



**PENGEMBANGAN APLIKASI PERANGKAT BERGERAK
BERBASIS LOKASI UNTUK PERENCANAAN PERJALANAN
WISATA BLITAR BERDASARKAN RUTE TERPENDEK**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Abidin Adi Prasetyo

NIM: 155150201111067



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2020**



Repository Universitas Brawijaya
Repository Universitas Brawijaya
Repository Universitas Brawijaya
Repository Universitas Brawijaya
Repository Universitas Brawijaya
Repository Universitas Brawijaya

Repository Universitas Brawijaya
Repository Universitas Brawijaya
Repository Universitas Brawijaya
Repository Universitas Brawijaya
Repository Universitas Brawijaya
Repository Universitas Brawijaya

Repository
Repository
Repository
Repository
Repository
Repository

PENGESAHAN

PENGEMBANGAN APLIKASI PERANGKAT BERGERAK BERBASIS LOKASI UNTUK
PERENCANAAN PERJALANAN WISATA BLITAR BERDASARKAN RUTE TERPENDEK

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Nama Abidin Adi Prasetyo
NIM: 155150201111067

Skripsi ini telah diuji dan dinyatakan lulus pada
6 Januari 2020

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Agi Putra Kharisma, S.T, M.T
NIK: 201304 860430 1 001

Issa Arwani, S.Kom., M.Sc.
NIP: 198309 22201212 1 003

Mengetahui

Ketua Jurusan Teknik Informatika



Tri Astoto Kurniawan, S.T, M.T, Ph.D
NIP: 19710518 200312 1 001

Repository Universitas Brawijaya
Repository Universitas Brawijaya
Repository Universitas Brawijaya
Repository Universitas Brawijaya

Repository Universitas Brawijaya
Repository Universitas Brawijaya
Repository Universitas Brawijaya
Repository Universitas Brawijaya

Repository
Repository
Repository
Repository



PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 12 Desember 2019

Abidin Adi Prasetyo

NIM: 155150201111067



PRAKATA

Puji syukur kehadiran Allah SWT yang telah melimpahkan rahmat, taufik dan hidayah-Nya sehingga laporan skripsi yang berjudul "PENGEMBANGAN APLIKASI PERANGKAT BERGERAK BERBASIS LOKASI UNTUK PERENCANAAN PERJALANAN WISATA BLITAR BERDASARKAN RUTE TERPENDEK" ini dapat terselesaikan. Penulis menyadari bahwa laporan ini tidak akan berhasil tanpa bantuan dari beberapa pihak. Oleh karena itu, penulis ingin menyampaikan rasa hormat dan terima kasih kepada:

1. Agi Putra Kharisma, S.T., M.T selaku dosen pembimbing satu yang telah dengan sabar membimbing dan mengarahkan penulis sehingga dapat menyelesaikan laporan ini.
2. Issa Arwani, S.Kom., M.Sc. selaku dosen pembimbing dua yang telah dengan sabar membimbing dan mengarahkan penulis sehingga dapat menyelesaikan laporan ini.
3. Bapak Agus Wahyu Widodo, S.T, M.Cs. selaku ketua Program Studi Teknik Informatika.
4. Bapak Tri Astoto Kurniawan, S.T., M.T., Ph.D. selaku ketua Jurusan Teknik Informatika.
5. Ayah dan Ibu dan seluruh keluarga besar atas segala nasehat, kasih sayang, perhatian serta yang senantiasa tiada henti-hentinya memberikan doa dan semangat demi terselesaikannya laporan ini.
6. Seluruh civitas akademika Teknik Informatika Universitas Brawijaya yang telah banyak memberi bantuan dan dukungan selama penyelesaian laporan Skripsi ini.

Penulis menyadari bahwa dalam penyusunan laporan ini masih banyak kekurangan, sehingga saran dan kritik yang membangun sangat penulis harapkan. Akhir kata penulis berharap skripsi ini dapat membawa manfaat bagi semua pihak yang menggunakannya.

Malang, 12 Desember 2019

Penulis

abidinadip@student.ub.ac.id



ABSTRAK

Abidin Adi Prasetyo, Pengembangan Aplikasi Perangkat Bergerak Berbasis Lokasi Untuk Perencanaan Perjalanan Wisata Blitar Berdasarkan Rute Terpendek

Pembimbing: Agi Putra Kharisma, S.T, M.T dan Issa Arwani, S.Kom, M.Sc

Meningkatnya jumlah kunjungan wisatawan di Kota Blitar tiap tahun berdampak pada bertambahnya titik-titik lokasi tempat wisata yang ada di Blitar. Dengan Fakta bahwa terdapat banyak tempat wisata di wilayah Blitar dengan lokasi yang berdekatan maka membuat kecenderungan wisatawan untuk mengunjungi beberapa tempat wisata dalam sekali rute perjalanan. Wisatawan tentunya menginginkan rute yang optimal dengan jarak terpendek untuk perjalanan wisata mereka sehingga dapat menghemat waktu serta biaya bahan bakar kendaraan mereka. Permasalahan dalam menentukan rute dengan jarak terpendek adalah terdapat banyak opsi jalur yang ada pada antarlokasi karena pada kenyataannya dari tempat A ke tempat B saja tidak hanya memiliki satu jalur. Dengan permasalahan tersebut peneliti mengembangkan Aplikasi Perangkat Bergerak Berbasis Lokasi Untuk Perencanaan Perjalanan Wisata Blitar Berdasarkan Rute Terpendek dengan menggunakan pendekatan Traveling Salesman Problem untuk menyelesaikan permasalahan ini. Dalam tahap perancangannya menggunakan metode prototyping supaya bila ada perubahan rancangan dapat dilakukan iterasi untuk proses revisi atau perbaikan. Pada tahap pengujian ini menggunakan pengujian validasi pada kebutuhan fungsional dan non fungsional serta pengujian usabilitas menggunakan Kuisisioner System Usability Scale. Berdasarkan hasil pengujian pada pengembangan aplikasi berstatus valid pada kebutuhan fungsional dan non-fungsional serta dapat dijalankan di beberapa perangkat Android.

Kata kunci: *Traveling Salesman Problem, Java, Android, Prototyping, Validasi, Usabilitas, Kuisisioner System Usability Scale.*



ABSTRACT

Abidin Adi Prasetyo, Location-Based Mobile Device Development Application for Blitar Travel Planning Based on the Shortest Route

Supervisors: Agi Putra Kharisma, S.T, M.T and Issa Arwani, S.Kom, M.Sc.

The increasing number of tourist arrivals in Blitar City each year has an impact on the increasing points of tourist sites in Blitar. The fact that there are many tourist attractions in the Blitar region close to the location makes the tendency of tourists to visit several tourist attractions in one travel route. Tourists certainly want the optimal route with the shortest distance for their tour so that they can save time and fuel costs of their vehicles. The problem in determining the route with the shortest distance is that there are many path options that exist between locations because in reality from place A to place B only not only has one pathway. With these problems researchers developed the Location Based Mobile Device Application for Blitar Travel Planning Based on the Shortest Route by using the Traveling Salesman Problem approach to solve this problem. In the design phase it uses the prototyping method so that if there are changes to the design iterations can be carried out for the revision or improvement process. At this stage of testing, it uses validation testing on functional and non-functional requirements and reusability testing using the System Usability Scale Questionnaire. Based on the test results on the development of applications that are valid on functional and non-functional requirements and can be run on several Android devices.

Keywords: Traveling Salesman Problem, Java, Android, Prototyping, Validation, Usability, System Usability Scale Questionnaire.



DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
PRAKATA	iv
ABSTRAK	v
ABSTRACT	vi
DAFTAR ISI	vii
BAB 1 PENDAHULUAN	1
1.1 Latar belakang	1
1.2 Rumusan masalah	2
1.3 Tujuan	2
1.4 Manfaat	2
1.5 Batasan masalah	3
1.6 Sistematika pembahasan	3
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Kajian Pustaka	5
2.2 Aplikasi Perangkat Bergerak	5
2.3 Location Based Service	2
2.4 Traveling Salesman Problem	2
2.5 Google Distance Matrix Web Service	2
2.6 Metode Prototyping	2
2.7 Pengujian Perangkat Lunak	3
BAB 3 METODOLOGI PENELITIAN	4
3.1 Analisis Kebutuhan	4
3.2 Membuat Prototype	5
3.3 Menyesuaikan Prototype Dengan Keinginan User	5
3.4 Menggunakan Prototype	5
3.5 Perancangan Aplikasi	5
3.6 Implementasi	6
3.7 Pengujian Aplikasi	6
3.8 Kesimpulan	6
BAB 4 ANALISIS KEBUTUHAN	7



4.3.3 Use Case Diagram Iterasi 1.....	10
4.3.4.1 Merencanakan Rute Iterasi 1.....	11
4.3.4.2 Menampilkan Rekomendasi Rute Perjalanan Terpendek Iterasi 1.....	12
4.3.4 Prototype Iterasi 1.....	14
4.4.1 Use Case Diagram Iterasi ke-2.....	18
BAB 5 PERANCANGAN.....	21
5.3 Sequence Diagram.....	24
5.5 Perancangan Basis Data.....	27
BAB 6 IMPLEMENTASI.....	28
6.1.2 Spesifikasi Perangkat Lunak.....	29
6.3 Implementasi Basis Data.....	30
6.4 Implementasi Kode Program.....	31
BAB 7 PENGUJIAN.....	39
7.2 Pengujian Usabilitas.....	41
BAB 8 PENUTUP.....	42
8.1 Kesimpulan.....	42
8.2 Saran.....	42
DAFTAR REFERENSI.....	43
LAMPIRAN.....	45



