

Rancang Bangun *Prototype Smart Parking* Berbasis *Internet of Things (Iot)*

Arni Litha¹⁾, Sahbuddin Abdul Kadir²⁾, Divya Andini A.M³⁾, Wikhe Apriani Paulus⁴⁾,

^{1,2,3,4}, Jurusan Elektro, Politeknik Negeri Ujung Pandang

¹arnilitha@poliupg.ac.id,

²sahbuddin.ak@poliupg.ac.id

³divyaandini@gmail.com

⁴wikhe04april@gmail.com

Abstrak

Perancangan *prototype smart parking* bertujuan untuk merancang alat monitoring kondisi lokasi parkir roda empat baik dari segi ketersediaan lokasi parkir berbasis *Internet of Things (IoT)* maupun keteraturan pengguna dalam memarkirkan kendaraannya. Deteksi tempat parkir yang kosong menggunakan sensor infra merah. Ketika terdapat mobil yang menghalangi pancaran cahaya infra merah cahaya tersebut akan dipantulkan dan diterima oleh *receiver* sensor infra merah, kemudian sinyal output sensor infra merah diteruskan ke server IoT dengan aplikasi blynk menggunakan perangkat wifi yang terintegrasi pada mikrokontroler NodeMCU ESP8266. Aplikasi ini kemudian akan menampilkan lokasi parkir yang kosong atau yang sedang terisi. Setiap slot tempat parkir pada prototipe menggunakan pembatas yang dilengkapi dengan sensor kontak. Sensor ini membangkitkan alarm saat pengendara memarkirkan mobilnya melewati pembatas parkir. Hasil perancangan ini menunjukkan Pada Aplikasi *Blynk* apabila suatu slot parkir mendeteksi keberadaan mobil ditampilkan indikator LED akan berwarna hijau dan jika tidak ada mobil maka berwarna hitam. Modul WiFi ESP8266 dapat terhubung dengan Aplikasi *Blynk* dan mengirimkan data dengan delay 5.21 - 6.66 detik. Kemudian alarm akan berbunyi ketika kendaraan melewati pembatas parkir.

Keywords: *Internet of Things, Sensor Infra merah, Blynk, NodeMCU ESP8266*

I. PENDAHULUAN

Sistem parkir di tempat umum saat ini kurang efisien, apabila seseorang ingin mencari tempat parkir saat keadaan ramai memerlukan waktu yang cukup lama yang menyebabkan terbuangnya waktu, tenaga, dan bahan bakar, selain itu masalah yang sering ditemui adalah cara parkir yang melewati batas slot parkir. Oleh karena itu, diperlukan suatu aplikasi *Monitoring System* yang akan memudahkan pengemudi untuk menemukan tempat untuk memarkirkan mobil. Selain itu, juga diperlukan sebuah sistem peringatan dalam bentuk alarm jika kendaraan diparkir melewati batas slot parkir. Penggunaan teknologi *wireless* pada saat ini telah memberikan banyak perubahan yang sangat besar dalam kehidupan sehari-hari, salah satunya adalah teknologi *Wireless Sensor Network*. *Wireless Sensor Network* merupakan pengembangan dari teknologi *wireless* yang dapat memantau kondisi suatu lingkungan menggunakan sensor dan mengirimkan data tersebut ke node yang saling terhubung. Infrastruktur jaringan *Wireless Sensor Network* ini terdiri dari sekumpulan node yang tersebar di suatu daerah tertentu yang saling terhubung secara nirkabel.

Pada penelitian ini didesain dan implementasikan suatu *prototype smart parking* yang terhubung dengan aplikasi *Internet of Thing* untuk memantau keadaan lokasi parkir. Prototipe ini menggunakan teknologi *Wireless Sensor Network* dengan topologi *Star* untuk mendeteksi keadaan setiap slot tempat parkir. Selain itu, prototipe ini

juga dilengkapi dengan sistem alarm. Sedangkan kondisi tempat parkir dapat diakses melalui aplikasi *Blynk*.

Pada penelitian sebelumnya, telah terdapat beberapa jurnal yang merancang *smart parking* salah satunya dari mahasiswa Universitas Diponegoro. *Monitoring Smart parking* dirancang menggunakan aplikasi *Android Studio*.

