, yakni Bapak Fransiskus Astha Ekadiyanto, yang telah melimpahkan ilmunya kepada kami sehingga kami bisa membuat proyek kali ini dengan baik. Tidak lupa, kami juga mengucapkan terima kasih kepada segenap asisten laboratorium Digital Laboratory, yang telah mendampingi dan membantu kami dalam penyusunan proyek ini.

Seperti yang kita ketahui bersama, di dalam era teknologi sekarang ini, Internet of Things (IoT) telah menjadi bagian dari kehidupan kita sehari-hari bahkan hingga ke hal yang tidak terduga sekalipun, yakni sektor transportasi. Laporan ini akan merangkum hasil dari proyek pengembangan sistem palang parkir menggunakan ESP32 sebagai kerangka utamanya dan LoRa untuk cadangan penghubungan yang bertujuan untuk meningkatkan efisiensi serta membawa inovasi yang praktis bagi para penggunanya perihal kebutuhan keluar - masuk lahan tempat memarkir kendaraan.

Secara garis besar, laporan ini berisi penjelasan atas alat dan bahan yang digunakan serta proses dan peran di dalamnya. Terdapat juga bagian aplikasi buatan kami yang dirancang khusus untuk mengelola id unik dari setiap pelanggan kami. Proyek yang kami susun mungkin masih kurang dari kata sempurna, oleh sebab itu kami menerima segala kritik dan saran yang membangun untuk hasil yang lebih baik lagi kedepannya. Kami berharap, proyek yang kami susun bisa berguna untuk khalayak ramai dan bisa menjadi bahan pertimbangan untuk kelancaran mata kuliah Sistem Waktu Nyata dan IoT ini.

Depok, December 9, 2023

Group A3

TABLE OF CONTENTS

CHAPTER 1.....	4
INTRODUCTION.....	4
1.1 PROBLEM STATEMENT.....	4
1.3 ACCEPTANCE CRITERIA.....	5
1.4 ROLES AND RESPONSIBILITIES.....	5
1.5 TIMELINE AND MILESTONES.....	5
CHAPTER 2.....	7
IMPLEMENTATION.....	7
2.1 HARDWARE DESIGN AND SCHEMATIC.....	7
2.2 SOFTWARE DEVELOPMENT.....	7
2.3 HARDWARE AND SOFTWARE INTEGRATION.....	8
CHAPTER 3.....	9
TESTING AND EVALUATION.....	9
3.1 TESTING.....	9
3.2 RESULT.....	9
3.3 EVALUATION.....	10
CHAPTER 4.....	11
CONCLUSION.....	11

CHAPTER 1

INTRODUCTION

1.1 PROBLEM STATEMENT

Di tengah berkembangnya infrastruktur dan tempat terbuka publik untuk tujuan rekreasi maupun aktivitas sehari-hari baik itu di kota metropolitan hingga daerah sekitar, proses pengelolaan tempat parkir kendaraan bermotor seperti mobil dan motor seringkali menjadi tantangan yang membutuhkan solusi efektif nan efisien. Sejauh ini sistem pembayaran elektronik maupun manual yang digunakan untuk melakukan pembayaran jam parkir, sering kali rentan terhadap permasalahan baik itu dari sisi keefektifitasan yang kurang baik untuk kecepatan proses pembayaran yang bisa saja menimbulkan antrian hingga kendala seperti saldo yang habis saat hendak melakukan proses pembayaran. Adapun proses pencatatan manual lainnya yang masih menggunakan sistem pembayaran tunai dan penggunaan karcis yang tentu lebih rentan lagi terhadap permasalahan lainnya seperti kehilangan karcis yang bisa berujung pada upaya pencurian maupun kerugian yang tidak perlu untuk biaya ganti rugi karcis hilang. Sejauh ini, proses yang berlangsung adalah upaya memperbanyak pembangunan sistem pembayaran parkir dari manual ke kartu elektronik di berbagai titik keluar masuk kendaraan yang justru juga meningkatkan biaya operasional.i.

Oleh sebab itu, penggeraan proyek ini bertujuan untuk merespon dinamika tren permasalahan yang sedang berkembang dalam lingkup pembayaran dan otomatisasi sistem pembayaran lahan parkir guna mengantisipasi lambatnya proses pembayaran manual hingga meminimalisir segala bentuk kerancuan maupun titik lemah yang masih ada di dalam sistem yang sedang berlangsung saat ini. Berangkat dari keterbatasan yang dimiliki oleh sistem manual maupun sistem penempelan kartu elektronik yang sedang ada saat ini, upaya transparansi maupun ketepatan jam pembayaran waktu parkir yang pada akhirnya bisa berujung pada melambatnya aktivitas di lapangan terutama bagi pihak pengelola lahan parkir seperti pemilik gedung maupun koperasi sebagai pihak ketiga yang diajak untuk bekerja sama menjadi landasan berpikir dari proyek bernama PARK-IR yang sengaja kami susun ini dengan tujuan untuk mengatasi pernyataan masalah yang sudah dijabarkan di atas dengan menerapkan teknologi berupa IoT, Infrared, dan LoRa yang diharapkan dapat menjadi solusi yang dapat meningkatkan efektivitas dan kemudahan pengalaman baru bagi para pengguna kendaraan bermotor di lapangan.

1.2 PROPOSED SOLUTION

Dalam menjawab identifikasi permasalahan yang sudah dijabarkan pada bagian sebelumnya, solusi yang ditawarkan pada proyek ini akan menggunakan pendekatan teknologi yang memberikan kesan efektif dan inovatif dalam penggunaan di dalamnya, yakni penerapan internet of things atau IoT dengan perangkat ESP 32 sebagai pusat utama kendali sistem, Infrared sebagai solusi terpadu yang akan menjawab permasalahan keluar masuk lahan parkir dengan menembakan maupun menerima dari sisi perangkat transmitter maupun receiver, LoRa sebagai cadangan penjamin terhubungnya proses pencocokan ID unik dengan sistem back end yang juga sudah kami buat andaikata jika konektivitas WiFi sedang buruk maupun tidak ada yang juga bertujuan sebagai bentuk manajemen database yang dirancang secara sistematis dan bisa memastikan Interaksi yang lancar antara perangkat keras yang kami jual dengan para pengguna.