DAFTAR TABEL

Tabel 4.1 Identifikasi Aktor Iterasi 1.....	13
Tabel 4.1 Identifikasi Aktor Iterasi 2.....	14
Tabel 4.3 Kebutuhan Fungsional Sistem Iterasi 1.....	15
Tabel 4.4 Kebutuhan Non-Fungsional Sistem Iterasi 1.....	16
Tabel 4.5 Use Case Scenario Merencanakan Rute Iterasi.....	18
Tabel 4.6 Use Case Scenario Menampilkan Rekomendasi Rute Perjalanan Terpendek Iterasi 1.....	19
Tabel 4.7 Use Case Scenario Melihat Navigasi dari Lokasi saat ini ke Lokasi Tempat Wisata yang dipilih Iterasi 1.....	20
Tabel 4.8 Kebutuhan Fungsional Sistem Iterasi 2.....	25
Tabel 4.9 Use Case Scenario Menambah Tempat Wisata Iterasi 2.....	16
Tabel 6.1 Spesifikasi Perangkat Keras Komputer.....	37
Tabel 6.2 Spesifikasi Perangkat Bergerak.....	37
Tabel 6.3 Spesifikasi Perangkat Lunak Komputer.....	38
Tabel 6.4 Spesifikasi Perangkat Lunak pada Perangkat Bergerak.....	38
Tabel 6.5 Struktur Tabel Basis Data Place.....	39
Tabel 7.1 Pengujian Validasi Kebutuhan Fungsional.....	50
Tabel 7.2 Pengujian Validasi Kebutuhan Non Fungsional.....	53
Tabel 7.3 Hasil Pengujian Usabilitas Sistem.....	54



DAFTAR GAMBAR

Gambar 2.1 Struktur Layer Platform Android.....	6
Gambar 2.2 Metode Prototyping.....	9
Gambar 2.3 Diagram Blok Metodologi Penelitian.....	10
Gambar 2.2 Metode Prototyping.....	11
Gambar 4.1 Use Case Diagram Iterasi 1.....	17
Gambar 4.2 Tampilan Prototype Halaman Utama Aplikasi Iterasi 1.....	21
Gambar 4.3 Tampilan Prototype Halaman Find Place Activity Iterasi 1.....	22
Gambar 4.4 Tampilan Prototype Halaman Seleted Place Activity Iterasi 1.....	23
Gambar 4.5 Tampilan Prototype Halaman Map Activity Iterasi 1.....	24
Gambar 4.6 Tampilan Prototype Halaman Utama Aplikasi Iterasi 2.....	28
Gambar 4.7 Tampilan Prototype Halaman Add Place Activity Iterasi 2.....	29
Gambar 5.1 Activity Diagram Use Case Merencanakan Rute.....	30
Gambar 5.2 Activity Diagram Use Case Menampilkan Rekomendasi Rute Perjalanan Terpendek.....	31
Gambar 5.3 Activity Diagram Use Case Melihat Navigasi dari Lokasi saat ini ke Lokasi Tempat Wisata yang dipilih.....	31
Gambar 5.4 Activity Diagram Use Case Menambah Lokasi Tempat Wisata.....	32
Gambar 5.5 Sequence Diagram Merencanakan Rute.....	33
Gambar 5.6 Sequence Diagram Rekomendasi Rute Perjalanan.....	33
Gambar 5.7 Sequence Diagram Melihat Navigasi dari Lokasi saat ini ke destinasi wisata.....	34
Gambar 5.8 Sequence Diagram Menambah Lokasi Tempat Wisata.....	34
Gambar 5.9 Rancangan Class Diagram.....	35
Gambar 5.10 Perencanaan Basis Data.....	36
Gambar 6.1 Implementasi Basis Data.....	39
Gambar 6.2 Halaman Utama Aplikasi.....	45
Gambar 6.3 Halaman Find Place Activity.....	46
Gambar 6.4 Halaman Selected Place Activity.....	47
Gambar 6.5 Halaman Map Activity.....	48
Gambar 6.6 Halaman Add Place Activity.....	46



DAFTAR LAMPIRAN

Lampiran 1 Kuisiener Pertanyaan SUS.....	59
Lampiran 2 Data Hasil Wawancara.....	60



BAB 1 PENDAHULUAN

1.1 Latar belakang

Berdasarkan Survey Pengembangan Potensi Pariwisata Dan Ekonomi Kreatif Dinas Pariwisata Dan Kebudayaan Kota Blitar terjadi perkembangan jumlah kunjungan wisatawan di Kota Blitar dari tahun ke tahun. Melihat perkembangan angka Pariwisata di Blitar tiap tahun maka berdampak pada bertambahnya titik-titik lokasi tempat wisata yang ada di Wilayah Blitar. Dengan Fakta bahwa terdapat banyak tempat wisata di wilayah Blitar dengan lokasi yang berdekatan maka membuat kecenderungan wisatawan untuk mengunjungi beberapa tempat Wisata Blitar dalam sekali rute perjalanan. Wisatawan tentunya menginginkan rute yang optimal dengan jarak terpendek untuk perjalanan wisata mereka sehingga dapat menghemat waktu serta biaya. Permasalahan dalam menentukan rute dengan jarak tempuh terpendek menimbulkan masalah disebabkan terdapat banyak ditemukan pilihan jalur dari suatu tempat ke tempat lain karena fakta di lapangan jika seseorang dari tempat A ke tempat B terdapat opsi pilihan banyak jalur yang dapat dilalui sehingga menimbulkan banyak pilihan rute.

Untuk solusi dari permasalahan tersebut penulis menggunakan pendekatan Travelling Salesman Problem (TSP). Travelling Salesman Problem di analogikan seperti seseorang yang melakukan kunjungan ke sejumlah titik lokasi dalam rute perjalanannya dengan asumsi dia boleh melewati satu tempat hanya satu kali dalam perjalanan tersebut. Hasil dari penyelesaian terhadap permasalahan TSP ini adalah untuk memperoleh jalur dengan rute terpendek dari banyaknya pilihan jalur.

Dalam sudut pandang platform dalam pembuatan tools untuk kasus ini penulis berlandaskan pada Data Survey Lembaga riset digital marketing Emarketer dimana memperkirakan pada 2018 jumlah pengguna aktif perangkat bergerak di Indonesia lebih dari 100 juta orang (Rudi Antara, 2018). Dengan jumlah sebesar itu, Indonesia akan menjadi negara dengan pengguna aktif perangkat bergerak terbesar keempat di dunia setelah Cina, India, dan Amerika. Hal tersebut membuat Aplikasi Perangkat Bergerak sebagai Platform yang tepat untuk merangkul sebanyak mungkin pengguna di Indonesia, terutama pengguna untuk Aplikasi Perangkat Bergerak Berbasis Lokasi untuk Perencanaan Perjalanan Wisata Blitar Berdasarkan Rute Terpendek.

Aplikasi ini menggunakan Teknologi Berbasis Lokasi yang merupakan sebuah teknologi yang dapat mengidentifikasi secara *real-time* posisi geografis keberadaan suatu objek tertentu, serta juga dapat mengidentifikasi lokasi terkini dari seseorang menggunakan GPS dan layanan teknologi Google Map API v2 pada Perangkat Bergerak, serta inti dari solusi permasalahan ini yaitu menyelesaikan permasalahan Traveling Salesman Problem menggunakan layanan Google Distance Matrix API yaitu sebuah layanan API dari Google yang dikhususkan untuk menyelesaikan permasalahan Traveling Salesman Problem.



1.2 Rumusan masalah

Penelitian ini memiliki rumusan masalah yaitu sebagai berikut :

1. Apa saja kebutuhan fungsional dan nonfungsional dari Aplikasi Perangkat Bergerak Berbasis Lokasi untuk Perencanaan Perjalanan Wisata Blitar Berdasarkan Rute Terpendek?
2. Bagaimana perancangan dari Aplikasi Perangkat Bergerak Berbasis Lokasi untuk Perencanaan Perjalanan Wisata Blitar Berdasarkan Rute Terpendek?
3. Bagaimana Implementasi dari Aplikasi Perangkat Bergerak Berbasis Lokasi untuk Perencanaan Perjalanan Wisata Blitar Berdasarkan Rute Terpendek?
4. Bagaimana hasil Pengujian dari Aplikasi Perangkat Bergerak Berbasis Lokasi untuk Perencanaan Perjalanan Wisata Blitar Berdasarkan Rute Terpendek?

1.3 Tujuan

Berdasarkan rumusan masalah diatas penelitian ini memiliki tujuan sebagai berikut:

1. Menentukan kebutuhan fungsional dan non fungsional dari Aplikasi Perangkat Bergerak Berbasis Lokasi Perencanaan Perjalanan Wisata Blitar Berdasarkan Rute Terpendek
2. Mengetahui hasil rancangan Aplikasi Perangkat Bergerak Berbasis Lokasi Perencanaan Perjalanan Wisata Blitar Berdasarkan Rute Terpendek.
3. Mengimplementasikan rancangan dari Aplikasi Perangkat Bergerak Berbasis Lokasi untuk Perencanaan Perjalanan Wisata Blitar Berdasarkan Rute Terpendek.
4. Memperoleh hasil pengujian Aplikasi Perangkat Bergerak Berbasis Lokasi untuk Perencanaan Perjalanan Wisata Blitar Berdasarkan Rute Terpendek.

1.4 Manfaat

Berdasarkan rumusan masalah diatas manfaat dari penelitian sebagai berikut:

1. Memberikan informasi kepada Wisatawan Blitar rute Optimal untuk mengunjungi beberapa tempat Wisata Blitar dalam sekali perjalanan.
2. Menerapkan pengetahuan yang didapat penulis selama perkuliahan tentang Pengembangan Aplikasi Perangkat Bergerak untuk menyelesaikan solusi permasalahan di masyarakat.
3. Sebagai bahan referensi sebagai referensi pada penelitian yang selanjutnya.



1.5 Batasan masalah

Batasan masalah penelitian ini supaya lebih terfokus yaitu sebagai berikut:

1. Aplikasi pada penelitian ini dikembangkan pada sistem operasi Android.
2. Pemakainya adalah wisatawan yang ingin mengunjungi beberapa lokasi tempat Wisata Blitar dalam sekali rute perjalanan.
3. Dalam penyelesaian masalah dalam penelitian ini penulis menggunakan pendekatan permasalahan Traveling Salesman Problem.

1.6 Sistematika pembahasan

Bagian ini menjelaskan struktur serta isi dari setiap Bab pada Penelitian.

Penjelasannya dapat dilihat dibawah ini:

BAB 1 PENDAHULUAN

Menjelaskan tentang latar belakang penelitian, rumusan masalah, tujuan, manfaat, batasan masalah dan sistematika pembahasan.

BAB 2 LANDASAN KEPUSTAKAAN

Berisi Informasi mengenai beberapa referensi studi literatur sebagai penunjang untuk membantu dalam proses pembuatan laporan skripsi.