II. KAJIAN LITERATUR

A. *Internet of Thing*

Internet of Things, atau dikenal juga dengan singkatan IoT, merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus. Pada dasarnya, IoT mengacu pada benda yang dapat diidentifikasi secara unik sebagai representasi virtual dalam struktur berbasis Internet. Istilah IoT awalnya disarankan oleh Kevin Ashton pada tahun 1999 dan mulai terkenal melalui *Auto-ID Center di MIT*.

IoT dapat dikatakan sebagai sebuah konsep dari benda-benda di sekitar yang mampu berkomunikasi dan berbagi data antara satu sama lain melalui sebuah jaringan seperti internet. Adapun kemampuan seperti berbagi data, *remote control*, dan sebagainya, termasuk juga pada benda di dunia nyata. Contohnya bahan pangan, elektronik, koleksi, peralatan apa saja, termasuk benda hidup yang semuanya tersambung ke jaringan lokal dan global melalui sensor yang tertanam dan selalu aktif kemudian data tersebut dikumpulkan lalu dikirim ke sebuah *database*

atau *server*, sehingga dengan adanya teknologi ini pekerjaan yang dilakukan oleh manusia menjadi lebih cepat, muda dan efisien.

Sebuah perangkat IoT memiliki sebuah radio yang dapat mengirim dan menerima koneksi *wireless*. Protokol *wireless* IoT didesain untuk memenuhi beberapa servis dasar yaitu beroperasi dengan daya dan bandwidth yang rendah, dan bekerja dalam jaringan mesh. Beberapa perangkat bekerja pada frekuensi bidang 2.4 GHz yang juga digunakan oleh Wi-fi dan *Bluetooth*, dan cakupan sub- GHz. Frekuensi sub-GHz tersebut termasuk 868 dan 915 MHz, memiliki keuntungan dalam rendahnya interferensi. Perangkat-perangkat IoT terhubung dalam sebuah jaringan mesh satu sama lain dan mengirimkan sinyal seperti pelari dalam lari estafet. jaringan ini berbalikan dengan jaringan tersentralisasi. Cakupan transmisi dari perangkat IoT dalam jaringan *mesh* ialah \pm 9 meter hingga lebih dari 90 meter (Muhammad Mufid Luthfi, 2020). Karena perangkat dalam jaringan *mesh* mampu untuk “mentransfer” sinyal, tentu dan dapat terhubung dengan ribuan sensor dalam suatu area yang luas, seperti sebuah kota, dan beroperasi dengan selaras. Jaringan mesh memiliki kemampuan tambahan untuk bekerja di sekitar area perangkat yang gagal (tidak terkoneksi).

Protokol jaringan mesh IoT antara lain Z-Wave Alliance, Digbee Alliance, dan Insteon, yang juga bekerja sama dengan vendor. Protokol- protokol tersebut tidak memiliki interoperabilitas, yang berarti protocol tersebut tidak mampu untuk bekerja sama antar beberapa macam sistem, meskipun dapat juga dihubungkan melalui hubs. Digbee merupakan protokol terbuka (*open protocol*), namun banyak kritik yang menyatakan tidak semua pengimplementasiannya harus sama. Digbee menyediakan sertifikasi untuk memastikan standar pengaplikasian. Insteon dan Z-Wave merupakan protokol berpaten, sehingga standarisasi implementasinya lebih terjamin. Untuk meningkatkan skalabilitas akses komunikasi IoT, setelah bekerja keras sejak tahun 2007, akhirnya memiliki 6LoWPAN sebagai standar integrasi IP pada jaringan IoT berdaya rendah.

B. Blynk

Blynk dirancang untuk IoT dengan tujuan dapat mengontrol *hardware* dari jarak jauh, dapat menampilkan data sensor, dapat menyimpan data, visual dan melakukan banyak hal canggih lainnya. Ada tiga komponen utama dalam *platform*:

1. *Blynk App*, memungkinkan untuk membuat antarmuka menakjubkan untuk proyek-proyek dengan menggunakan berbagai *widget* yang tersedia
2. *Blynk Server*, bertanggung jawab untuk semua komunikasi antara *smartphone* dan perangkat keras. Dapat digunakan *Blynk Cloud* atau menjalankan *server Blynk* pribadi secara lokal. *Blynk* bersifat *open-source*, bisa dengan mudah menangani ribuan perangkat dan bahkan dapat diluncurkan pada Raspberry Pi.
3. *Blynk Library*, dapat digunakan untuk semua *platform* perangkat keras yang populer serta memungkinkan

komunikasi dengan server dan memproses semua perintah *incoming* dan *outcoming*. [1]