Terdapat juga fitur tambahan yang akan menambahkan kemudahan dari proyek ini seperti LED yang akan mengirimkan konfirmasi langsung sebagai bentuk indikasi keberhasilan hingga servo yang akan menggerak naik turunkan palang penjaga sebagai bagian dari Prototype kami. Adapun ID unik yang sempat kami singgung sebelumnya juga akan dikirimkan dengan perantara bluetooth antara device pengguna ke transmitter serta hanya dimiliki setiap pelanggan kami secara unik yang ditujukan untuk memastikan tidak adanya kerancuan maupun kesalahan sistem pembayaran parkir yang ada di lapangan.

Diharapkan melalui langkah-langkah yang ada Ini para pengguna terdaftar yang sudah melengkapi persyaratan yang dibuat oleh sistem kami dapat melakukan proses registrasi maupun pembayaran parkir secara lebih terstruktur, mudah, dan cepat di lapangan. Selain daripada itu, kami juga berharap agar sekiranya proyek ini tidak hanya memberikan kesan positif bagi para penggunanya di lapangan melainkan juga turut berkontribusi bagi para pemangku kepentingan yang berhubungan untuk turut mendukung dan mensukseskan proyek ini.

1.3 ACCEPTANCE CRITERIA

Berikut ini adalah acceptance criteria dari proyek ini:

1. Proyek bisa melakukan registrasi pengguna dengan menampilkan id unik yang secara urut akan tersambung dari device pengguna ke backend akhir dengan perantara bluetooth, infrared, dan konektivitas wifi (LoRa sebagai cadangan untuk

mengatasi permasalahan tidak adanya konektivitas WiFi) untuk bisa melakukan pembayaran di akhir secara akurat dan mengatur palang keluar masuk lahan parkir.

2. Proyek bisa memvalidasi id unik saat pengguna hendak melakukan proses pembayaran dengan memakai transmitter yang ada serta infrared sebagai media perantaranya ke bagian receiver.
3. Aplikasi bisa memberikan respons yang tepat dalam menampilkan data registrasi dan pembayaran parkir bagi penggunanya. Hal ini mencakup saldo yang dimiliki para penggunanya.
4. Proyek dapat berjalan secara simultan tanpa mengalami gangguan yang berarti, seperti kegagalan verifikasi id, penghubungan bluetooth, pembacaan infrared, hingga kegagalan database maupun aplikasi yang ada di device para pengguna dalam melaksanakan fitur yang tertera di nomor 3.

1.4 ROLES AND RESPONSIBILITIES

The roles and responsibilities assigned to the group members are as follows:

Roles	Responsibilities	Person
Pembuatan kode full stack	Bertanggung jawab terhadap <i>frontend</i> dan <i>backend</i> dari aplikasi	Bernanda Nautval
Rangkaian	Bertanggung jawab terhadap rangkaian seperti penyambungan kabel dan penyolderan.	Althaf Nafi, Bernanda Nautval, Michael Gunawan, Rayhan Akbar
Pembuatan laporan dan materi presentasi	Bertanggung jawab terhadap laporan	Michael Gunawan
Pembuatan kode Hardware	Bertanggung jawab terhadap kode implementasi ESP32, IR,	Althaf Nafi, Rayhan Akbar

	Bluetooth, LoRa, dan koneksi ke <i>backend</i> .	
--	---	--

Table 1. Roles and Responsibilities

1.5 TIMELINE AND MILESTONES

Month	November						Desember					
Date	Pekan 3			Pekan 4			Pekan 1			Pekan 2		
Design Hardware												
Pengembangan Software												
Penyambungan dan Testing												
Debugging dan Finalisasi administrasi serta produk												

a. Hardware Design completion:

21-24 November menjadi masa pengembangan ide proyek dan 25-26 November menjadi masa proses desain produk

b. Software Development:

28 November - 7 Desember menjadi masa perancangan interface, aplikasi, dan backend serta ESP32 maupun alat dan bahan lainnya yang akan digunakan.

c. Integration and Testing of Hardware and Software:

4-8 Desember menjadi masa penyambungan sinergi antar perangkat keras seperti ESP32, LoRa, Infrared, Led, dengan bluetooth, WiFi, dan aplikasi yang telah dibuat.

d. Final Product Assembly and Testing:

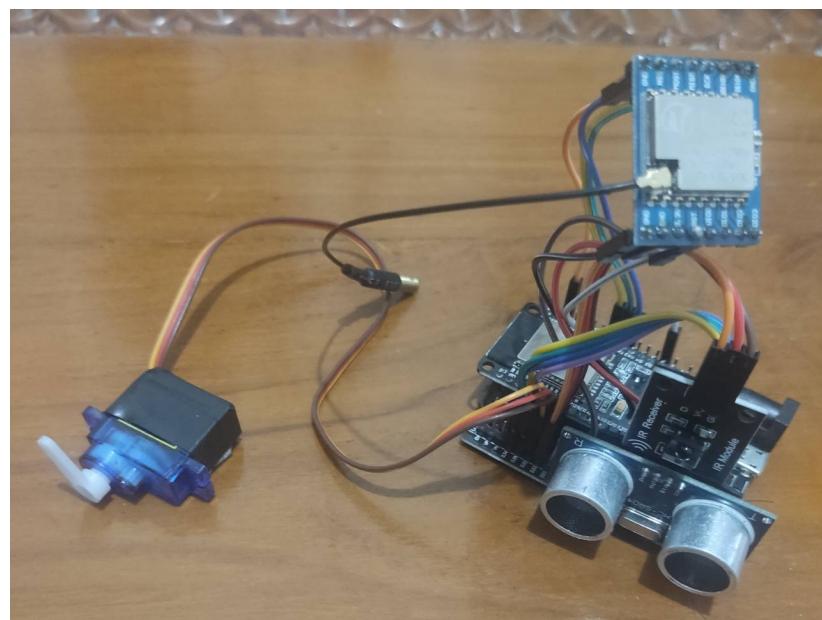
7 - 10 Desember menjadi masa pengujian produk yang telah dibuat untuk menguji ketepatan dan keberhasilannya sekaligus masa pembuatan laporan

