

USER INTERFACE DALAM APLIKASI NAVIGASI PERJALANAN PAKET BERBASIS MOBILE

USER INTERFACE IN A MOBILE-BASED TRAVEL NAVIGATION APPLICATION

Muhammad Azdan Khalafatallah¹, Anton Siswo Raharjo Ansori² M Husni Syahbani³

^{1,2,3} Universitas Telkom, Bandung

¹azdankhala@student.telkomuniversity.ac.id, ²raharjo@telkomuniversity.ac.id,

³eksmud@telkomuniversity.ac.id

Abstrak

Ekspedisi pengiriman paket adalah perusahaan yang bergerak pada pengiriman berupa paket. Pengiriman paket dilakukan oleh seorang kurir yang hanya mengandalkan pengetahuan wilayah dari kurir tersebut, ke tidak tahu akan rute perjalanan terkadang dapat berdampak pada efisiensi dan waktu sampainya paket ke penerima paket. Dengan adanya permasalahan tersebut tugas akhir ini bertujuan untuk memberikan solusi dalam memecahkan masalah tentang efisiensi pengambilan rute dan waktu perjalanan paket. Proses tersebut memerlukan data yang berguna untuk diproses agar dapat menghasilkan keluaran berupa prediksi pengambilan rute yang ideal, data tersebut dari masukan *user* berupa titik koordinat daerah tertentu dengan fitur GPS (*Global Positioning System*) selanjutnya data tersebut diolah oleh RPA (*Robotic Process Automation*) untuk dapat menghasilkan keluaran berupa pengambilan rute ideal untuk mempercepat proses pengiriman barang bagi *user* yang dapat dilakukan lewat aplikasi *mobile*. Berdasarkan hasil pengujian, dilakukannya percobaan untuk mengetahui jarak dan sudut terbaik untuk melakukan *scanning barcode*. Pada penelitian ini jarak dan sudut terbaik yaitu pada jarak 10 cm dan sudut 0° dengan rata-rata waktu paling singkat. Hasil yang didapat akan berbeda-beda, hal tersebut dapat terjadi karena terdapat faktor tambahan seperti cahaya, dan lama waktu kamera untuk fokus. Pada penelitian ini aplikasi berhasil menambahkan 100 alamat dan tersimpan ke dalam *database*.

Kata Kunci : Pengiriman paket, RPA (*Robotic Process automation*), aplikasi *mobile*, GPS (*Global Positioning System*)

Abstract

Expeditionary package delivery is a company engaged in the delivery of packages. Package delivery is carried out by a courier who only relies on regional knowledge from the courier, ignorance of the route of travel can sometimes have an impact on the efficiency and time of the package's arrival to the recipient of the package. With these problems, this final project aims to provide solutions in solving problems regarding the efficiency of the route taking and package travel time. The process requires user data to be processed in order to produce output in the form of predictions for the ideal route, the data from user input in the form of the coordinates of certain areas with the GPS (Global Positioning System) feature then the data is processed by RPA (Robotic Process Automation) to be able to produce output in the form of taking the ideal route to speed up the process of sending goods for users which can be done via mobile applications. Based on the test results, an experiment was conducted to determine the best distance and angle for scanning barcodes. In this study, the best distance and angle is at a distance of 10 cm and an angle of 0° with the shortest average time. The results obtained will vary, this can happen because there are additional factors such as light, and the length of time the camera takes to focus. In this study, the application successfully added 100 addresses and stored them in the database.

Keywords: Package delivery, RPA (*Robotic Process automation*), mobile applications, GPS (*Global Positioning System*)