C. Sensor Infra merah

Sensor inframerah adalah komponen elektronika yang dapat mengidentifikasi cahaya merah. Sistem sensor inframerah pada dasarnya menggunakan inframerah sebagai media untuk komunikasi data antara *receiver* dan *transmitter*. Cara kerja dari sensor infrared FC-51 ini adalah dengan memancarkan sinar infra merah melalui dioda pemancar infra merah. Jika tidak ada benda yang ada di wilayah pancaran infra merah, maka tidak ada media yang dapat memantulkan sinar infra merah tersebut. Penerima infra merah tidak akan mendeteksi apapun. Pada keadaan ini, *LED* indikator sinyal akan mati (*OFF*) dan sinyal keluaran akan berlogika *HIGH* (5 V). Jika ada benda yang ada di wilayah pancaran infra merah dioda tersebut, maka sinar infra merah tersebut akan dipantulkan kembali. Pantulan sinar infra merah ini akan dideteksi oleh dioda photo dan akan diproses oleh IC LM393. Pada keadaan seperti ini, *LED* indikator sinyal akan hidup (*ON*) dan sinyal keluaran akan berlogika *LOW* (0V). [2]

D. NodeMCU

Istilah NodeMCU secara default mengacu pada *firmware* daripada kit pengembangan. *firmware* menggunakan bahasa Lua *scripting*. Ini didasarkan pada proyek eLua, dan dibangun diatas *Espressif Non-OS SDK* untuk *ESP8266*. Menggunakan banyak proyek *open source*, seperti *lua-cjson* dan *SPIFFS*. Modul ini juga dilengkapi dengan prosesor, memori dan GPIO dimana jumlah pin bergantung dengan jenis *ESP8266* yang kita gunakan. Sehingga modul ini bisa berdiri sendiri tanpa menggunakan mikrokontroler apapun karena sudah memiliki perlengkapan layaknya mikrokontroler. [3]

E. Arduino Uno

Arduino Uno adalah salah satu development kit mikrokontroler yang berbasis pada ATmega28. Uno memiliki 14 pin digital input / output (dimana 6 dapat digunakan sebagai output PWM), 6 input analog, resonator keramik 16 MHz, koneksi USB, jack listrik, header ICSP, dan tombol reset. Uno dibangun berdasarkan apa yang diperlukan untuk mendukung mikrokontroler, sumber daya bisa menggunakan power USB (jika terhubung ke komputer dengan kabel USB) dan juga dengan adaptor atau baterai. [4]

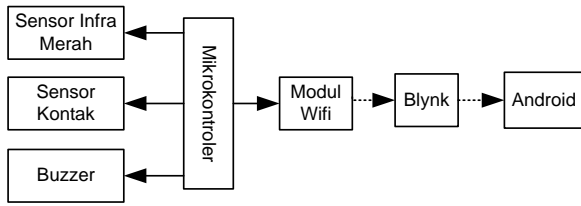
F. Buzzer

Buzzer adalah sebuah komponen elektronika yang dapat mengubah sinyal listrik menjadi getaran suara. Buzzer ini biasa dipakai pada sistem alarm. Juga bisa digunakan sebagai indikasi suara. Cara Kerja Buzzer pada saat aliran listrik atau tegangan listrik yang mengalir ke rangkaian yang menggunakan piezoelectric tersebut. Piezo buzzer dapat bekerja dengan baik dalam menghasilkan frekwensi di kisaran 1 - 6 kHz hingga 100 kHz. [5]

III. METODE PERANCANGAN

A. Perancangan

Perancangan sistem *prototype smart parking* pada penelitian ini dapat dilihat pada diagram blok gambar 2.