CHAPTER 2

IMPLEMENTATION

2.1 HARDWARE DESIGN AND SCHEMATIC

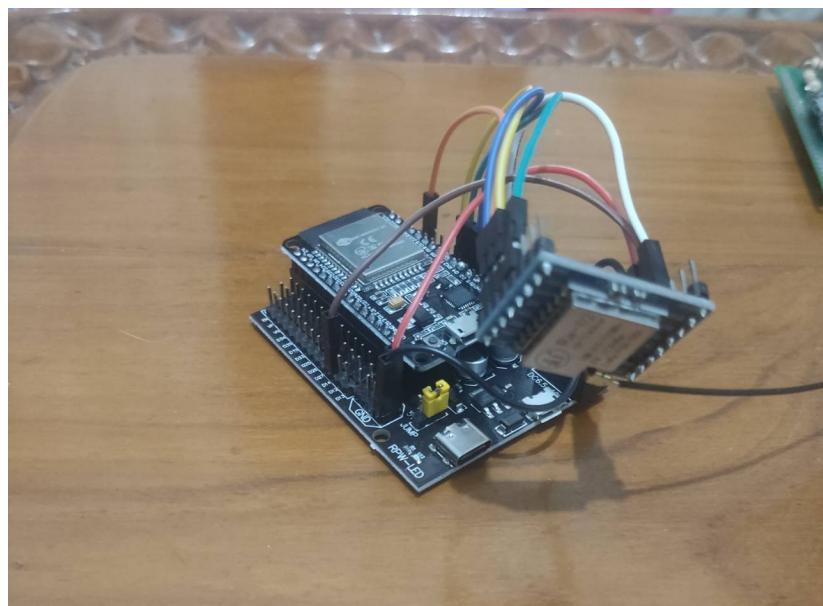
Pada bagian perangkat keras ini, kelompok kami memakai ESP32 karena fitur koneksi WiFi dan kapasitas pemroses data yang cukup baik sehingga dipercaya sebagai pengatur utama yang disinergikan dengan bahan pendukung lainnya, seperti Infrared yang ada pada sisi transmitter dan receiver untuk mengelola id yang dimiliki oleh si pengguna guna mencapai proses identifikasi yang cepat dan akurat, LED sebagai indikasi koneksi bluetooth, serta base station untuk yang telah ditempatkan di tempat terisolasi yang sudah sengaja dirancang untuk selalu mendapatkan internet sehingga bisa melakukan request ke sisi backend yang telah dibuat.



Gambar 1. Receiver



Gambar 2. Transmitter



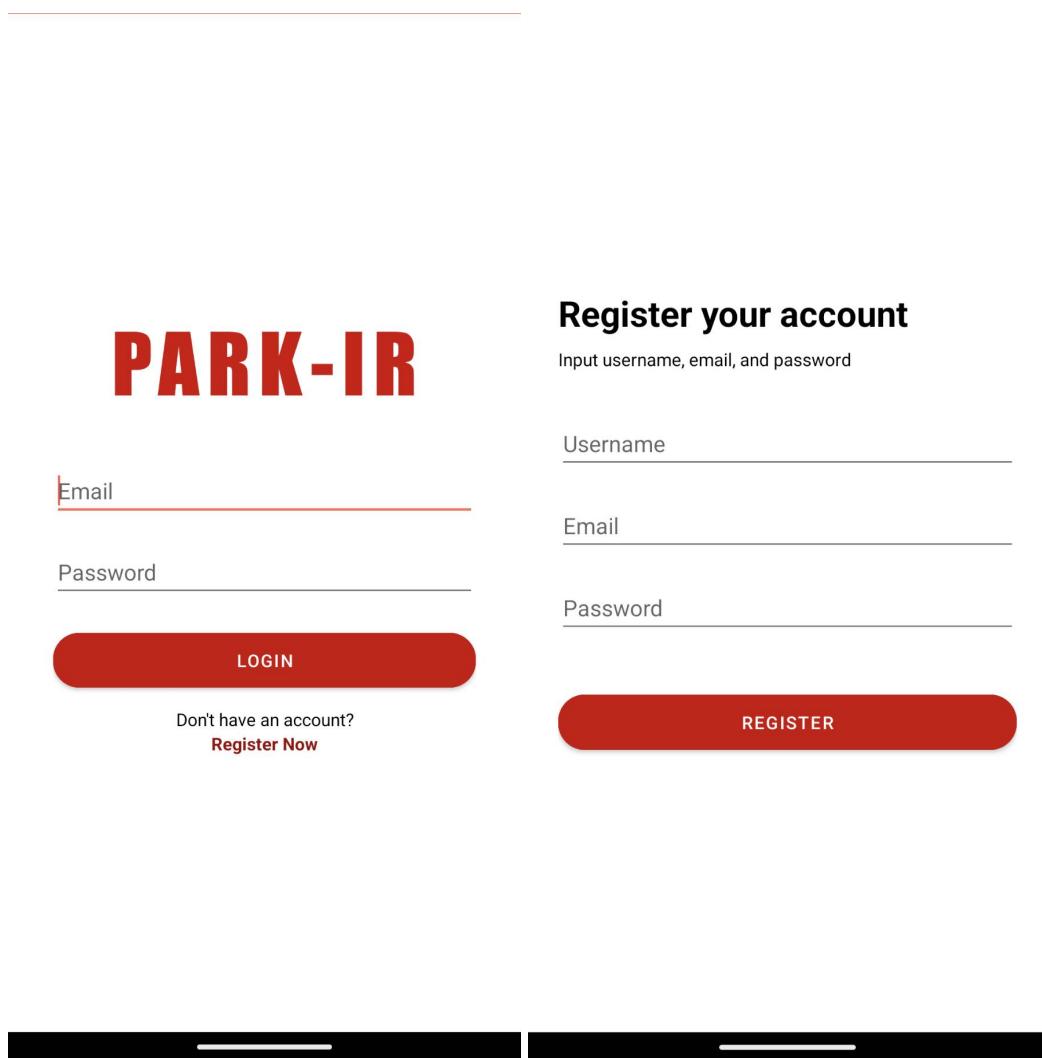
Gambar 3. Base Station

Dari ketiga gambar dapat ditinjau skematik sistem yang akan membangun koneksi atas transmitter, receiver, dan base station dengan komponen berupa ESP32, Resistor, Button, Led, Switch, LoRa, Sensor, Servo, dan komponen pendukung lainnya yang akan bahu-membahu untuk memastikan kelancaran dari jalannya koneksi dan pengaturan