1. Pendahuluan

Electronic commerce (e-commerce) adalah serangkaian kegiatan perdagangan yang dilakukan dengan sarana elektronik di internet. Sebanyak 88,1% pengguna internet di Indonesia memakai layanan *e-commerce* untuk membeli produk tertentu dalam beberapa bulan terakhir. Persentase tersebut merupakan yang tertinggi di dunia dalam hasil survei We Are Social pada April 2021[1]. Di Indonesia terdapat berbagai macam situs *e-commerce* seperti Tokopedia, Shopee, Bukalapak, Lazada dan masih banyak lagi. Para *e-commerce* ini menjalin sebuah kerja sama pada perusahaan yang bergerak pada bidang ekspedisi seperti JNE, SiCepat, dan Pos Indonesia yang nantinya digunakan untuk mempermudah para penjual agar mengirim barang mereka dengan selamat dan aman. Saat ini kurir atau pengantar paket memiliki beberapa kesulitan yang dialami, seperti ketidaktahuan jalan, pemilihan rute yang salah, dan juga terlalu banyaknya paket yang menumpuk sehingga kurir tidak dapat mengurutkan alamat dari yang terdekat dengan kurir sampai dengan yang terjauh, ketiga hal tersebut dapat membuat pengantaran barang menjadi tidak sampai dengan waktu yang sudah ditentukan, karena itu para kurir harus paham akan rute yang mereka pilih, dan juga urutan pemilihan alamat dengan tepat dari yang terdekat dengan kurir sampai yang terjauh dengan kurir agar tidak terjadinya keterlambatan pada pengantaran paket. Untuk mengatasi masalah tersebut adalah dengan membuat aplikasi berbasis android dengan fitur yang memberikan kemudahan bagi kurir dalam memilih rute yang tepat dan efisien, dengan cara melakukan *scan barcode* terhadap setiap paket yang akan dikirim ke konsumen dan nantinya aplikasi ini memberitahu kurir rute terbaik untuk

dilalui. Aplikasi ini dibuat dengan menggunakan algoritma *shortest path* sebagai penentu titik-titik rute perjalanan yang akan ditempuh kurir secara berurut dari rute yang terdekat dengan lokasi kurir hingga rute terjauh.

2. Kajian Pustaka

1. Google Map API

API (*Application Programming Interface*) merupakan suatu dokumentasi yang terdiri dari *interface*, fungsi, kelas, struktur dan lainnya. Dengan adanya API ini dapat membantu *programmer* dalam membuat suatu *software* untuk dapat dikembangkan dengan perangkat lunak yang lain. Keunggulan dari API adalah dapat menghubungkan suatu aplikasi dengan aplikasi lainnya untuk saling berinteraksi. Bahasa yang digunakan adalah HTML, JavaScript, AJAX, dan XML[2]. Google menyediakan layanan Google Map API yang memungkinkan para *developer* untuk memasukkan Google map ke dalam aplikasi. Diperlukannya API *key* untuk menampilkan Google map di dalam aplikasi. API *key* merupakan kode unik yang diberikan oleh Google untuk aplikasi.

2. Polyline

Polyline digunakan dalam mengembangkan aplikasi ini untuk menampilkan rute sehingga memudahkan *driver* dalam melakukan proses penjemputan dan pengantaran. Algoritma polyline adalah algoritma yang digunakan untuk menampilkan atau menunjukkan rute tertentu antara dua titik pada Google map. *Encode* polyline adalah satu set pasangan koordinat yang telah diubah menjadi *string ASCII* untuk secara signifikan mengurangi keseluruhan ukuran data. Polyline yang dikodekan digunakan untuk menyimpan kumpulan data koordinat besar untuk memproyeksikan garis atau bentuk pada peta. Koordinat titik di-*encode*-kan menggunakan nilai yang di masukan[3]. Format *encode* di dalam polyline harus mewakili dua koordinat yang merepresentasikan garis lintang dan bujur hingga presisi yang wajar, polyline dapat di tampilkan jika sudah menentukan titik awal dan titik akhir perjalanan.