Gambar 2. Diagram Blok *Prototype Smart Parking*

Perancangan perangkat keras (*Hardware*) menggunakan mikrokontroler Arduino Uno, NodeMCU, Modul Sensor Inframerah. Modul sensor infra merah berfungsi untuk mendeteksi ada atau tidaknya parkiran mobil yang kosong, yang mengirimkan sinyal detektor ke Arduino Uno. Arduino Uno berfungsi sebagai Pembaca data dari sensor infra merah melalui pin yang terhubung pada modul sensor inframerah. NodeMCU berfungsi sebagai penerima data yang dari Arduino Uno yang kemudian melalui pin Tx Rx diteruskan ke Aplikasi Blynk. Pada Aplikasi Blynk terhubung pada NodeMCU melalui Wi-Fi, sehingga perangkat keras ini dapat dimonitor dan dikontrol pada smartphone dari jarak jauh.

Pada Perancangan Mekanik di diagram blok terdapat Buzzer dan sensor kontak masing-masing sebanyak 9 buah terpasang pada masing-masing pembatas jalur parkiran yang berfungsi untuk mendeteksi apabila terdapat kendaraan yang salah memarkir atau mengambil jalur parkir orang lain.

B. Pengujian Alat

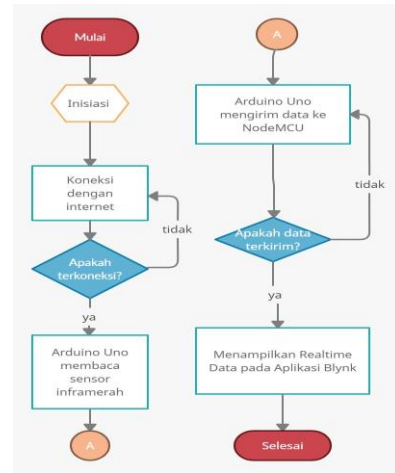
Pengujian dilakukan dengan membandingkan *realtime* pada tampilan Aplikasi Blynk dengan keadaan *real prototype smart parking*, mengukur jarak pembacaan pada sensor inframerah ketika potensiometer pada sensor inframerah diatur untuk memperoleh tegangan sama dan pengujian terhadap sensitivitas buzzer ketika terdapat kesalahan parkir yang dilakukan si pengendara.

C. Analisis data

Analisis data yang diperoleh dalam perancangan ini bertujuan untuk memaparkan langkah dalam pembuatan *Prototipe Smart Parking* Berbasis IoT, seperti dijelaskan pada diagram alir pada Gambar 1.

Sistem deteksi kendaraan pada tempat parkir diawali dengan inisiasi pada NodeMCU. setelah inisiasi melakukan koneksi dengan modem wifi, jika tidak terkoneksi maka dilakukan koneksi ulang. Ketika sudah terkoneksi maka dilanjutkan dengan membaca sensor inframerah jika cahaya terdeteksi menghasilkan data *LOW* atau bernilai 0v dan jika tidak terdeteksi menghasilkan data *HIGH* atau bernilai 5v. Pada Arduino UNO membaca hasil data dari masing – masing sensor inframerah, jika data sensor *HIGH* maka bernilai 0 dan jika data sensor *LOW* maka bernilai 1, setelah itu dilakukan pengiriman data sensor beserta posisinya ke NodeMCU melalui internet, jika data terkirim maka akan terbaca pada

aplikasi blynk, kemudian blynk akan menampilkan posisi parkir yang kosong dan yang telah terisi.