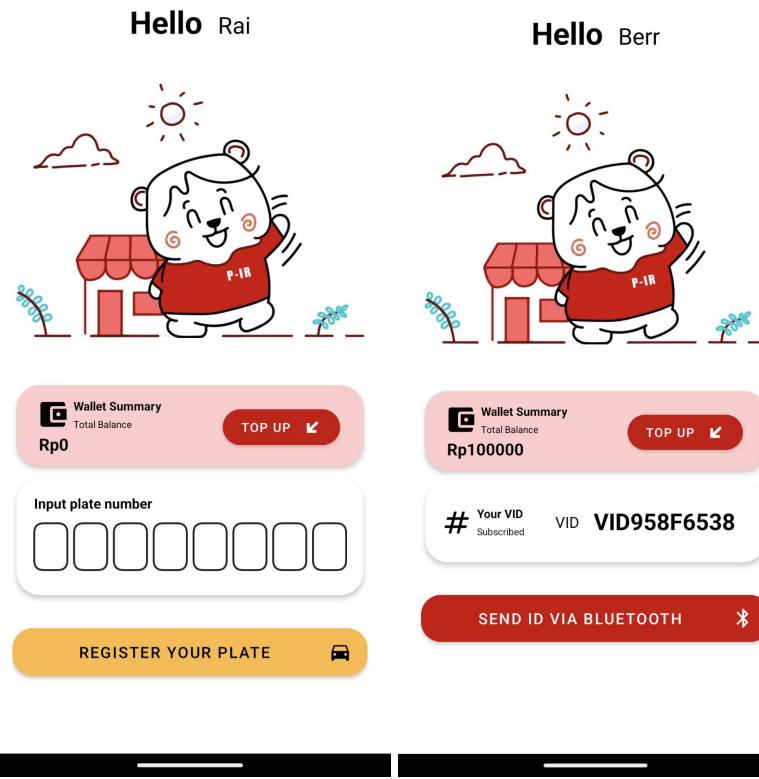
komponen sesuai yang diharapkan, hal ini termasuk pada peran koneksi fisik dan kabel jumper yang akan menggabungkan rangkaian yang ada dengan power supply yang dalam kasus ini adalah baterai untuk memastikan pasokan daya yang seperlunya.

2.2 SOFTWARE DEVELOPMENT

Pengembangan perangkat lunak pada sesi ini akan berfokus pada bagaimana implementasi dari berbagai fitur yang diperlukan, seperti interaksi dari berbagai perangkat keras yang digunakan hingga pengembangan perangkat lunak untuk web aplikasi yang juga dibuat. Berikut kami tampilkan beberapa tangkapan layar atas tampilan aplikasi android yang telah kami buat untuk digunakan dari sisi user:



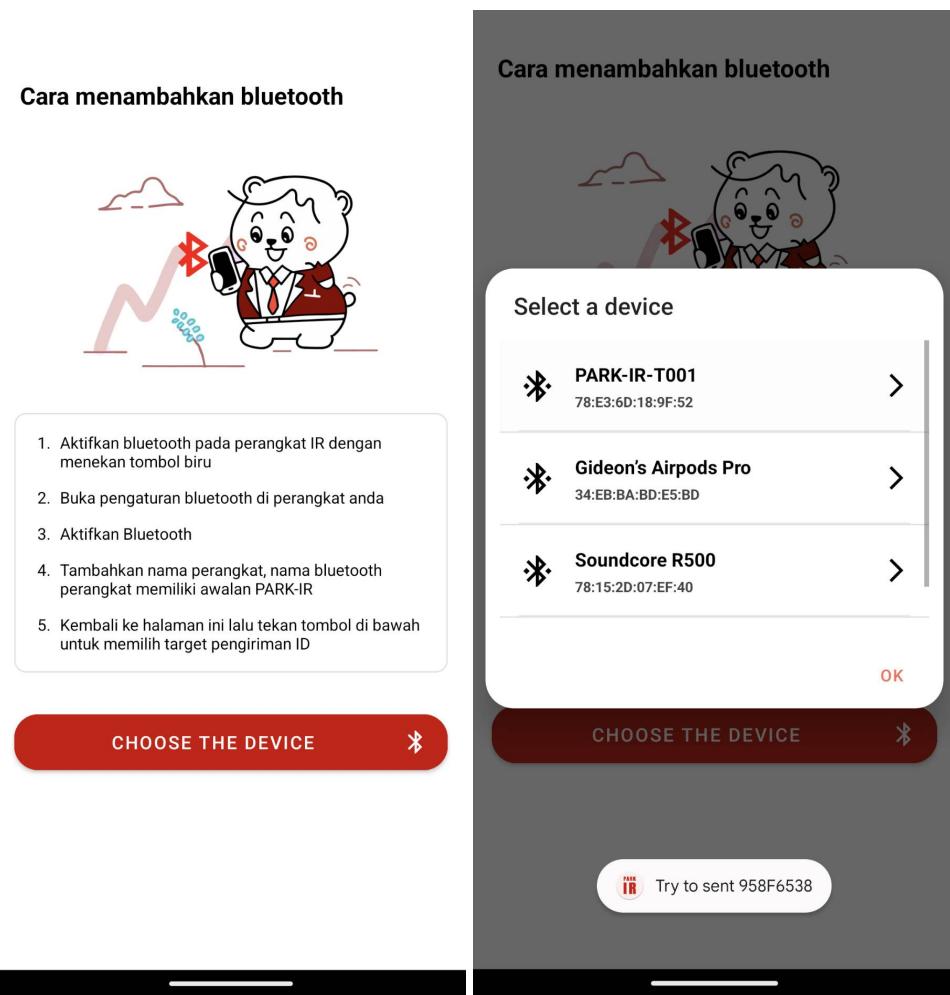
Gambar 4. Halaman untuk login dan registrasi pengguna



Gambar 5. Halaman utama, untuk registrasi plat nomor kendaraan pengguna



Gambar 6. Halaman untuk isi ulang saldo dari pengguna



Gambar 7. Halaman panduan untuk menyambungkan bluetooth dan proses penyambungan bluetooth

Pada bagian selanjutnya kami akan membahas perkembangan dari kode IoT yang telah kami buat. Pertama-tama pada program untuk perangkat transmitter, kami membuat kode yang nantinya memungkinkan perangkat untuk dapat menjalankan koneksi bluetooth dan terhubung ke smartphone, membaca *push button* sebagai tombol untuk mematikan dan menyalakan bluetooth, dan menjalankan komunikasi ke perangkat receiver menggunakan infrared. Secara detail, cara kerja dari program yang telah dibuat adalah sebagai berikut:

Selanjutnya, dibuat empat fungsi yang berbeda, yaitu fungsi callback, vButtonRead, vBTinit, dan vBTtask. Keempat fungsi ini nantinya akan dijalankan sebagai task pada void setup. Fungsi callback adalah fungsi yang akan dipanggil ketika bluetooth berhasil terkoneksi. Tujuan dari fungsi callback hanyalah untuk mengubah boolean BT.isConnected dari false menjadi true apabila terkoneksi, dan sebaliknya. fungsi callback ini akan di-attach ke SerialBT dengan method register_callback().