3. Geolocator

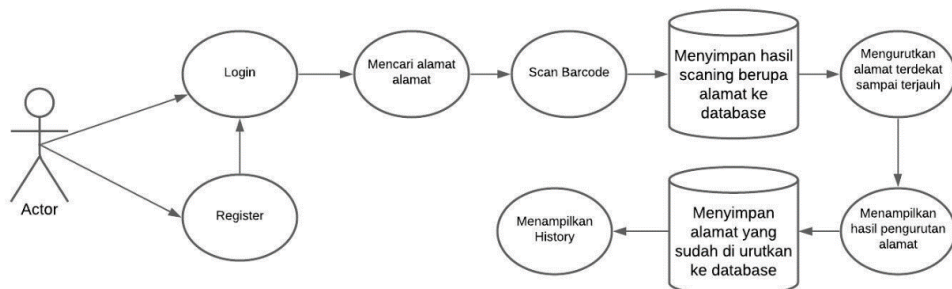
Geolocator adalah sistem identifikasi lokasi geografis dunia nyata atas suatu objek, seperti sumber radar, ponsel atau terminal komputer yang tersambung ke *internet*. *Geolocator* bisa diartikan sebagai pencarian lokasi yang dapat diakses secara *real time*. *Geolocator* berkaitan dengan penggunaan sistem penentuan posisi yang lebih spesifik dan bermakna, bukan hanya satu set koordinat saja (seperti alamat jalan, lokasi desa terpencil)[4]. Implementasinya adalah untuk mengetahui posisi terkini *user* secara *real time*.

4. Qr & Barcode

Barcode adalah sekumpulan kode untuk mendefinisikan huruf dan angka yang terdiri dari kombinasi garis dengan pengaturan jarak yang berbeda-beda. Aturan tersebut merupakan metode untuk dapat memasukkan data ke dalam komputer. Informasi pada *Barcode* berisi enkripsi dari sejumlah digit angka. Saat *barcode* tersebut di *scan* dengan alat *barcode scanner*, maka kode tersebut secara otomatis terhubung ke data barang yang sudah disimpan dalam *database*. Hasil dari pemindaian tersebut berisi data-data dari berbagai produk seperti nama vendor, nama produk, harga dan data lainnya sesuai dengan apa yang sudah dimasukkan pada *database* [5].

3. Implementasi Pengujian

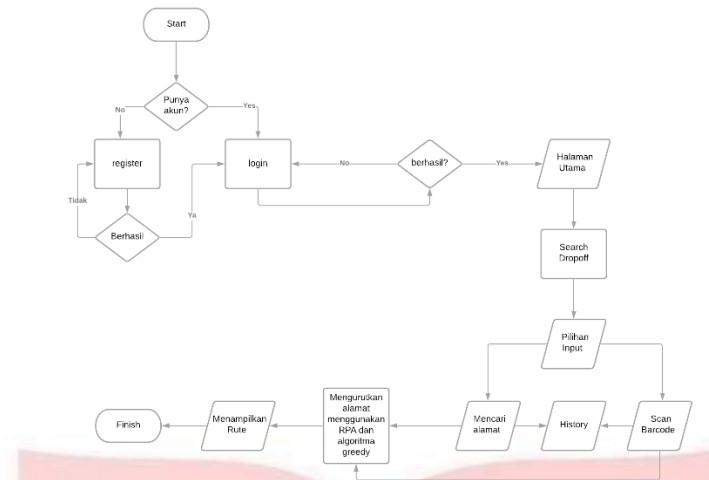
3.1 Gambaran Umum Sistem



Gambar 3. 1 Gambaran Umum Sistem

Aplikasi navigasi perjalanan paket ini berbasis android menggunakan flutter pada bagian *frontend* dan *firebase realtime database* pada bagian *backend*. *Scanning barcode* merupakan salah satu fitur yang terdapat pada aplikasi ini. Berdasarkan pada gambar, alur sistem akan dilakukan dimulai dari registrasi akun jika belum memiliki atau *login* jika sudah memiliki akun. Setelah *login* berhasil akan diarahkan ke halaman utama dan *user* dapat mencari alamat melalui *scan barcode* atau secara manual setelah melakukan pencarian alamat, alamat tersebut akan tersimpan ke dalam *database* dan nantinya alamat akan diurutkan dari alamat terdekat pada kurir sampai dengan alamat terjauh. Setelah alamat sudah diurutkan data tersebut akan tersimpan ke dalam *database* dan ditampilkan di *menu history*.