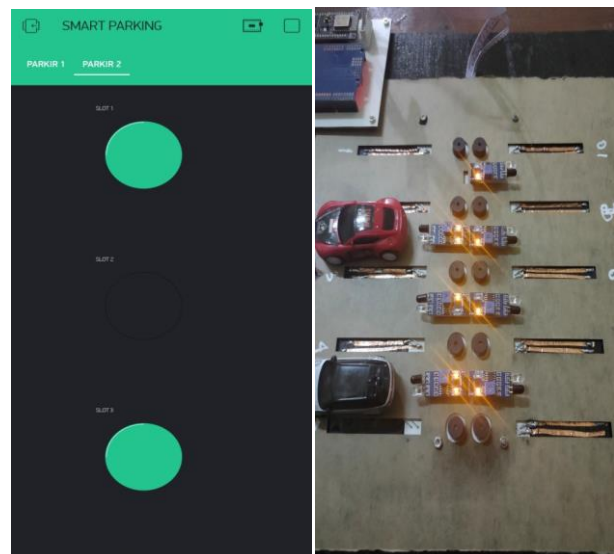
Gambar 1. Flowchart Perancangan

IV. HASIL DAN PEMBAHASAN

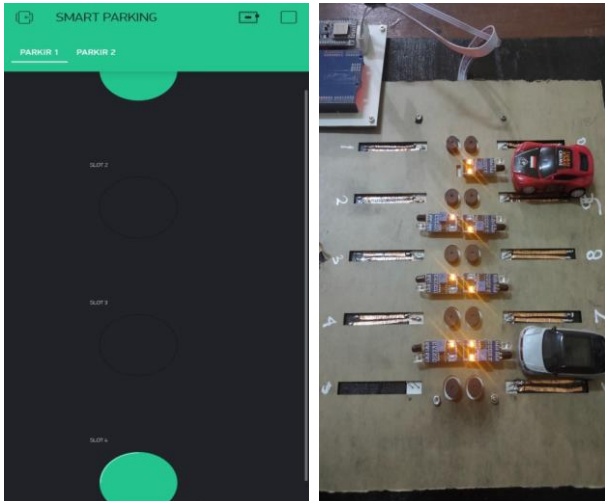
Setelah melalui tahap perancangan *Prototype Smart Parking* Berbasis IoT, maka pada bab ini akan dipaparkan hasil pengujian, pengukuran dan analisis.

A. Pengujian Prototype

Pada pengujian ini menggunakan dua buah mniatur mobil kemudian diparkirkan pada prototipe parkiran yang telah dirancang, lalu memperhatikan antara kesesuaian keadaan pada Aplikasi *Blynk* dengan keadaan *real* prototipe parkiran selama 30 menit. Selain itu dilakukan pengujian terhadap pendeteksi kesalahan parkir dengan memerhatikan kesesuaian antara kondisi real prototipe parkiran dengan buzzer yang akan berbunyi jika didapati kesalahan dalam memarkirkan kendaraan.



Gambar 3. Keadaan *Real Prototype Smart Parking* dan Aplikasi *Blynk*



Gambar 4. Keadaan Real Prototype Smart Parking dan Aplikasi Blynk Setelah 30 menit

Dari pengujian yang telah dilakukan dapat diperhatikan bahwa pada awal pengujian slot 1 dan slot 3 pada lorong 2 terisi, sementara itu pada 30 menit kemudian dilakukan pengujian kembali dan didapatkan hasil slot 1 dan slot 4 pada lorong 1 terisi dengan catatan *prototype smart parking* masih terhubung dengan Aplikasi *Blynk* hasil pengujian ini dikatakan berhasil ditandai dengan LED indikator sensor inframerah akan hidup (ON) dan LED indikator sinyal juga hidup (ON) sehingga sinyal keluaran akan berlogika LOW (0V) serta pada aplikasi *blynk* akan ditampilkan informasi slot yang terisi (fitur indikator LED berwarna hijau) dan slot yang masih kosong (fitur indikator LED tidak berwarna) sesuai dengan keadaan *real* pada perancangan *prototype smart parking*.

Pada pengujian perancangan mekanik didapatkan hasil yaitu buzzer akan berbunyi ketika ada pengendara yang salah memarkirkan kendaraannya dengan menginjak pembatas parkir di setiap slot.

B. Pengukuran Jarak

Pada pengujian ini, akan diuji *Prortype Smart Parking* dengan data yang ditampilkan pada Aplikasi *Blynk* yang diakses oleh *smartphone* yang berperan sebagai master dan *smartphone* yang berperan sebagai slave. Pengujian dilakukan dengan cara menghubungkan perancangan prototipe dengan koneksi internet dan *smartphone* yang digunakan kemudian membandingkan tampilan *real prototype smart parking* dengan tampilan Aplikasi *Blynk* di *Smartphone*. Untuk mengetahui apakah perancangan prototipe terkoneksi dengan aplikasi pada *smartphone* maka dilakukan uji jarak sebagai berikut:

Tabel 1. Pengujian Jarak Hubung antara Prototipe Parkiran dengan Aplikasi *Blynk*