BAB 3 METODOLOGI

Menjelaskan Bagaimana urutan dan tata cara peneliti dalam membuat laporan skripsi ini dan juga tahapan-tahapan dalam pengembangan aplikasi.

BAB 4 ANALISIS KEBUTUHAN

Membahas tahapan menggali kebutuhan yang diperoleh dari beberapa proses yaitu wawancara dan juga berdiskusi bersama calon pemangku kebutuhan dimana disini aktornya adalah wisatawan kota blitar

BAB 5 PERANCANGAN

Menjelaskan bagaimana peneliti merancang aplikasi mulai dari awal hingga siap diimplementasikan dimana kebutuhan pengguna telah diperoleh melalui tahap analisis kebutuhan.

BAB 6 IMPLEMENTASI

Menjelaskan penerapan dan pengerjaan dari rancangan yang telah dirancang oleh peneliti

BAB 7 PENGUJIAN



Menjelaskan proses mencari tahu kualitas dari aplikasi yang telah dibuat apakah sudah sesuai dengan yang direncanakan atau tidak. Peneliti menggunakan 2 Pengujian yaitu validasi dan usability

BAB 8 PENUTUP

Membahas bagaimana hasil kesimpulan dari penelitian ini termasuk juga saran dari peneliti setelah selesai melakukan penelitian supaya dapat berguna untuk pihak lain dan lebih baik untuk kedepannya.



BAB 2 LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Bagian pada laporan ini membahas tentang penelitian terdahulu yang dilaksanakan oleh pihak lain yang bisa dijadikan referensi penulis pada penelitian ini. Berikut penelitian yang dapat dijadikan referensi oleh peneliti :

Penelitian yang dilakukan oleh Helga Aditya Rizqi Geovani (2016) dengan judul “Implementasi Algoritma Dijkstra Untuk Mengetahui Lokasi Tempat Ibadah Umat Muslim Di Kota Malang Pada Aplikasi Mobile Phone (Studi Kasus Tempat Ibadah Di Wilayah Kecamatan Lowokwaru”. Pada penelitian tersebut penulis melakukan penelitian tentang implementasi algoritma Dijkstra untuk memberi informasi user mengenai rute terdekat menuju lokasi tempat ibadah pada aplikasi mobile phone di Kota Malang.

Lalu oleh Fahri Hikmawan (2017) yang melaksanakan penelitian dengan judul “Aplikasi Pencarian Rute Terpendek Bengkel Mobil Menggunakan Algoritma Dijkstra”. Pada penelitian tersebut penulis mengembangkan Aplikasi Perangkat Bergerak untuk mencari rute bengkel mobil terpendek dari lokasi pengguna menggunakan Algoritma Dijkstra.

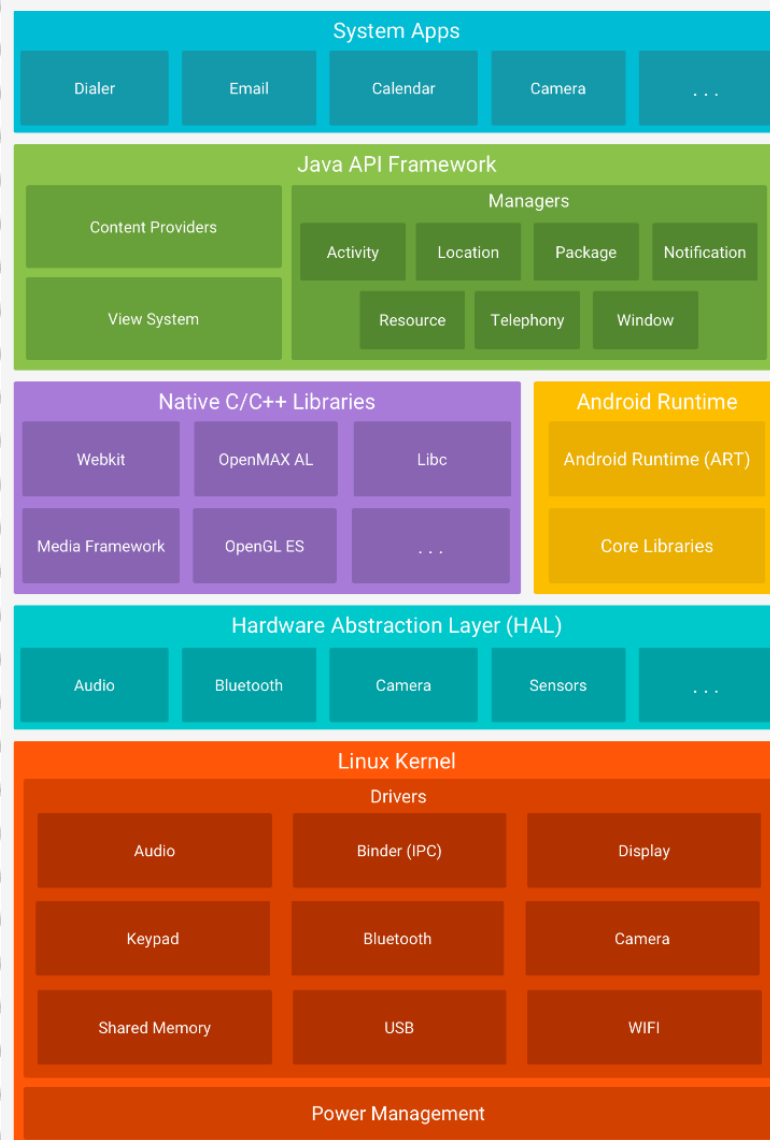
Penelitian yang dilakukan oleh Abdurrahman Albes Fajri, Endina Putri Purwandari dan Funny Farady Coastera (2018) dengan judul “Implementasi Sistem Informasi Geografis Untuk Pencarian Indeks Dengan Jarak Terpendek Menggunakan Metode Algoritma Dijkstra”. Dalam penelitian ini penulis mencantumkan bahwa Sistem Aplikasi ini sudah bisa digunakan oleh pengguna untuk mencari Indeks. Aplikasi ini dapat memberikan seputar informasi mengenai Indeks. Dengan menggunakan algoritma Dijkstra, rute perjalanan menuju lokasi Indeks berhasil menampilkan rute jarak terpendek menuju Indeks.

2.2 Aplikasi Perangkat Bergerak

Aplikasi perangkat bergerak merupakan aplikasi yang terdapat pada perangkat dan dapat digunakan ketika pengguna berpindah - pindah tempat. Perkembangan teknologi perangkat bergerak menunjukkan peningkatan yang signifikan di setiap tahunnya. Kemampuan aplikasi perangkat bergerak seperti *smartphone* sekarang ini dapat disejajarkan dengan kemampuan *desktop* tetapi kemampuan dalam hal proses program tetap dipegang oleh komputer. Pemrograman aplikasi perangkat bergerak tidak banyak berbeda dengan pemrograman pada *desktop*. Pada pemrograman aplikasi perangkat bergerak mempunyai kelebihan dalam beberapa aspek teknis karena adanya keterbatasan dibandingkan *desktop* (Budi Darytamo, 2007). Pada aplikasi layanan masyarakat di Rumah Sakit yang akan dibuat penulis bertujuan mengembangkan aplikasi yang dapat digunakan oleh pengguna di berbagai tempat, jadi pengguna cukup melakukan layanan di *smartphone*, dan layanan akan terhubung ke *database* sistem layanan rumah sakit.

2.2.1 Android

Sistem operasi berbasis *linux* yang dirilis pada tahun 2007 salah satunya adalah Android (Teguh, 2011). Sistem Android dibuat agar mengembangkan persaingan pasar perangkat seluler, khususnya perangkat Android dan komputer tablet. Pada Android umumnya menggunakan antarmuka manipulasi langsung oleh pengguna menggunakan sentuhan jari, misalnya menggeser, mencubit, dan mengetuk. Adapun struktur layer *platform* Android yang ditunjukkan pada Gambar 2.1.



Gambar 2.1 Struktur *Layer Platform* Android

(Sumber : Android Studio, 2019)



2.3 Location Based Service

Sistem Layanan Berbasis Lokasi, atau lebih dikenal dengan Location-based Services (LBS), menggabungkan antara proses dari layanan mobile dengan posisi geografis dari penggunanya. Poin pentingnya adalah ketika posisi target, di mana sebuah target bisa jadi adalah pengguna LBS itu sendiri atau entitas lain yang tergabung dalam suatu layanan (Kupper, 2005). LBS mengarah pada aplikasi-aplikasi dan layanan-layanan yang menyediakan informasi lokasi mengenai suatu pengguna mobile, untuk menyediakan sebuah layanan atau informasi khusus pada pengguna. Location Based Service memerlukan suatu mekanisme untuk melakukan koneksi dengan penyedia layanan lokasi (Location Provider), yang akan memberikan informasi posisi fisik device.

2.4 Traveling Salesman Problem

Traveling Salesman Problem merupakan permasalahan dimana seorang salesman harus mengunjungi setiap kota, salesman tersebut berangkat dari kota awal dan berakhir pada kota yang sama, serta setiap kota harus dikunjungi tidak lebih dari satu kali. Fungsi tujuan dari TSP adalah untuk mencari rute terpendek yang bisa dilalui oleh salesman sehingga dihasilkan total jarak terpendek (Munir, Rinaldi., 2005) Dimana dalam penelitian kali ini peneliti menggunakan layanan Google Distance Matrix API. Pada Layanan API tersebut menyelesaikan permasalahan TSP dari beberapa titik lokasi tempat yang dipilih menggunakan algoritma Brute Force dengan Teknik Exhaustive Search dimana Exhaustive search adalah teknik pencarian solusi secara solusi brute force untuk masalah yang melibatkan pencarian elemen dengan sifat khusus.