Kemudian, task vButtonRead adalah task yang akan melakukan pembacaan tombol. Tombol akan bersifat *toggling* sehingga ketika tombol ditekan sekali maka bluetooth akan diaktifkan, dan ketika tombol ditekan sekali lagi maka bluetooth akan dimatikan. Selain itu terdapat LED yang akan menyala ketika bluetooth aktif, dan mati ketika bluetooth dimatikan. Kemudian, task vBTinit adalah task yang akan memulai koneksi bluetooth, apabila bluetooth dinyalakan (menggunakan button). Koneksi blueooth ini diaktifkan dengan method begin(), dan diakhiri dengan method end().

Terakhir, method vBTtask merupakan task yang bertugas untuk melakukan pembacaan bluetooth serial ketika user mengirimkan id melalui smartphone ke perangkat ESP. setelah data diterima dari bluetooth serial, selanjutnya data tersebut akan disimpan secara permanen menggunakan library preferences dengan method-method seperti begin(), putUInt(), dan end().

Kemudian pada void setup, pin button akan diatur sebagai input_pullup, pin LED akan diatur sebagai output dan fungsi callback akan di-register ke serialBT. Kemudian, pengiriman data menggunakan infrared akan dimulai pada pin yang telah ditentukan, lalu program akan mengambil data yang telah tersimpan secara permanen menggunakan preferences.getUInt() apabila datanya tersedia, dan ketiga task yang sudah dibuat fungsinya sebelumnya akan dijalankan.

Terakhir pada void loop, pengiriman data menggunakan infrared akan dilakukan setiap satu detik, apabila perangkat sudah memiliki ID yang diberikan oleh user.

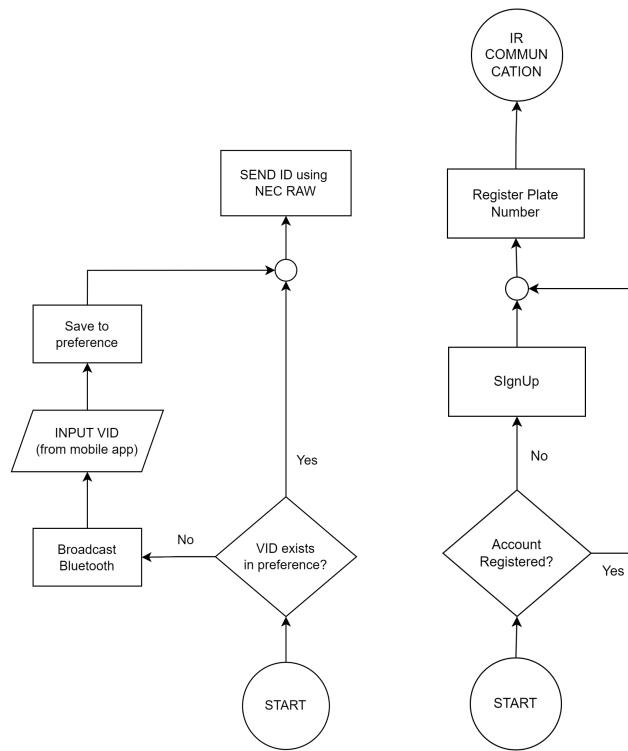
Pada perangkat receiver yang dilakukan adalah untuk menerima data dari IR Blaster dari perangkat transmitter yang merupakan ID dari user. Kemudian data ID tersebut beserta dengan lokasi dari receiver, berdasarkan dengan lokasi pemasangan receiver, akan digunakan untuk melakukan request HTTP ke backend. Dimana response dari hasil validasi request tersebut akan membuat gerbang dapat terbuka ataupun tertutup.

Request yang dilakukan ke backend merupakan request HTTP, namun saat kondisi dimana receiver tidak mendapatkan koneksi Internet maka receiver dapat melakukan koneksi ke base station lewat LoRa, dimana base station akan *merelay* data yang digunakan untuk membuat *request* dan meneruskan response HTTP tersebut balik ke receiver. Komunikasi LoRa juga dilakukan secara terenkripsi menggunakan AES.

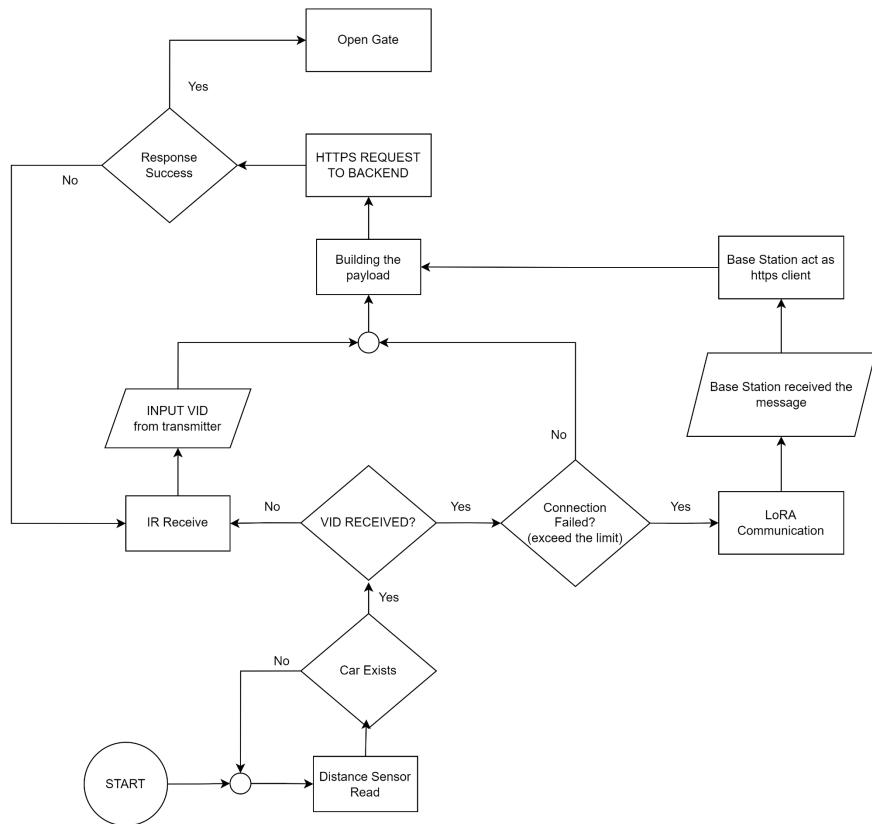
Untuk melakukan tugasnya, receiver memiliki empat task utama yaitu task HTTP, task IR, task onSuccess, dan task sensor. Task sensor berguna untuk melakukan pembacaan jarak menggunakan distance sensor dimana data *distance* tersebut akan selalu diupdate untuk stream data jarak yang selalu dapat diakses oleh task-task lain. Kemudian, task IR akan melakukan pengecekan terlebih dahulu terhadap data jarak dari *distance sensor*, jika terdeteksi sebuah objek yang berada di bawah threshold (5 meter pada kasus ini), maka IR receiver akan melakukan pembacaan data *id* yang seharusnya dikirimkan oleh transmitter. Jika data berhasil diterima dengan benar, maka task IR akan memberikan *receiverSemaphore* yang dibutuhkan oleh task HTTP untuk melakukan prosesnya. Dengan data *id* yang baru diterima task HTTP akan melakukan request HTTP ke backend dengan *id* tadi dan lokasi dari receiver tersebut. Jika request telah dilakukan dan response code yang diberikan oleh backend adalah 200, maka request dianggap berhasil dan data *id* berhasil divalidasi oleh backend. Sehingga, task HTTP memberikan semaphore ke task terakhir yaitu task onSuccess yang berfungsi untuk membukaan *gate* sehingga kendaraan dapat lewat.