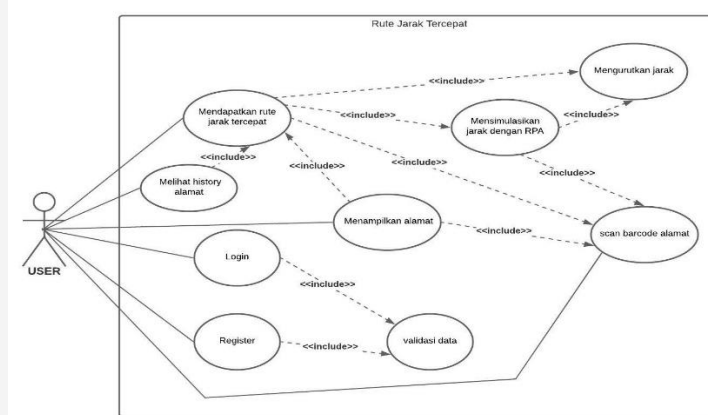
3.2 Flowchart Sistem



Gambar 3. 2 Gambaran Flowchart Sistem

Untuk memulai aplikasi ini dengan membuka aplikasi tersebut lalu di halaman utama terdapat dua pilihan yaitu *login* dan *register*, *user* diharuskan untuk *register* terlebih dahulu jika belum mempunyai akun, setelah itu *login* untuk masuk ke dalam aplikasi jika tidak berhasil *user* akan kembali ke halaman utama. Setelah *login* berhasil lalu aplikasi akan menampilkan Google maps yang belum terdapat titik koordinat alamat yang dituju, oleh karena itu *user* diharuskan memilih menu *search dropoff* lalu akan muncul tampilan *where to*, *scan barcode*, dan *history* untuk meng-*input* alamat secara manual bisa dengan cara menekan kolom *where to* untuk meng-*input* alamat secara otomatis bisa dengan cara menekan tombol *scan barcode* setelah melakukan peng-*input*-an alamat *user* akan dikembalikan ke halaman maps yang sudah terdapat jalan yang dapat *user* lalui.

3.3 Diagram Use Case

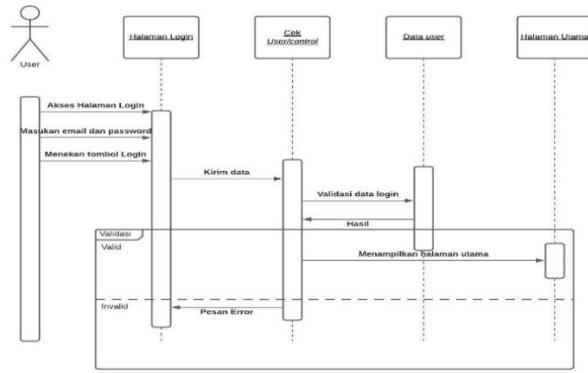


Gambar 3. 3 Use Case Diagram System

Pada diagram *use case* di atas terdapat dua aksi yang dapat *user* lakukan yaitu *sign up* dan *login* di mana jika *user* sebelumnya belum mempunyai akun, *user* dapat melakukan proses *sign up* terlebih dahulu agar nantinya dapat masuk ke aplikasi dan akun *user* akan tersimpan di *database* jika sudah selesai melakukan *sign up*, *user* dapat memilih *login* untuk masuk ke dalam aplikasi. Opsi selanjutnya *user* dapat melakukan *scan barcode* di setiap paket yang ada dengan menggunakan kamera, setiap alamat yang sudah dipindai akan masuk ke *database* setelah *user* selesai melakukan *scan* di setiap alamat *user* akan ditampilkan jarak tujuan dan tujuan mana yang akan diambil lebih dahulu yang mana sudah diproses oleh RPA (*Robotic Process Automation*).