2.5 Google Distance Matrix Web Service

Suatu layanan yang memberikan informasi jarak dan waktu perjalanan untuk matriks asal dan tujuan. API mengembalikan informasi berdasarkan rute yang direkomendasikan antara titik awal dan akhir, sebagaimana dihitung oleh Google Maps API, dan terdiri dari baris yang berisi nilai durasi dan jarak untuk setiap pasangan. Menggunakan pendekatan Brute Force Search untuk mendapatkan rekomendasi rute dengan jarak tempuh terpendek

2.6 Metode Prototyping

Menurut Raymond McLeod (Sidik : 2013), prototype didefinisikan sebagai alat yang memberikan ide bagi pembuat maupun pemakai potensial tentang cara system berfungsi dalam bentuk lengkapnya, dan proses untuk menghasilkan sebuah prototype disebut prototyping. Prototype bisa dikatakan sebagai contoh desain dan juga contoh sistem yang sudah jadi, namun belum berfungsi secara sempurna. Prototype akan memberikan secara garis besar bagaimana sebuah sistem dapat bekerja, dan kebanyakan sistem yang menjadi prototype belumlah menjadi sebuah sistem yang sempurna untuk dijalankan. Pada dasarnya, dalam melakukan pengembangan sistem dengan menggunakan metode prototyping,



memiliki awalan yang sama seperti pada metode waterfall, yaitu analisa sistem dan juga kebutuhan user. Akan tetapi, analisa ini tidaklah berlaku general atau luas seperti pada waterfall, namun hanya mengarah pada suatu kebutuhan yang spesifik, misalnya User ingin sebuah sistem yang bisa melakukan proses penghitungan akuntansi secara cepat, sehingga sistem analis hanya akan membuat sebuah sistem yang bisa melakukan penghitungan dengan cepat, tanpa menganalisa desain, serta jenis bahasa pemrograman yang akan digunakan. Setelah analisis selesai, maka programmer akan langsung membuat prototype dari sistem tersebut, dan langsung mengujicobanya dengan bantuan user. Setiap complain dan juga kekurangan serta kelebihan dari sistem tersebut kemudian akan dicatat dan kemudian dilakukan pengembangan terus menerus, hingga akhirnya sistem tersebut selesai dan bisa digunakan secara penuh.

2.7 Pengujian Perangkat Lunak

Menurut Galin (2004) Pengujian Perangkat Lunak merupakan proses formal yang ditentukan oleh tim pengujian yang meliputi unit perangkat lunak, beberapa unit perangkat lunak terintegrasi atau seluruh package perangkat lunak yang ditentukan oleh program yang berjalan di komputer. Seluruh tes saling terkait dan adanya prosedur pengujian dan kasus pengujian. Tujuan pengujian Perangkat Lunak adalah Menilai apakah perangkat lunak yang dikembangkan telah memenuhi kebutuhan pemakai, menilai apakah tahap pengembangan perangkat lunak telah sesuai dengan metodologi yang digunakan, Membuat dokumentasi hasil pengujian yang menginformasikan kesesuaian perangkat lunak yang diuji dengan spesifikasi yang telah ditentukan

Berikut adalah pengujian aplikasi yang dilakukan penulis:

1. Pengujian Validasi

Pengujian ini bertujuan untuk memastikan apakah semua elemen konfigurasi *software* telah dikembangkan dengan tepat. Dari kebutuhan fungsional sistem layanan masyarakat, dibuatlah *test case* yang digunakan untuk memeriksa hasil *output* dari sistem apakah berjalan sesuai harapan atau tidak. Dalam proses sebelumnya dituliskan data apa saja yang dibuat percobaan, jika data telah diisikan maka hasil *output* akan dituliskan dalam *result* pengujian, dari hasil *result* akan diketahui fungsi sistem telah berjalan sesuai harapan atau tidak sesuai harapan.

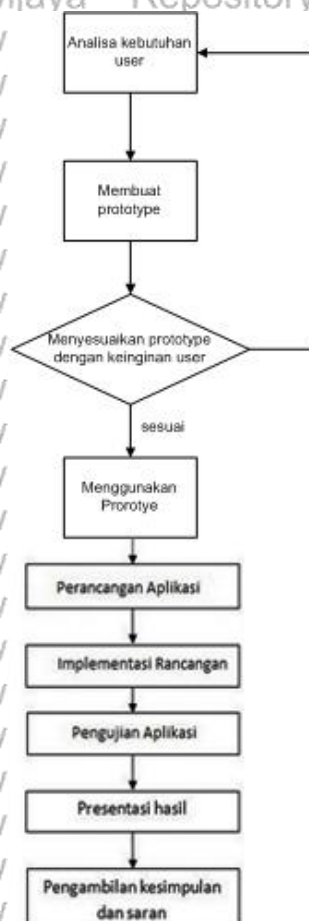
2. Pengujian Usabilitas

Pengujian usabilitas merupakan pengujian untuk memberi nilai baik atau buruknya produk (aplikasi atau website). Pada tahap ini pengguna menjalankan semua fitur aplikasi atau website, selanjutnya pengguna mengisi kuesioner yang berisi pernyataan tentang penggunaan aplikasi, hasil kuesioner nantinya akan diolah untuk melakukan penilaian dari usabilitas aplikasi.



BAB 3 METODOLOGI PENELITIAN

Pada bagian ini dibahas tentang bagaimana metode penulis dalam melaksanakan penelitian ini. Berikut diperlihatkan di Gambar 3.1



Gambar 3.1 Diagram Blok Metodologi Penelitian

3.1 Analisis Kebutuhan

Pada bagian penelitian ini dilaksanakan penggalan kebutuhan dalam perancangan Aplikasi. Dalam pembuatan Aplikasi tersebut yang pertama dilakukan adalah menganalisis kebutuhan-kebutuhan apa saja yang perlu ada di dalam Aplikasi tersebut. Dimana Analisa kebutuhan untuk penelitian ini meliputi :

1. Wawancara

Teknik wawancara dilakukan dengan melakukan pengajuan pertanyaan secara langsung pada orang-orang yang berkaitan atau pemangku kebutuhan dari Aplikasi yang dibuat. Wawancara dilakukan guna mengetahui apa saja hal-hal yang dibutuhkan diaplikasi yang akan dikembangkan oleh penulis. Pemangku kebutuhan yang terlibat antara lain warga Blitar yang membutuhkan aplikasi ini guna kepentingan pariwisata. Wawancara akan



dilakukan kepada 2 jenis narasumber yaitu kepada Expert yang dimana dalam kasus ini penulis akan mewawancarai serta berdiskusi dengan orang yang terbiasa atau sering berwisata mengunjungi beberapa tempat wisata dalam sekali rute perjalanan hanya dengan menggunakan aplikasi Google Map dan juga kepada orang awam atau masyarakat secara umum.

2. Riset Pustaka

Riset pustaka dilakukan dengan cara menghimpun informasi yang relevan dengan permasalahan yang ada. Informasi dapat diperoleh dari sumber-sumber tertulis baik tercetak maupun elektronik. Riset pustaka dibutuhkan untuk mencari maupun mengkonfirmasi data yang kurang jelas kebenarannya. Riset pustaka bertujuan untuk membantu dalam analisis kebutuhan dari proses pengerjaan skripsi.

3.2 Membuat Prototype

Tahap membuat prototype pengembang membuat prototype dari sistem yang telah dijelaskan oleh pengguna atau pemilik sistem. Membangun prototyping dengan membuat perancangan sementara yang berpusat pada penyajian kepada pelanggan (misalnya dengan membuat input dan contoh outputnya).

3.3 Menyesuaikan Prototype Dengan Keinginan User

Menyesuaikan prototype dengan keinginan user pengembang menanyakan kepada pengguna atau pemilik sistem tentang prototype yang sudah dibuat, apakah sesuai atau tidak dengan kebutuhan sistem. Membangun prototyping dengan membuat perancangan sementara yang berpusat pada penyajian kepada pelanggan (misalnya dengan membuat input dan contoh outputnya).

3.4 Menggunakan Prototype

Menggunakan *prototype* sistem mulai dikembangkan dengan *prototype* yang sudah dibuat. Dalam tahap ini prototyping yang sudah disepakati diterjemahkan ke dalam bahasa pemrograman yang sesuai. Prototype yang sudah siap digunakan akan menjadi acuan pada tahap perancangan dan implementasi.

3.5 Perancangan Aplikasi

Pada tahapan ini peneliti merancang hal-hal yang diperlukan untuk pengembangan aplikasi. Pada bagian perancangan penelitian ini menggunakan *activity diagram*, *sequence diagram*, *class diagram* dan perancangan basis data



3.6 Implementasi

Proses selanjutnya dilakukan implementasi pada pengembangan aplikasi dengan menerapkan rancangan sistem. Implementasi Aplikasi Perangkat Bergerak Berbasis Lokasi untuk Perencanaan Perjalanan Wisata Blitar Berdasarkan Rute Terpendek terdiri dari implementasi basis data, implementasi kode program, implementasi antar muka dan implementasi Penyelesaian *Traveling Salesman Problem*.

3.7 Pengujian Aplikasi

Pada bagian ini diperlihatkan hasil pengujian pada aplikasi. Jika sudah dilakukan perbaikan maka akan diuji lagi apakah kesalahan-kesalahan tersebut sudah diselesaikan seperti semestinya. Pada penelitian ini menggunakan pengujian validasi untuk menguji kebutuhan fungsional pada aplikasi dan pengujian usabilitas dengan menggunakan kuisioner *SUS*.

3.8 Kesimpulan

Pada bagian ini penulis menuliskan kesimpulan dan saran yang supaya untuk kedepannya penelitian ini dapat dijadikan referensi untuk penelitian dengan tajuk yang serupa dan dapat bermanfaat untuk pihak yang membutuhkan untuk kedepannya.



BAB 4 ANALISIS KEBUTUHAN

Menjelaskan ulasan kebutuhan untuk pengembangan aplikasi perangkat bergerak berbasis lokasi untuk perencanaan perjalanan wisata blitar berdasarkan rute terpendek.

4.1 Gambaran Umum Aplikasi

Aplikasi ini memberikan informasi untuk menemukan jalur terpendek atau tercepat di peta berdasarkan tempat yang telah kita pilih, menyelesaikan permasalahan berdasarkan Traveling Salesman Problem. Aplikasi ini menggunakan 3 layanan API dari Google yaitu :

- Google Places web service
- Google Distance Matrix web service
- Google Directions web service

Komunikasi dengan semua layanan API tersebut didasarkan pada protokol HTTP REST. Aplikasi ini bekerja dengan mendapatkan file JSON sebagai respons, yang diuraikan untuk mendapatkan semua data yang diperlukan.