Kemudian pada program untuk perangkat base station, kami membuat kode yang mengimplementasikan LoRa untuk melakukan request ke backend menggunakan HTTP. Base station disini akan digunakan sebagai *fallback* ketika perangkat receiver tidak bisa terkoneksi dengan internet dan berubah modenya menggunakan LoRa sebagai penengah. Komunikasi LoRa base station dengan receiver menggunakan enkripsi AES yang akan mengamankan pesan yang dikirimkan menggunakan sebuah key yang sama pada kedua perangkat. AES merupakan enkripsi yang simetris sehingga, key yang sama akan digunakan untuk mengekripsi data yang dikirimkan dan mendekripsi data yang diterima pada kedua perangkat (*receiver* dan *base station*).

Karena itu, pada *base station* terdapat dua task utama yaitu taskHTTP dan receivePayload_task. ReceivePayload_task berfungsi untuk selalu melakukan *listening* terhadap data yang dikirimkan lewat LoRa, dan karena kedua task ini berjalan pada core yang berbeda, keduanya bisa berjalan secara paralel. Data yang diterima pada task ini melalui LoRa adalah *id* dan *location* yang menandakan user dan lokasi dari *receiver* yang melakukan request. Kemudian, pada taskHTTP akan diterima data tersebut dan melakukan POST request untuk memvalidasi user. Data *id* dan *location* tadi akan dipassing oleh task receiver ke task HTTP menggunakan queue, yang dimana jika terdapat data baru, task HTTP akan langsung melakukan *request* ke backend. Terakhir, hasil dari request tersebut akan diteruskan kembali ke receiver melalui LoRa secara terenkripsi.



Gambar 8. Flowchart user dan pengiriman ID

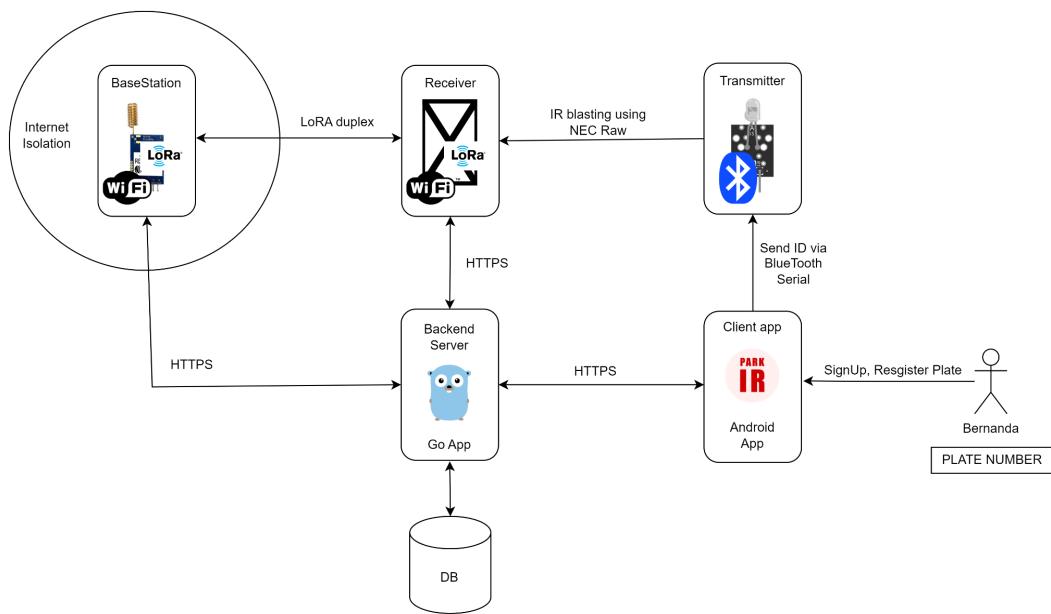


Gambar 9. Flowchart pembukaan palang dan user

2.3 HARDWARE AND SOFTWARE INTEGRATION

Dalam integrasi perangkat keras dan lunak kami berhasil menghubungkan kode program yang telah dikembangkan ke bagian perangkat kerasnya (ESP32, Infrared, LoRa, dan komponen lainnya yang mendukung) sehingga dapat dipastikan bahwa setiap elemen telah bekerja secara efektif dan mampu menjawab tujuan proyek atas pernyataan masalah yang telah kami susun di awal.

Integrasi hardware dan software dapat dilihat melalui diagram arsitektur berikut



Gambar 10. Kerangka kerja keseluruhan

Kode perangkat lunak kami terdiri dari tiga bagian, yakni untuk mengatur masing-masing dari sisi : IoT, Aplikasi Android, dan sisi Back-Endnya yang dirancang untuk berinteraksi langsung dengan para komponen perangkat keras yang telah disebutkan termasuk : *Led, switch, sensor, servo, button*, dan komponen pendukung lainnya. Hal ini bertujuan agar proses registrasi pengguna, pengisian saldo, pendapatan id pengguna yang akan dikirimkan ke sisi transmitter dengan bluetooth sebelum dipancarkan dengan infrared ke sisi receiver yang kemudian lantas akan dicocokan dengan sisi backend untuk mengatur penentuan palang parkir dengan servonya di dalam kasus proyek ini. Selain itu, program perangkat lunak ini juga akan mengendalikan proses pembacaan id hingga pengiriman kecocokan data ke sisi backend maupun interface pengguna didepannya.