3.4 Pemodelan proses login aplikasi navigasi perjalanan paket untuk user.

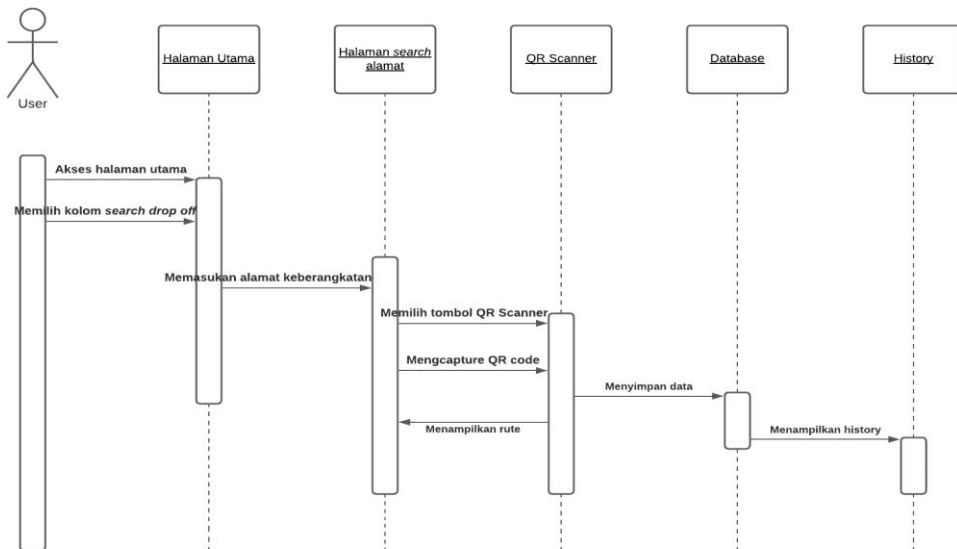
Pada pemodelan proses *login* untuk *user* ini digambarkan menggunakan *sequence diagram* yang berfungsi untuk mendeskripsikan bagaimana *user* melakukan proses *Login*. terlihat pada gambar 3.3.



Gambar 3. 4 Sequence Diagram Proses Login

3.5 Pemodelan proses memasukkan alamat pada aplikasi navigasi perjalanan mobile untuk user.

Pada pemodelan proses memasukkan alamat untuk user ini digambarkan menggunakan *sequence diagram* terlihat pada gambar 3.4.



Gambar 3.4 Sequence Diagram Memasukkan Alamat

4. Implementasi dan Pengujian Sistem

4.1 Perangkat yang digunakan

Mobile yang digunakan memiliki spesifikasi sebagai berikut:

- a. Jenis Perangkat : Android
- b. Merek dan Model : Oppo A3S
- c. RAM : 3 GB
- d. Operating System (OS) : Android Oreo 8.1.0
- e. Processor : Qualcomm SDM 450 Octa Core
- f. Memory : 32 GB

4.2 Pengujian Fungsional

Tabel 4.1 Pengujian fungsional pada *menu* utama

NO	Fitur	Pengujian yang dilakukan	Keluaran yang diharapkan	Hasil		Jenis Pengujian
				Sesuai	Tidak Sesuai	
1	<i>Sign Up</i>	Menekan tombol <i>SignUp</i>	Masuk ke menu <i>SignUp</i>	✓		<i>Black box</i>
2	<i>Login</i>	Menekan tombol <i>Login</i>	Masuk ke menu <i>Login</i>	✓		<i>Black box</i>
3	<i>Scan</i>	Memindai <i>barcode</i>	Menampilkan rute alamat	✓		<i>Black box</i>
4	<i>Search Dropoff</i>	Menekan tombol <i>search dropoff</i>	Dapat mencari alamat	✓		<i>Black box</i>
5	<i>Distance</i>	Melakukan pencocokan antara aplikasi yang dibuat dengan Google maps	Menampilkan jarak yang sesuai	✓		<i>Black box</i>
6	<i>logout</i>	Menekan tombol <i>logout</i>	Kembali ke menu <i>login</i>	✓		<i>Black box</i>
7	<i>History</i>	Mencocokkan antara pencarian alamat yang sudah dilakukan	Menampilkan alamat mana saja yang sudah di cari	✓		<i>Black box</i>
8	<i>Your location</i>	Mencocokkan antara posisi terkini dengan posisi yang ada di aplikasi	Menampilkan posisi dengan akurat	✓		<i>Black box</i>
9	<i>Zoom in / zoom out</i>	Menekan tombol <i>zoom out</i> dan <i>zoom in</i>	Google maps dapat diperbesar dan diperkecil	✓		<i>Black box</i>
10	<i>Location prediction</i>	Mencari alamat dengan menggunakan satu huruf	Menampilkan beberapa alamat dengan huruf depan yang sama	✓		<i>Black box</i>
11	<i>Exit</i>	Menekan tombol <i>exit</i>	Dapat mencari alamat baru	✓		<i>Black box</i>
12	<i>Next</i>	Menekan tombol <i>next</i>	Berpindah ke alamat selanjutnya		✓	<i>Black box</i>
13	<i>Pin point</i>	Menekan tombol <i>Pin point</i>	Menunjukkan <i>detail</i> alamat <i>user</i> maupun alamat tujuan	✓		<i>Black box</i>
14	<i>Direction</i>	Menekan tombol <i>Direction</i>	Membuka Google maps dan menunjukkan arah melalui Google maps	✓		<i>Black box</i>
15	<i>Maps</i>	Menekan tombol <i>Maps</i>	Membuka lokasi yang dituju lewat Google maps	✓		<i>Black box</i>
16	<i>Remove marker</i>	Menekan tombol <i>remove marker</i>	Menghapus <i>pin point</i> yang berada di map		✓	<i>Black box</i>