4.2 Identifikasi Aktor

Aktor pada penelitian ini ditunjukkan pada tabel 4.1 dan tabel 4.2 untuk iterasi 2.

Tabel 4.1 Identifikasi Aktor Iterasi 1

Aktor	Deskripsi
Wisatawan Kota Blitar	Wisatawan kota blitar dapat menentukan radius dalam hitungan meter, Ini akan memungkinkan aktor untuk mencari tempat hanya pada radius yang ditentukan dari lokasi mereka saat ini. Kemudian Aktor dapat memilih pilihan Rencanakan Rute dimana aktor harus membuat daftar tempat yang akan dikunjungi, rekomendasi rute dengan jarak tempuh

terpendek akan didasarkan pada tempat-tempat ini. Kemudian aktor juga dapat menemukan informasi lebih lanjut tentang rute yang ditemukan, misaknya jalur terpendek, yang telah ditemukan atau jalur terpanjang yang mungkin. Ada juga informasi tentang urutan tempat-tempat yang dipilih harus dikunjungi sehingga rute akan optimal. Di bagian ini, ada juga matriks jarak atau matriks waktu dari mana rute terbaik ditemukan. Matriks ini dibuat menggunakan layanan web Google Distance Matrix.

Tabel 4.2 Identifikasi Aktor Iterasi 2

Aktor	Deskripsi
Wisatawan Kota Blitar	wisatawan kota blitar dapat menentukan radius dalam hitungan meter, Ini akan memungkinkan aktor untuk mencari tempat hanya pada radius yang ditentukan dari lokasi mereka saat ini. Kemudian Aktor dapat memilih pilihan Rencanakan Rute dimana aktor harus membuat daftar tempat yang akan dikunjungi, rekomendasi rute dengan jarak tempuh terpendek akan didasarkan pada tempat-tempat ini. Kemudian aktor juga dapat menemukan informasi lebih lanjut tentang rute yang ditemukan, misaknya jalur terpendek, yang telah ditemukan atau jalur terpanjang yang mungkin. Ada juga informasi tentang urutan tempat-tempat yang dipilih harus dikunjungi sehingga rute akan optimal.. Data tempat wisata ini akan otomatis ditambahkan ke database sqlite aplikasi dan dapat ditampilkan pada daftar tempat yang dapat dikunjungi aktor dalam aplikasi ini.

Terdapat penambahan kebutuhan berupa menambah data tempat wisata berdasarkan data nama tempat wisata, latitude, longitude & deskripsi tempat wisata yang akan otomatis ditambahkan ke daftar tempat yang dapat dikunjungi aktor dalam aplikasi ini.

4.3 Kebutuhan Fungsional Sistem

Kebutuhan Fungsional dapat diartikan sebagai layanan atau proses yang dilakukan oleh aplikasi. Dalam Penelitian ini setiap Kebutuhan Fungsional diberi kode KF X untuk kebutuhan Fungsional dan KNF X untuk kebutuhan



non fungsional dimana X menunjukkan nomor dari definisi kebutuhan tersebut.

4.3.1 Iterasi ke-1

Terdapat 3 kebutuhan fungsional. Merencanakan rute, menampilkan rekomendasi rute perjalanan terpendek berdasarkan tempat wisata yang dipilih, dan Melihat Navigasi dari Lokasi saat ini ke Lokasi Tempat Wisata yang dipilih. Terdapat 4 Kebutuhan Fungsional aplikasi pada iterasi 2 dengan pengguna Wisatawan Kota Blitar pada tabel 4.3, yaitu merencanakan rute, menampilkan rekomendasi rute perjalanan terpendek berdasarkan tempat wisata yang dipilih, Melihat Navigasi dari Lokasi saat ini ke Lokasi Tempat Wisata yang dipilih, dan menambah Lokasi Tempat Wisata.

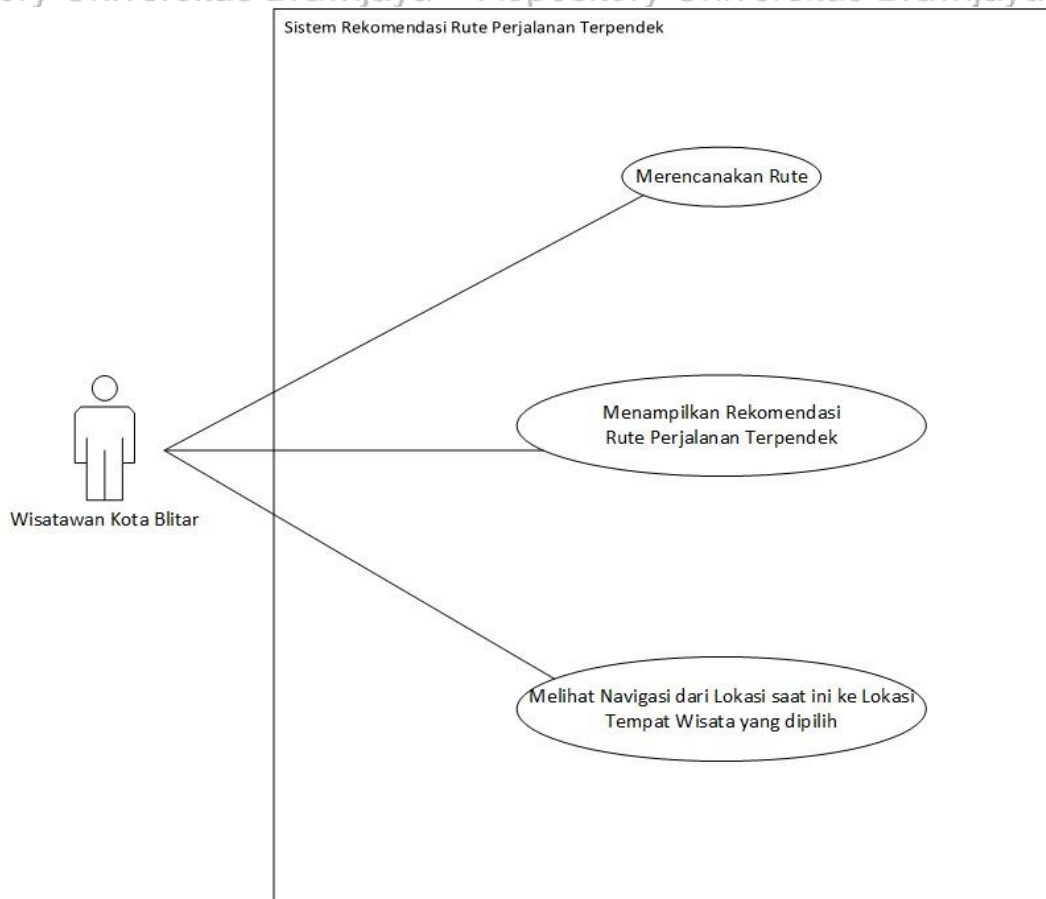
Tabel 4.3 Kebutuhan Fungsional Sistem Iterasi 1

Kode Fungsi	Nama Fungsi	Deskripsi
KF_01	Merencanakan Rute	Sistem harus dapat menerima inputan actor untuk menentukan radius lokasi tempat wisata yang dapat dipilih untuk dikunjungi actor dari lokasi actor saat ini dan Sistem harus dapat menampilkan tempat wisata mana saja yang akan dikunjungi actor.
KF_02	Menampilkan Rekomendasi Rute Perjalanan Terpendek berdasarkan tempat wisata yang dipilih	Sistem harus dapat menampilkan rekomendasi rute perjalanan terpendek berdasarkan data tempat wisata yang telah dipilih actor
KF_03	Melihat Navigasi dari Lokasi saat ini ke Lokasi Tempat Wisata yang dipilih	Sistem harus dapat menampilkan rute perjalanan dari marker yang dipilih actor ke lokasi terkini aktor menggunakan bantuan Google Maps.



4.3.3 USE CASE DIAGRAM ITERASI 1

Diagram Use Case mempunyai fungsi melihat hubungan antara aktor dengan use case nya. Aplikasi ini mempunyai aktor berupa Wisatawan Kota Blitar dengan 3 use case pada gambar 4.1 untuk use case diagram iterasi 1.



Gambar 4.1 Use Case Diagram Iterasi 1

Use case diagram iterasi 1 memiliki 3 use case, yaitu merencanakan rute, menampilkan rekomendasi rute perjalanan terpendek dan Melihat Navigasi dari Lokasi saat ini ke Lokasi Tempat Wisata yang dipilih.

4.3.4 Use Case Scenario Iterasi 1

Use case scenario menjelaskan tahapan scenario yang antara aktor dan aplikasi. Terdapat 2 iterasi yang dilakukan pada skripsi ini. Iterasi 1 memiliki 3 use case scenario yang terdapat pada tabel 4.7-4.9 dan iterasi 2 memiliki 1 use case scenario yang terdapat pada tabel 4.10.



4.3.4.1 MERENCANAKAN RUTE ITERASI 1

Tabel 4.5 merupakan penjelasan dari use case scenario merencanakan rute iterasi ke-1.

Tabel 4.5 Use Case Scenario Merencanakan Rute Iterasi 1

Nama Usecase	Merencanakan Rute
Kode	KF_01
Objective	Untuk menentukan radius lokasi tempat wisata yang akan dikunjungi dan menentukan tempat wisata mana saja yang akan dikunjungi
Actor	Wisatawan Kota Blitar
Pre - Condition	Actor telah berada di halaman utama aplikasi.
Main Flow	1. Actor menekan tombol Plan Route pada Halaman Utama Aplikasi 2. Sistem berada di halaman Activity find place 3. Actor memasukkan radius jarak tempat wisata yang bisa ditemukan dari lokasi saat ini, kemudian memilih tempat wisata mana saja yang akan dikunjungi termasuk memilih lokasi saat ini sebagai Starting point 4. Sistem akan beralih ke halaman Selected Place Activity setelah actor selesai menginput semua data tempat wisata yang akan dikunjungi.
Alternatif Flow	
Post-Condition	Actor berhasil menginputkan tempat-tempat wisata blitar yang akan dikunjungi



4.3.4.2 MENAMPILKAN REKOMENDASI RUTE PERJALANAN TERPENDEK ITERASI 1

Tabel 4.6 merupakan penjelasan dari use case scenario merencanakan rute perjalanan terpendek iterasi 1.