Setelah kode perangkat lunak diimplementasikan dengan para perangkat keras ini, serangkaian proses *testing & debugging* juga telah kami laksanakan untuk memastikan bahwa interaksi di antara mereka berdua telah berjalan sesuai semestinya yang akan kami paparkan lebih lanjut pada bagian dibawah ini.

CHAPTER 3

TESTING AND EVALUATION

3.1 TESTING

Untuk memvalidasi fungsionalitas dan performa dari sistem yang telah kami buat, dilakukan beberapa uji coba dengan prosedur yang berbeda-beda. Fase uji coba ini bertujuan untuk memverifikasi kemampuan sistem dalam menjalankan koneksi bluetooth, menjalankan komunikasi dengan infrared, memvalidasi data yang diterima ke server backend, dan menjalankan LoRa apabila tidak terdapat WiFi pada perangkat receiver. Uji coba yang dilakukan adalah sebagai berikut:

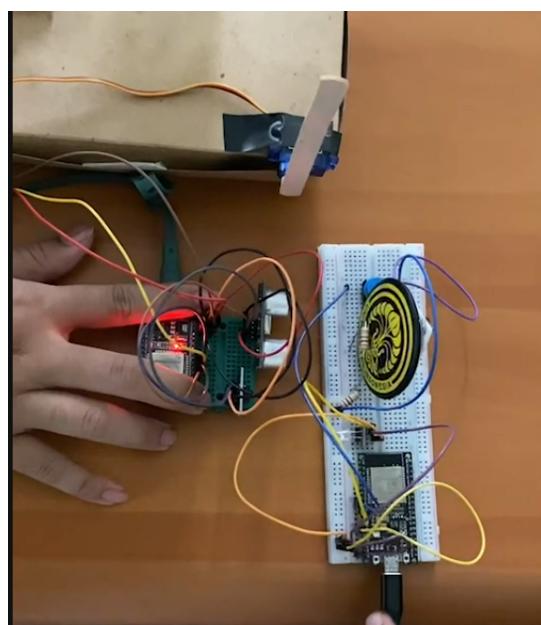
- Tes aplikasi: Uji coba ini memverifikasi kemampuan aplikasi yang telah dibuat dalam menampilkan data yang sesuai dan telah terhubung dengan database. Selain itu, dilakukan juga pengujian mengirim data melalui bluetooth ke perangkat transmitter.
- Tes koneksi bluetooth: Uji coba ini memverifikasi kemampuan sistem dalam menerima data dari smartphone melalui bluetooth serial. Pada uji coba ini dilakukan beberapa skenario yang berbeda seperti perbedaan jarak untuk memastikan konsistensi dan reliabilitas sistem dalam komunikasi bluetooth. Kemudian, terdapat uji coba tambahan yaitu tes button untuk menyalakan dan mematikan bluetooth.
- Tes komunikasi menggunakan infrared: Uji coba ini memverifikasi kemampuan sistem dalam mengirim dan menerima data menggunakan infrared. Dilakukan beberapa skenario yang berbeda seperti jarak dan penempatan yang berbeda antara IR blaster dan IR receiver untuk memastikan konsistensi data yang dikirimkan. Kemudian, terdapat uji coba tambahan yaitu mencoba sensor jarak yang akan mengatur perangkat receiver sehingga hanya memproses data dari infrared apabila perangkat transmitter berada pada jarak tertentu.
- Tes koneksi HTTP ke backend: Uji coba ini memverifikasi kemampuan sistem dalam mengirim data ke server menggunakan protokol HTTP, dan mendapatkan HTTP response code-nya. Uji coba ini juga melibatkan

pengujian penggantian protokol dari HTTP menjadi HTTPS untuk menjaga keamanan data pengguna.

- Tes protokol LoRa: Uji coba ini memverifikasi kemampuan sistem dalam menggunakan protokol komunikasi lora jika seandainya perangkat receiver tidak mendapatkan wifi. Dilakukan beberapa skenario yang berbeda seperti perbedaan jarak dan penghalang untuk memastikan konsistensi data.

3.2 RESULT

Hasil dari fase uji coba telah mengkonfirmasi keberhasilan implementasi dari sistem yang telah dibuat. Sistem telah menunjukkan koneksi bluetooth yang reliable, komunikasi menggunakan infrared dengan data yang konsisten, koneksi server yang reliable, serta dapat menggunakan protokol LoRa apabila dibutuhkan dengan data yang konsisten juga. aktuator dan indikator yang ada seperti servo dan LED juga sudah sesuai dengan keluaran dari program, seperti misalnya jika bluetooth dinyalakan, LED menyala, jika bluetooth mati, LED mati, dan misalnya jika server membalas dengan http response code 200, servo akan bergerak untuk membuka palang, dan jika selain itu maka servo akan tetap tertutup. Sistem yang dibuat telah menunjukkan kemampuannya dalam membuat pengalaman parkir yang *seamless* dan mengurangi waktu yang terbuang akibat harus scan tiket terlebih dahulu.



Gambar. Testing Result

Hasil dari fase uji coba ini mengindikasikan bahwa sistem yang telah dibuat sudah memenuhi *acceptance criteria* yang sudah ditentukan sebelumnya. Sistem juga sudah secara efektif menyelesaikan masalah parkir yang kurang efisien akibat perlunya scan tiket secara manual, dan menunjukkan potensinya untuk mengoptimasi pengoperasian lahan parkir.

3.3 EVALUATION

Tahapan evaluasi menjadi sektor penentu dari keberhasilan program ini dan penemuan celah yang dapat dieksplorasi lebih lanjut untuk perbaikan dan pengembangan kedepannya atas parameter yang didasarkan pada tujuan utama proyek dengan beberapa kriteria penilaian atas keberhasilan proyek yang telah kami tuliskan di seksi sebelumnya.