Jarak (m)	Status
1	Terhubung
2	Terhubung
3	Terhubung
4	Terhubung
5	Terhubung

6	Tidak Terhubung
7	Tidak Terhubung

Berdasarkan tabel 1 dapat dilihat informasi mengenai jarak maksimal *smartphone* yang berperan sebagai *master* pada *Blynk* dapat berkomunikasi yaitu pada jarak 5 meter dengan catatan NodeMCU dan *smartphone* masih terhubung dengan internet dan wifi yang sama. Sementara itu, *smartphone* yang berperan sebagai *slave* atau *smartphone* pengendara dapat mengakses informasi tentang parkir yang tersedia dengan menginstal Aplikasi *Blynk* terlebih dahulu kemudian melakukan *scan barcode* yang telah disediakan oleh *smartphone master* sehingga pada *smartphone* yang berperan sebagai *slave* dapat menampilkan informasi jumlah parkir yang kosong dan terisi dimanapun selama masih terhubung dengan internet.

C. Pengukuran Jarak Deteksi Modul Sensor Inframerah

Pada pengukuran jarak deteksi modul sensor inframerah dilakukan dengan ;cara menyesuaikan tegangan potensiometer pada setiap sensor pada tegangan terkecil yaitu 0.19 Volt dan tegangan 0.58 Volt kemudian membandingkan output sensor dengan alat ukur penggaris.

Tabel 2. Pengukuran Jarak Deteksi Sensor Infamerah Tegangan 0.19 Volt

No	Slot Parkiran	Jarak Depan (cm)	Jarak Samping Kiri (cm)	Jarak Samping Kanan (cm)
1	1	2	0.2	0.2
2	2	2	0.2	0.2
3	3	2	0.2	0.2
4	4	2	0.2	0.2
5	5	2	0.2	0.2
6	6	2	0.2	0.2
7	7	2	0.2	0.2

Tabel 3. Pengukuran Jarak Deteksi Sensor Infamerah Tegangan 0.58 Volt

No	Slot Parkiran	Jarak Depan (cm)	Jarak Samping Kiri (cm)	Jarak Samping Kanan (cm)
1	1	3	0.35	0.35
2	2	3	0.35	0.35
3	3	3	0.35	0.35
4	4	3	0.35	0.35
5	5	3	0.35	0.35
6	6	3	0.35	0.35
7	7	3	0.35	0.35

Berdasarkan tabel 2 dan 3 terlihat bahwa jarak depan deteksi sensor inframerah di tegangan yang sama menghasilkan besar deteksi yang sama dan untuk jarak deteksi kiri dan kanan juga memiliki besar yang sama. Berdasarkan hasil pengukuran dapat diketahui bahwa ketika suatu benda yang menghalangi inframerah yang dipancarkan berada tepat didepan IR *Transmitter* maka cahaya inframerah tersebut akan terpantul dan diterima oleh IR *Receiver* namun ketika suatu benda tidak dapat dideteksi oleh IR *Receiver* ini dipengaruhi oleh besar tegangan potensial yang telah diatur, semakin besar tegangannya maka semakin besar jarak deteksi depan kiri

dan kanan yang menyebabkan penghalang tidak lagi berada di area cahaya inframerah yang dipancarkan.

D. Pengujian Modul ESP8266

Pengujian Modul ESP8266 dilakukan dengan cara mencatat delay pengiriman data modul sensor inframerah ke Aplikasi *Blynk* yang akan diakses oleh pengendara mobil menggunakan stopwatch. Pengujian ini dibagi menjadi dua operator seluler yang berbeda yaitu A dan B namun pada lokasi pengujian yang sama.

Tabel 4. Pengujian Waktu Pengiriman Data Sensor ke Aplikasi *Blynk* menggunakan Kartu A dan Kartu B.

No.	Slot	Delay (s)	
		Provider A	Provider B
1	1	5.71	8.32
2	2	3.01	8.94
3	3	4.91	8.84
4	4	6.91	3.20
5	1	6.79	4.17
6	2	6.05	9.13
7	3	5.21	4.03
Rata-rata Delay		5.51	6.66

Berdasarkan tabel 4 terlihat bahwa rata-rata delay pengiriman data ke Aplikasi *Blynk* pada penggunaan kartu A dan B adalah 5.51s dan 6.66s, namun data ini dapat berubah sewaktu-waktu karena kecepatan pengiriman bergantung pada kecepatan akses internet si pengguna kendaraan.

E. Pengujian Perancangan Mekanik

Pengujian perancangan mekanik dilakukan dengan cara mencatat kepekaan buzzer terhadap pengendara yang melanggar dalam memarkirkan mobilnya dan mengukur besar tegangan pada buzzer pada saat berbunyi dan tidak berbunyi.