Tabel 4.6 Use Case Scenario Menampilkan Rekomendasi Rute Perjalanan Terpendek Iterasi 1

Nama Usecase	Menampilkan Rekomendasi Rute Perjalanan Terpendek
Kode	KF_02
Objective	Untuk menampilkan rekomendasi rute perjalanan terpendek berdasarkan data tempat-tempat wisata yang telah dipilih oleh actor
Actor	Wisatawan Kota Blitar
Pre - Condition	Actor berhasil menginputkan tempat-tempat wisata blitar yang akan dikunjungi dan berada di halaman Selected Place Activity.
Main Flow	1. Actor menekan tombol Find Road pada halaman Selected Place Activity 2. Sistem akan mengolah dari data-data tempat wisata yang telah dimasukkan oleh actor lalu sistem akan beralih ke halaman activity map dan memberikan informasi rekomendasi rute dengan jalur terpendek yang telah ditemukan dan juga informasi tentang urutan tempat-tempat yang direkomendasikan dikunjungi terlebih dahulu sehingga rute perjalanan akan optimal. Di bagian ini, ada juga matriks jarak atau matriks waktu dari mana rute terbaik ditemukan. Matriks ini dibuat menggunakan layanan web Google Distance Matrix.
Alternatif Flow	-
Post-Condition	Actor berhasil mendapatkan rekomendasi rute perjalanan terpendek berdasarkan data tempat-tempat wisata yang telah dipilih.

4.3.4.3 Menampilkan Rekomendasi Rute Perjalanan Terpendek Iterasi 1

Tabel 4.7 merupakan penjelasan dari use case scenario Melihat Navigasi dari Lokasi saat ini ke Lokasi Tempat Wisata yang dipilih iterasi 1.

Tabel 4.7 Use Case Scenario Melihat Navigasi dari Lokasi saat ini ke Lokasi Tempat Wisata yang dipilih Iterasi 1

Nama Usecase	Melihat Navigasi dari Lokasi saat ini ke Lokasi Tempat Wisata yang dipilih
Kode	KF_03
Objective	Untuk menampilkan rute dari marker yang dipilih ke lokasi terkini aktor menggunakan bantuan Google Maps.
Actor	Wisatawan Kota Blitar
Pre - Condition	Actor berhasil mendapatkan rekomendasi rute perjalanan terpendek berdasarkan data tempat-tempat wisata yang telah dipilih dan berada di activity map.
Main Flow	1. Actor menekan Marker dari sebuah tempat di Activity Map 2. Sistem menampilkan tombol navigasi di bagian kanan bawah peta. 3. Actor menekan tombol navigasi di bagian kanan bawah peta. 4. Sistem menampilkan rute navigasi dari marker yang dipilih actor ke lokasi actor saat ini menggunakan bantuan aplikasi Google Maps.
Alternatif Flow	-
Post-Condition	Sistem menampilkan rute navigasi dari marker yang dipilih ke lokasi actor saat ini.

4.3.4 PROTOTYPE ITERASI 1

Bagian ini menunjukkan prototype atau purwarupa desain iterasi 1 dari aplikasi. Tampilan Prototype Iterasi 1 dapat dilihat pada gambar 4.2 sampai 4.5

a. Tampilan Prototype Halaman Utama Aplikasi Iterasi 1

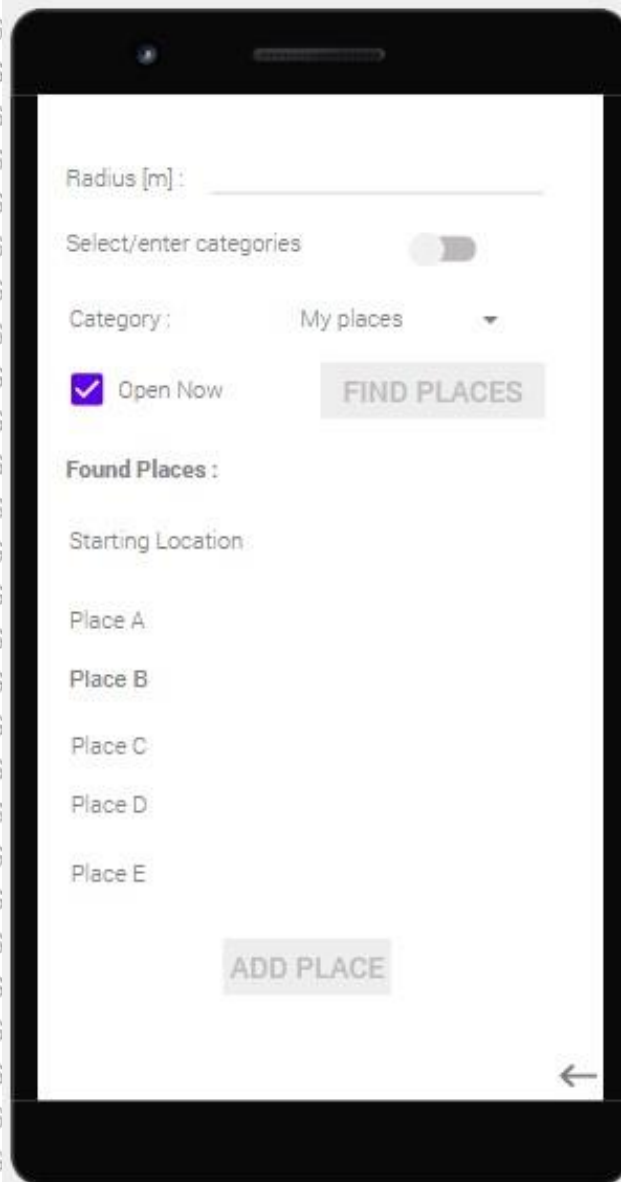
Gambar 4.2 menampilkan tampilan prototype halaman utama aplikasi iterasi 1 yang terdapat button Plan Route untuk berpindah ke halaman Activity Find Place.



Gambar 4.2 Tampilan Prototype Halaman Utama Aplikasi Iterasi 1

b. Tampilan Prototype Halaman Find Place Activity Iterasi 1

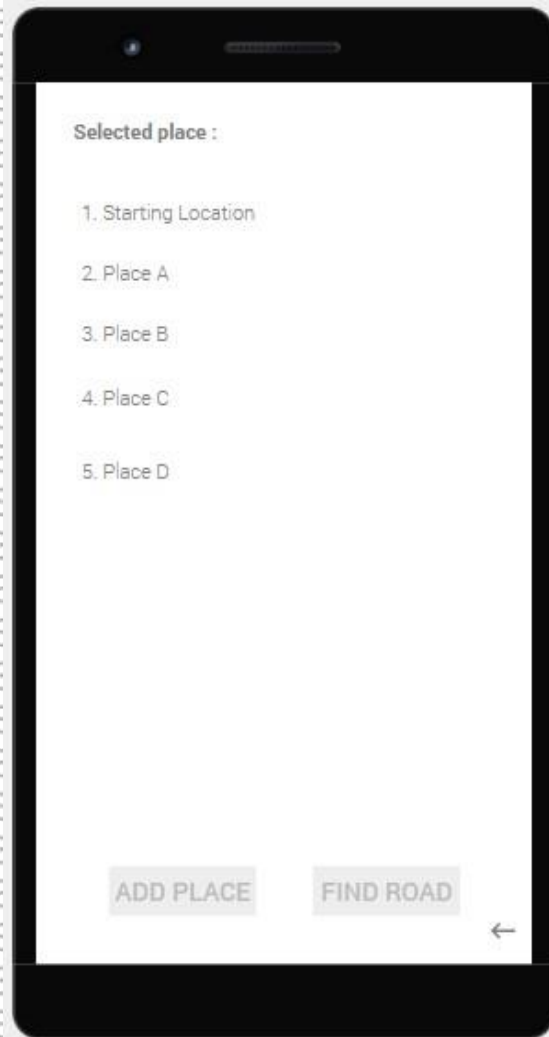
Gambar 4.3 menampilkan tampilan prototype halaman Find Place Activity iterasi 1. Halaman Find Place Activity digunakan untuk menemukan dan memilih beberapa lokasi tempat wisata yang akan dikunjungi actor dalam radius yang ditentukan oleh input actor dari lokasi mereka saat ini.



Gambar 4.3 Tampilan Prototype Halaman Find Place Activity Iterasi 1

c. Tampilan Prototype Halaman Selected Place Activity Iterasi 1

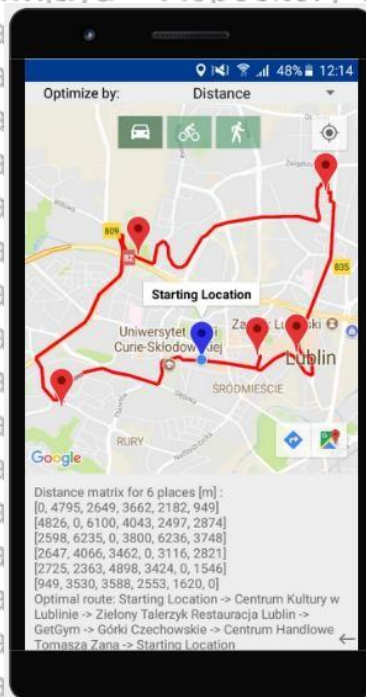
Gambar 4.4 menampilkan tampilan prototype halaman Selected Place Activity iterasi 1. Halaman Selected Place Activity digunakan untuk menampilkan daftar lokasi tempat wisata yang telah dipilih oleh actor untuk dikunjungi. Actor juga dapat menghapus salah satu list item tempat wisata sesuai kehendak mereka, terdapat 2 button pada halaman ini yaitu button add place untuk menambah daftar baru tempat wisata yang akan dikunjungi dan button find road untuk beralih ke halaman map activity dan mendapatkan rekomendasi rute perjalanan terpendek berdasarkan tempat yang telah dipilih.