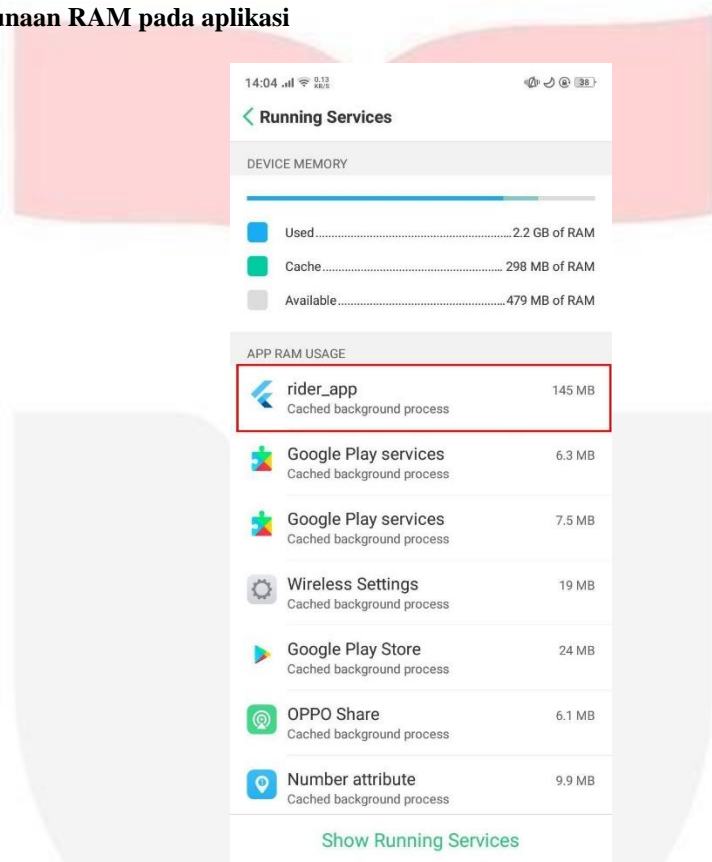
4.1 Pengujian lama waktu dalam membuka aplikasi

```
I/flutter (15049): This is your current location :: , Jalan Sukabirus, Sukapura, Kecamatan Dayeuhkolot, Kota Bandung, Jawa Barat  
I/.Azdan.riderap(15049): Background concurrent copying GC freed 109751(3476KB) AllocSpace objects, 59(2372KB) LOS objects, 49% free, 7561KB/14MB, paused 6  
98us total 162.693ms
```

Gambar 4. 1 Hasil Pengujian Dalam Membuka Aplikasi

Pada pengujian ini dilakukannya proses waktu membuka aplikasi menggunakan visual studio code dapat dilihat gambar di atas bahwa waktu untuk membuka aplikasi adalah 162.694ms.