Dengan besar hati kami menyatakan bahwa sukses adalah bagian akhir dari kriteria yang telah kami tetapkan sejauh ini dan pada tahapan evaluasi ini, kami menilai bahwa kesulitan dan tantangan yang kami rasakan juga telah berhasil terselesaikan dengan baik, semisal : pada saat upaya integrasi antara perangkat keras dan perangkat lunak yang kadang terjadi kegagalan karena disebabkan oleh kerusakan komponen, penyambungan kabel yang salah, proses solder yang kurang sempurna, penetapan pin yang salah dari sisi kode, dan lain sebagainya. Kami menggarisbawahi pentingnya penggerjaan yang bertahap juga selama proses pembuatan proyek ini karena kadang terjadi kesalahan / *error* yang muncul secara mendadak alias tidak terjadi sebelumnya. Perkembangan lebih lanjut seperti : pencatatan durasi waktu parkir, informasi lokasi tempat lahan parkir yang sedang dipakai, hingga perhitungan biaya pada sisi aplikasi untuk sisi pengguna hingga pengamanan data dengan melakukan enkripsi data dapat membuat proyek ini menjadi lebih sempurna lagi meskipun tak jarang kami dapati beberapa lahan parkir yang telah memiliki mesin otomatis pencatatan waktu dan menampilkan biaya parkir dengan lcd setelah pengguna menempelkan kartu member mereka hingga kartu pembayaran elektronik, seperti : *e-money*, *flazz*, *tapcash*, dan lain sebagainya. Dapat ditinjau pada bagian ini sisi manakah dari proyek yang memerlukan penambahan hingga memberikan gambaran dari kendala yang dihadapi untuk referensi perkembangan lebih lanjut kedepannya guna menghasilkan proyek yang dapat terus berevolusi dengan kebutuhan konsumen maupun posisi di lapangan yang tentu akan mengedepankan keefektivitasan.

CHAPTER 4

CONCLUSION

Proyek kami dengan nama PARK-IR: INFRARED-BASED SOLUTION FOR SEAMLESS PARKING ini telah berhasil membantu meringankan dinamika permasalahan proses pembayaran biaya sewa lahan parkir yang ada saat ini, dengan penggunaan ESP32, Infrared, LoRa, dan komponen pendukung lainnya yang telah kami susun dengan ciamik dan terintegrasi ini terbukti berhasil meningkatkan efisiensi dan kecepatan di lapangan pada masa pembayaran sewa lahan parkir yang acap kali mengalami masalah situasi *bottleneck*.

Implementasi software dan hardware yang saling terhubung dan mendukung juga menciptakan pengurangan waktu dengan meminimalisasi keperluan kontak fisik maupun masalah ketidaktersediaan kartu yang diperlukan saat hendak melakukan proses pembayaran. Selain itu, sistem ini juga menjamin ketepatan pembayaran lewat id unik yang akan dikirimkan melalui bluetooth ke transmitter serta jaminan backup yang memadai untuk mencegah kemungkinan loss karena batasan konektivitas WiFi. Terlebih lagi dengan adanya interface aplikasi maupun backend yang secara berurutan akan memudahkan pengalaman pengguna serta mengatur pencocokan data pengguna didalamnya turut menyempurnakan proyek ini untuk mencapai tahap matang dalam pemasaran pasar.

Kami juga mendapati adanya tantangan yang bisa dijadikan bahan pembelajaran maupun evaluasi untuk perkembangan proyek ini kedepannya yang terjadi pada tahap integrasi antara perangkat ESP32 dan perangkat lunaknya. Diperlukannya waktu yang cukup lama untuk bisa benar-benar membangun fondasi yang kuat dari sistem bagian perangkat lunak ini maupun penyesuaian yang bertujuan untuk memastikan agar program dapat berjalan sesuai dengan tujuan di awal dan terbukti berhasil ditangani saat proses debugging.

REFERENCES

- [1] “ESP32 with LoRa using Arduino IDE | Random Nerd Tutorials,” randomnerdtutorials.com, Jun. 23, 2018. [Online]. Available : <https://randomnerdtutorials.com/esp32-lora-rfm95-transceiver-arduino-ide/>. [accessed Dec. 1, 2023].
- [2] “ESP32 Based IoT Projects,” iotdesignpro.com. <https://iotdesignpro.com/esp32-projects>. [accessed Dec. 1, 2023].
- [3] A. M. Lubis, “IoT-Based Smart Parking Management System Using ESP32 Microcontroller,” repository.president.ui.ac.id, Oct. 7, 2022. [Online]. Available : http://repository.president.ac.id/bitstream/handle/123456789/11127/1.a.%20IoT-Based_Smart_Parking_Management_System_Using_ESP32_Microcontroller%20%28Published%29.pdf?sequence=1&isAllowed=y. [accessed Dec. 2, 2023].
- [4] L. A. Tiron, “ONLINE PARKING SYSTEM USING ESP32,” bmi.unikl.edu.my. [Online]. Available : https://bmi.unikl.edu.my/wp-content/uploads/2022/11/136_142_ONLINE-PARKING-SYSTEM-USING-ESP32.pdf. [accessed Dec. 3, 2023].
- [5] M. Chandran et al., “An IoT Based Smart Parking System,” Journal of Physics: Conference Series, 2019, vol. 1339, no. 1, doi: 10.1088/1742-6596/1339/1/012044. Available : https://www.researchgate.net/publication/341029639_An_IoT_Based_Smart_Parking_System. [accessed Dec. 4, 2023].
- [6] A. Mackey, P. Spachos, and K. N. Plataniotis, “Smart Parking System Based on Bluetooth Low Energy Beacons with Particle Filtering,” IEEE Syst. J., vol. 14, no. 3, pp. 3371–3382, Sep. 2020, doi: 10.1109/JYST.2020.2968883. Available : https://www.researchgate.net/publication/338737767_Smart_Parking_System_Based_on_Bluetooth_Low_Energy_Beacons_with_Particle_Filtering. [accessed Dec. 5, 2023].
- [7] Satria, Mahesa & Firman Ashari, Ilham & Idris, Mohamad, “Parking System Optimization Based on IoT using Face and Vehicle Plat Recognition via Amazon Web Service and ESP-32 CAM (Case Study: Institut Teknologi Sumatera)”. 11. 137-153. 10.18495/COMENGAPP.V11I2.409. Available : https://www.researchgate.net/publication/362910479_Parking_System_Optimization

[Based on IoT using Face and Vehicle Plat Recognition via Amazon Web Service and ESP-32 CAM Case Study Institut Teknologi Sumatera](#). [accessed Dec. 6, 2023].

- [8] Simatupang, Joni & Lubis, Aida & Vincent, Vincent. “IoT-Based Smart Parking Management System Using ESP32 Microcontroller”. 305-310. 10.23919/EECSI56542.2022.9946608. Available : https://www.researchgate.net/publication/365460017_IoT-Based_Smart_Parking_Management_System_Using_ESP32_Microcontroller. [accessed Dec. 7, 2023].

APPENDICES

Appendix A: Project Schematic



Appendix B: Documentation