Gambar 4.4 Tampilan Prototype Halaman Selected Place Activity Iterasi 1

d. Tampilan Prototype Halaman Map Activity Iterasi 1

Gambar 4.5 menampilkan tampilan prototype halaman Map Activity iterasi 1. Halaman Map Activity digunakan untuk menampilkan rekomendasi rute terpendek berdasarkan daftar lokasi tempat wisata yang telah dipilih oleh actor, pada halaman ini terdapat marker-marker yang merepresentasikan lokasi tiap tempat wisata dan polyline yang menghubungkan marker tersebut yang merepresentasikan rute perjalanan terpendek dari daftar lokasi tempat wisata tersebut. Pada Halaman ini juga memberikan informasi rekomendasi rute dengan jalur terpendek yang telah ditemukan dan juga informasi tentang urutan tempat-tempat yang direkomendasikan dikunjungi terlebih dahulu sehingga rute perjalanan akan optimal. Di bagian ini, ada juga matriks jarak atau matriks waktu dari mana rute terbaik ditemukan. Matriks ini dibuat menggunakan layanan web Google Distance Matrix.



Gambar 4.5 Tampilan Prototype Halaman Map Activity Iterasi 1

4.4 Iterasi ke-2

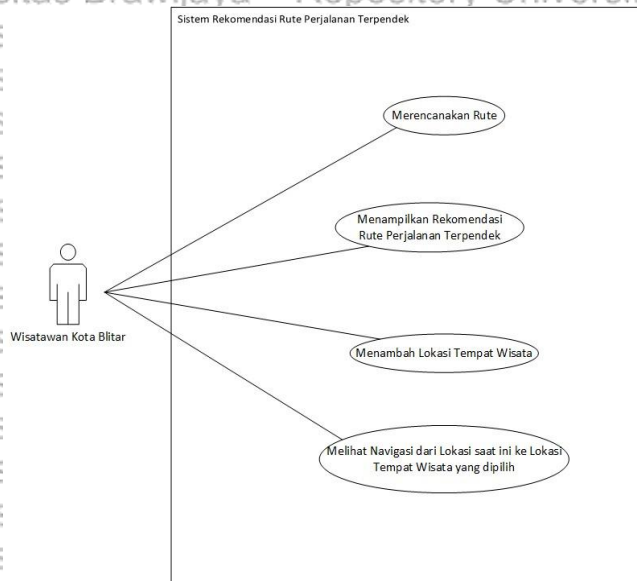
Terdapat penambahan Kebutuhan Fungsional aplikasi pada iterasi 2 dengan aktor Wisatawan Kota Blitar pada tabel 4.8 yaitu menambah lokasi tempat wisata.

Tabel 4.8 Kebutuhan Fungsional Sistem Iterasi 2

Kode Fungsi	Nama Fungsi	Deskripsi
KF_04	Menambah Lokasi Tempat Wisata	Sistem harus dapat menambahkan data tempat wisata baru berdasarkan data nama tempat wisata, latitude, longitude & deskripsi tempat wisata yang telah diinputkan actor. Data tersebut akan otomatis ditambahkan ke database sqlite aplikasi dan dapat ditampilkan pada daftar tempat yang dapat dikunjungi aktor dalam aplikasi ini.

4.4.1 USE CASE DIAGRAM ITERASI KE-2

Terdapat aktor berupa Wisatawan Kota Blitar dengan 4 use case pada gambar 4.2. untuk use case diagram iterasi 2.



Gambar 4.2 Use Case Diagram Iterasi 2

Use case diagram iterasi 2 memiliki 4 use case, yaitu merencanakan rute, menampilkan rekomendasi rute perjalanan terpendek dan Melihat Navigasi dari Lokasi saat ini ke Lokasi Tempat Wisata yang dipilih. Terdapat penambahan 1 use case pada iterasi 2, yaitu menambah lokasi tempat wisata

4.4.2 Use Case Scenario Iterasi ke-2

Iterasi 2 memiliki 1 use case scenario yang terdapat pada tabel 4.9

Tabel 4.9 Use Case Scenario Menambah Lokasi Tempat Wisata Iterasi 2

Nama Usecase	Menambah Lokasi Tempat Wisata
Kode	KF_04
Objective	Untuk menambahkan data tempat wisata baru berdasarkan data nama tempat wisata, latitude, longitude & deskripsi tempat wisata. Data tempat wisata ini akan otomatis ditambahkan ke database sqlite aplikasi dan dapat ditampilkan pada daftar tempat yang dapat dikunjungi aktor dalam aplikasi ini.
Actor	Wisatawan Kota Blitar

Pre - Condition	Actor telah berada di halaman utama aplikasi.
Main Flow	1. Actor menekan tombol Tambahkan Tempat Wisata 2. Aplikasi menampilkan halaman Addplace Activity 3. Actor memasukkan data nama tempat wisata, Latitude, Longitude & Deskripsi Tempat Wisata lalu menekan tombol add. 4. Aplikasi akan menyimpan input dalam database sqlite, lalu sistem akan beralih ke halaman Activity find place dan menampilkan data tempat yang baru saja diinputkan actor pada kategori My Places pada halaman Activity find place.
Alternatif Flow	-
Post-Condition	Actor berhasil menambahkan data tempat wisata baru dalam database sistem dan data tempat tersebut dapat ditampilkan di aplikasi.

Terdapat penambahan fungsional Menambah Lokasi Tempat Wisata, yaitu Untuk menambahkan data tempat wisata baru berdasarkan data nama tempat wisata, latitude, longitude & deskripsi tempat wisata. Data tempat wisata ini akan otomatis ditambahkan ke database sqlite aplikasi dan dapat ditampilkan pada daftar tempat yang dapat dikunjungi aktor dalam aplikasi ini. Penambahan ini dilakukan terkait hasil evaluasi dari seorang pemandu wisata dan satu pengguna umum yang terbiasa melakukan explore tempat wisata saat melakukan uji coba prototype 1 aplikasi ini dan berdiskusi, 2 responden tersebut sepakat menilai prototype 1 aplikasi ini perlu ditambah fungsional menambah lokasi tempat wisata oleh pengguna supaya mereka mampu mendapat informasi rekomendasi rute dengan jarak tempuh terpendek berdasarkan lokasi tempat-tempat wisata yang telah mereka tambahkan sendiri atau dengan kata lain lebih fleksibel dalam memilih tempat-tempat wisata yang akan dikunjungi.

4.4.3 Prototype Iterasi ke-2

Bagian ini membahas rancangan prototype pada iterasi 2. Tampilan Prototype Iterasi 1 dapat dilihat pada gambar 4.6 dan 4.7

a. Tampilan Prototype Halaman Utama Aplikasi Iterasi 2

Gambar 4.6 menampilkan tampilan prototype halaman utama aplikasi iterasi 2. Terdapat perubahan pada Halaman Utama Aplikasi pada Iterasi ke 2 yaitu penambahan button Add Your Own Place untuk berpindah ke halaman Add Place Activity



Gambar 4.6 Tampilan Prototype Halaman Utama Aplikasi Iterasi 2

b. Tampilan Prototype Halaman Add Place Activity Iterasi 2

Gambar 4.7 menampilkan tampilan prototype halaman Add Place Activity Iterasi 2. Halaman Add Place Activity digunakan untuk menambahkan data tempat wisata baru berdasarkan data nama tempat wisata, latitude, longitude & deskripsi tempat wisata. Data tempat wisata ini akan otomatis ditambahkan ke database sqlite aplikasi dan dapat ditampilkan pada daftar tempat yang dapat dikunjungi aktor dalam aplikasi ini. Terdapat tombol Tambahkan tempat pada halaman utama aplikasi untuk menambahkan data yang diinputkan dan ditampilkan di halaman Find Place Activity.

Gambar 4.7 Tampilan Prototype Halaman Add Place Activity Iterasi 2



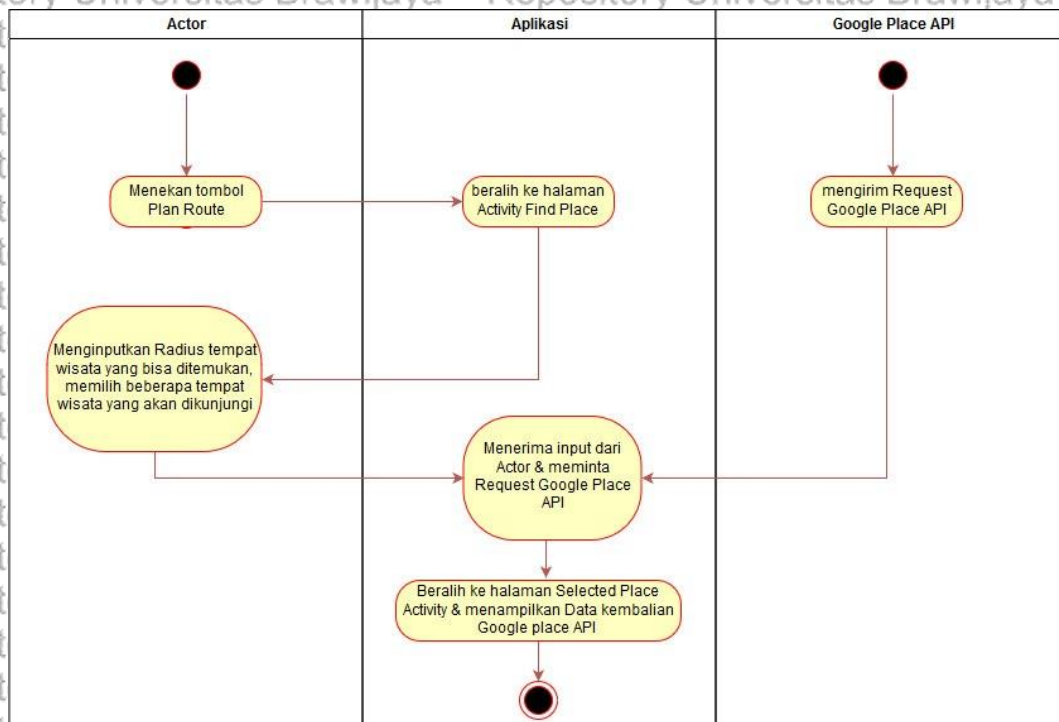
BAB 5 PERANCANGAN

5.1 Perancangan

Bab ini berisi tahap perancangan aplikasi perangkat bergerak berbasis lokasi untuk perencanaan perjalanan wisata kota blitar berdasarkan rute terpendek.