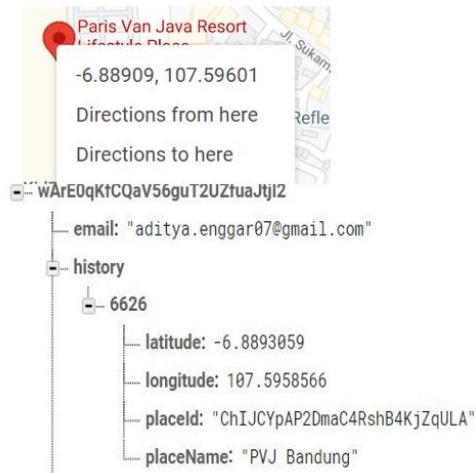
4.2 Pengujian penggunaan RAM pada aplikasi



Gambar 4. 2 Hasil Pengujian Konsumsi RAM

Pada pengujian ini dilakukan pengecekan konsumsi RAM yang dipakai saat menjalankan aplikasi dapat diketahui bahwa RAM yang digunakan dalam menjalankan aplikasi terpakai sebanyak 145 Mb.

4.3 Pengujian koordinat *scanning barcode*



Gambar 4. 3 Hasil Pengujian *Scanning Barcode*

Dapat dilihat gambar di atas adalah hasil dari *scanning barcode* aplikasi dan titik koordinat yang berada di google maps.

4.4 Pengujian parameter *scanning barcode*

Tabel 4. 2 Tabel Pengujian Barcode

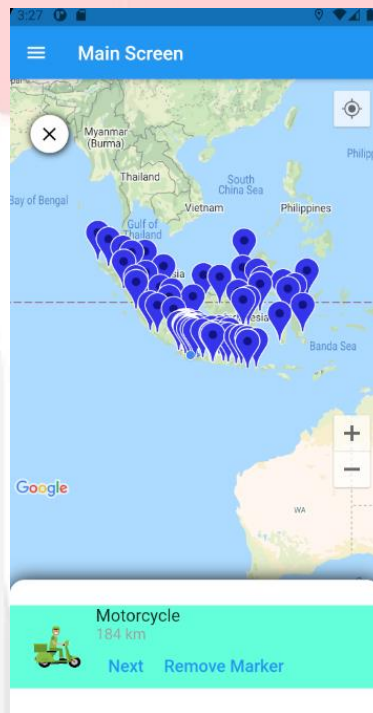
Jarak	Sudut	Percobaan ke-	Hasil (Detik)					
			Waktu Kiri	Waktu Kanan				
10 cm	0°	1	Tidak Bisa					
		2						
		3						
	15°	1						
		2						
		3						
	30°	1			3,27	2,30		
		2			5,12	2,75		
		3			4,72	2,37		
10 cm	45°	1	2,52	1,66				
		2	1,80	1,27				
		3	4,20	2,64				
	60°	1	2,80	1,99				
		2	1,89	2,36				
		3	1,72	1,50				
	90°	1	0,72					
		2	0,97					
		3	0,70					
15 cm	0°	1	Tidak Bisa					
		2						
		3						
	15°	1			Tidak Bisa			
		2						
		3						
	30°	1					2,81	2,05
		2					2,91	2,02
		3					3,26	1,49
15 cm	45°	1	3,45	9,45				
		2	5,78	2,65				
		3	2,85	5,40				

Jarak	Sudut	Percobaan ke-	Hasil (Detik)	
			Waktu Kiri	Waktu Kanan
	60°	1	1,89	1,32
		2	1,57	1,13
		3	1,25	1,02
	90°	1	1,74	
		2	1,14	
		3	1,31	
20 cm	0°	1	Tidak Bisa	
		2		
		3		
	15°	1		
		2		
		3		
	30°	1	1,74	1,77
		2	1,74	1,36
		3	1,78	1,69
20 cm	45°	1	3,25	4,44
		2	5,14	4,87
		3	4,72	11,75
	60°	1	1,87	2,14
		2	2,12	2,34
		3	1,78	1,50
	90°	1	1,08	
		2	1,94	
		3	1,29	
25 cm	0°	1	Tidak Bisa	
		2		
		3		
	15°	1		
		2		
		3		
	30°	1	x	2,26
		2	x	2,12
		3	x	3,22
25 cm	45°	1	5,20	4,75
		2	3,57	4,62
		3	3,62	2,91
	60°	1	1,45	1,99
		2	1,83	1,51
		3	2,13	1,86
	90°	1	0,95	
		2	2,01	
		3	1,99	
30 cm	0°	1	Tidak Bisa	
		2		
		3		
	15°	1		
		2		
		3		
	30°	1	3,45	2,26
		2	1,95	2,11
		3	1,65	2,38
30 cm	45°	1	5,40	2,56
		2	4,64	2,82
		3	3,87	13,20