Tabel 5. Pengujian Respons Buzzer

Pembatas Parkir Slot	Respons Buzzer	Tegangan (V)
1	Berbunyi	4.99
2	Berbunyi	4.99
3	Tidak berbunyi	0
4	Berbunyi	4.99
5	Tidak berbunyi	0
6	Berbunyi	4.9
7	Tidak berbunyi	0
8	Berbunyi	4.99
9	Berbunyi	4.99
10	Tidak berbunyi	0

Berdasarkan tabel 5 didapatkan hasil bahwa buzzer akan berbunyi pada tegangan 4.99 Volt ketika terdapat kendaraan yang memarkirkan kendaraannya secara sembrono, sementara itu ketika tidak terdapat pelanggaran parkir maka tegangan pada buzzer 0 Volt.

V. KESIMPULAN

1. Pada Aplikasi *Blynk* apabila suatu slot mendeteksi mobil maka indikator LED akan berwarna hijau dan jika tidak ada mobil maka berwarna hitam.

2. Besar tegangan potensiometer pada sensor dapat berpengaruh terhadap inframerah yang dipancarkan, yaitu semakin besar tegangan maka panjang cahaya inframerah yang dipancarkan akan semakin panjang, begitupun sebaliknya.
3. Modul WiFi ESP8266 dapat tersambung Aplikasi *Blynk* dan dapat mengirimkan data dengan rata-rata delay 5.21 detik untuk kartu A dan 6.66 detik untuk kartu B.
4. Buzzer dapat berfungsi dengan baik ketika terdapat pengendara yang memarkirkan mobilnya melewati pembatas parkir.

REFERENSI

- [1] Simor Technology (2020). "Prinsip Kerja Blynk" (diakses pada tanggal 10 Maret 2021)
- [2] Wahyudi, Ahmad. (2018). "Sensor Inframerah", (diakses pada tanggal 25 Agustus 2021)
- [3] Maulana, A. (2019). "perancangan sistem kendali Traffict Light secara realtime berbasis IoT", (diakses pada tanggal 8 April 2021)
- [4] Febriyanto,. 2014. "Apa Itu Arduino Uno?" (diakses pada tanggal 16 Mei 2021)
- [5] Fahreza, Aji (2017). "Menggunakan Buzzer Komponen Suara.", (diakses pada 10 Maret 2021)
- [6] Fahreza, Aji (2017). "Menggunakan Buzzer Komponen Suara.", (diakses pada 10 Maret 2021)
- [7] Adani, Muhammad Robith. (2020). "Mengenal Apa Itu Internet of Things dan Contoh Penerapannya" (diakses pada tanggal 10 Maret 2021)
- [8] Awaludin, Lukman. (2018). "Mengenal Wireless Sensor Network", (diakses pada tanggal 25 Agustus 2021)
- [9] Darshil, Patel. (2017). "Mengontrol LED Menggunakan Blynk", (diakses pada tanggal 8 April 2021)
- [10] Razor, Aldy. 2020. "Push Button Arduino: Pengertian, Fungsi, dan Prinsip Kerja" (diakses pada tanggal 16 Mei 2021)
- [11] Susandi, Dony dkk. (2017). Perancangan Smart Parking System Pada Prototype Smart Office Berbasis Internet Of Things. Jurnal Seminar Nasional Sains dan Teknologi.
- [12] TF, Arya. (2018). "Aplikasi Wireless Sensor Network Untuk Sistem Monitoring", (diakses pada tanggal 25 Agustus 2021)
- [13] TG, Adithya. (2018). "Control Arduino Uno Using ESP8266 and Blynk (diakses Pada 8 April 2021)
- [14] Wihandato, Arif dkk . 2021. Rancang Bangun Prototipe Sistem Smart Parking Berbasis Iot Menggunakan Node Mcu Esp8266 Jurnal Teknik Elektro Dan Komputer Triac.
- [15] Adwindea, Ardana dkk. (2018). "Perancangan Aplikasi Tatap Muka Smart Open Parking Berbasis Internet of Things (IoT) pada Perangkat Android" Vol 7. Diakses 17 September 2021, dari Universitas Diponegoro.