Jarak	Sudut	Percobaan ke-	Hasil (Detik)	
			Waktu Kiri	Waktu Kanan
10cm	60°	1	1,87	2.19
		2	1,39	1.91
		3	2,19	2,14
	90°	1	1,41	
		2	1,98	
		3	1,49	

Dapat dilihat pada tabel di atas bahwa jarak dan sudut terbaik untuk melakukan *scanning barcode* ada pada jarak 10cm dan sudut 0°

4.5 Pengujian *pin point* alamat



Gambar 4. 4 Hasil Pengujian *Pin Point*

Dapat dilihat pada gambar di atas bahwa aplikasi dapat meng-*input*-kan alamat sebanyak 100 kali.

3 Kesimpulan dan Saran

5.2 Kesimpulan

Setelah dilakukannya penelitian ini, maka dapat diambil kesimpulan yaitu:

1. Hasil dari pengujian *scan barcode* dengan menggunakan metode *black box* untuk mengetahui seberapa cepat aplikasi dapat melakukan proses *scanning barcode* dapat berfungsi dengan baik dan sesuai seperti perancangan yang dibuat.
2. Pada pengujian yang dilakukan terhadap 30 *user*, diketahui bahwa rata-rata 83.3% responden pernah mengalami keterlambatan dalam pengiriman paket.
3. Berdasarkan hasil pengujian waktu *scanning barcode* dengan menggunakan metode *black box* dapat disimpulkan ada faktor lain yang mempengaruhi waktu dalam proses *scanning barcode* yaitu cahaya, dan lama waktu kamera untuk fokus.

5.3 Saran

Setelah dilakukannya penelitian ini maka penulis dapat menyarankan agar ke depannya penelitian ini bisa dibuat dengan *user interface* yang lebih menarik lagi.

Referensi:

- [1] A. Lidwina, "Penggunaan E-Commerce Indonesia Tertinggi di Dunia," 10 Negara dengan Persentase Penggunaan E-Commerce Tertinggi di Dunia (April 2021), p. 1, 2021.
- [2] J. A. J. A. Anton Wahyudi, "Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Kristen Petra," PERANCANGAN DAN PEMBUATAN APLIKASI TRACKER, p. 7, 2013.
- [3] gP. Yusup Supriyadi, "Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Budi Luhur," MPLEMENTASI LOCATION BASED SERVICE DAN ALGORITMA POLYLINE PADA APLIKASI PEMESANAN MOBIL DI UNIVERSITAS BUDI LUHUR BERBASIS ANDROID, vol. 1, p. 7, 2018.
- [4] K. F. King, "The Columbia," GEOLOCATION AND FEDERALISM ON THE INTERNET: CUTTING INTERNET GAMBLING'S, p. 35, 2010.
- [5] M. S. P. A. M. I Putu Alit Putra Yudha, "Program Studi Teknik Elektro, Fakultas Teknik, Universitas Udayana," PERANCANGAN APLIKASI SISTEM INVENTORY, p. 9, 2017.
- [6] S. M. M. A. M. Novan Adi Musthofa, "Universitas Pesantren Tinggi Darul 'UlumJombang," IMPLEMENTASI QUICK RESPONSE (QR) CODEPADA APLIKASI VALIDASI DOKUMEN MENGGUNAKANPERANCANGAN UNIFIED MODELLING LANGUAGE(UML), vol. 10, p. 9, 2016.