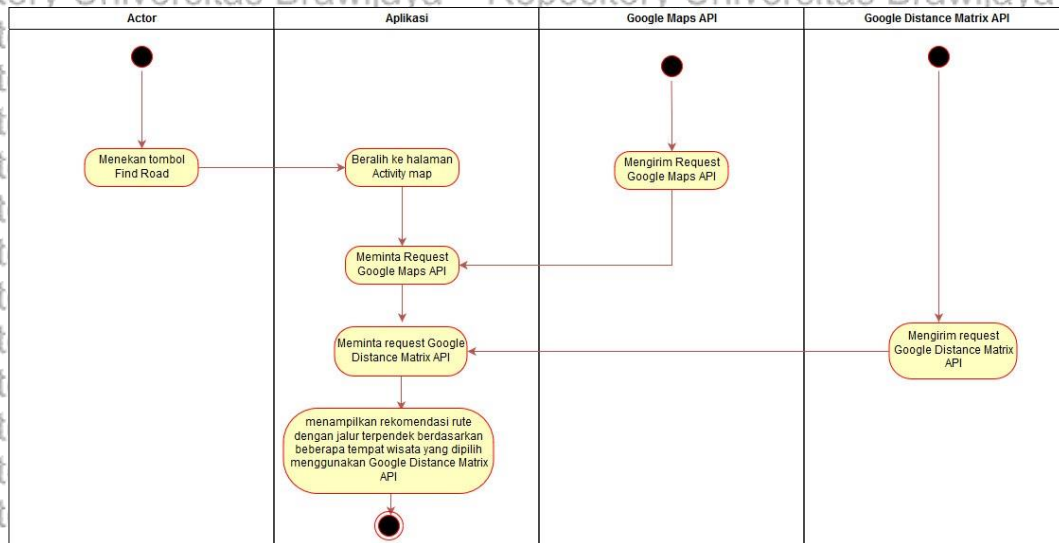
5.2 Activity Diagram

activity diagram dalam perancangan aplikasi dapat dilihat pada Gambar 5.1 sampai Gambar 5.4



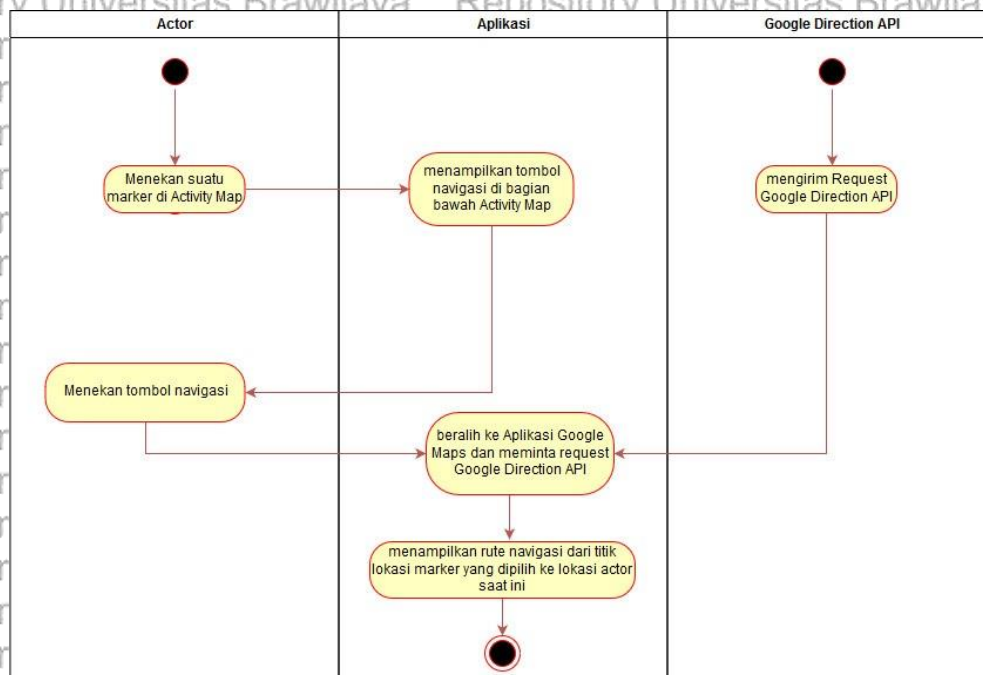
Gambar 5.1 Activity Diagram Use Case Merencanakan Rute

Gambar 5.1 menjelaskan tentang activity diagram use case merencanakan rute dimana alurnya adalah Actor menekan tombol Plan Route pada Halaman Utama Aplikasi kemudian Sistem beralih ke halaman Activity find place lalu Actor memasukkan radius jarak tempat wisata yang bisa ditemukan dari lokasi saat ini, kemudian memilih tempat wisata mana saja yang akan dikunjungi menggunakan request Google Place API, termasuk memilih lokasi saat ini sebagai Starting point kemudian Sistem akan beralih ke halaman Selected Place Activity setelah actor selesai menginput semua data tempat wisata yang akan dikunjungi.



Gambar 5.2 Activity Diagram Use Case Menampilkan Rekomendasi Rute Perjalanan Terpendek

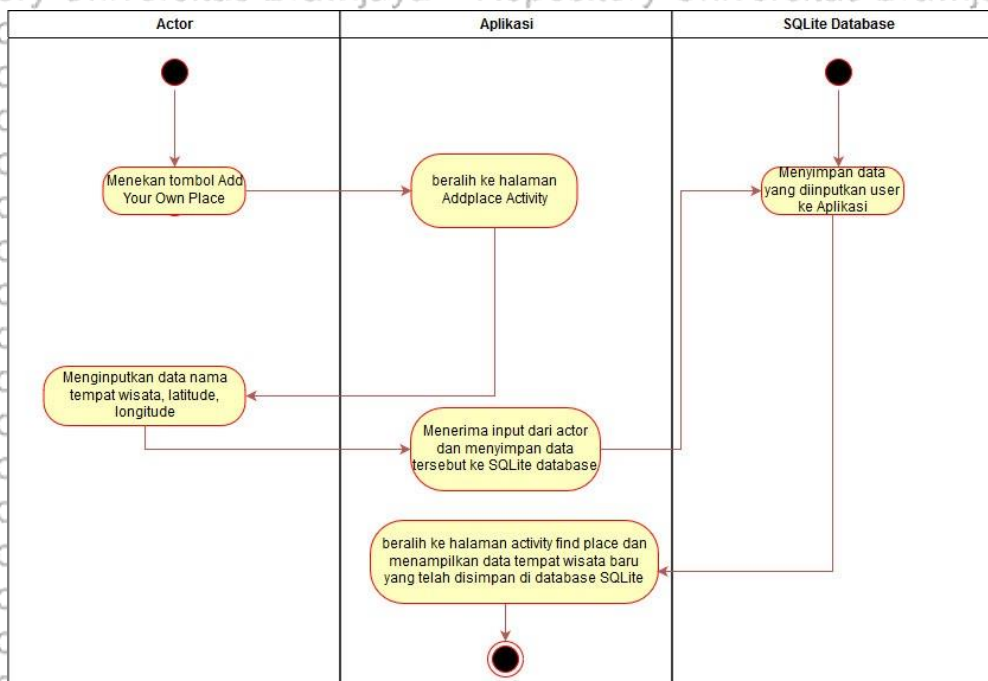
Gambar 5.2 menjelaskan tentang activity diagram use case menampilkan rekomendasi rute perjalanan terpendek dimana alurnya adalah Actor menekan tombol Find Road pada halaman Selected Place Activity kemudian Sistem akan mengolah dari data-data tempat wisata yang telah dimasukkan oleh actor menggunakan layanan Google Distance Matrix API lalu sistem akan beralih ke halaman activity map dan memberikan informasi rekomendasi rute dengan jalur terpendek yang telah ditemukan dan juga informasi tentang urutan tempat-tempat yang direkomendasikan dikunjungi terlebih dahulu sehingga rute perjalanan akan optimal.



Gambar 5.3 Activity Diagram Use Case Melihat Navigasi dari Lokasi saat ini ke Lokasi Tempat Wisata yang dipilih



Gambar 5.3 menjelaskan tentang activity diagram use case Melihat Navigasi dari Lokasi saat ini ke Lokasi Tempat Wisata yang dipilih dimana alurnya adalah Actor menekan Marker dari sebuah tempat di Activity Map kemudian Sistem menampilkan tombol navigasi di bagian kanan bawah peta lalu Actor menekan tombol navigasi di bagian kanan bawah peta berlanjut Sistem menampilkan rute navigasi dari marker yang dipilih actor ke lokasi actor saat ini menggunakan bantuan layanan Google Direction API dan aplikasi Google Maps.



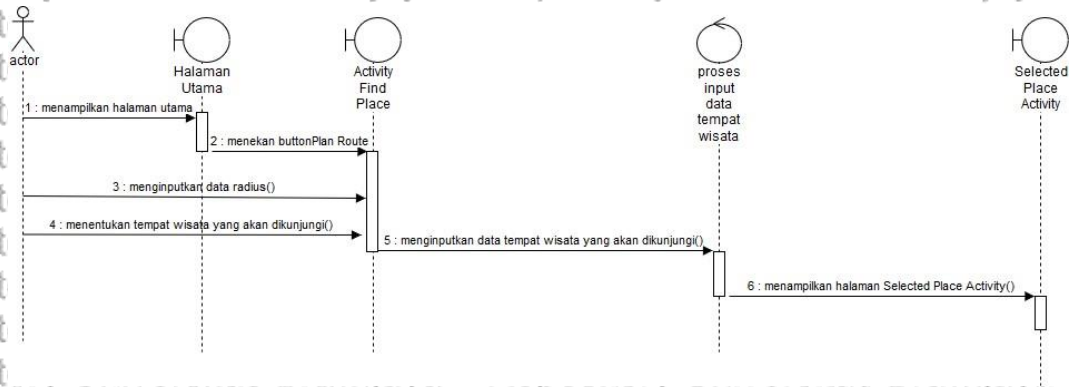
Gambar 5.4 Activity Diagram Use Case Menambah Lokasi Tempat Wisata

Gambar 5.4 menjelaskan tentang activity diagram use case menambah lokasi tempat wisata dimana alurnya adalah Actor menekan tombol Add Your Own Place kemudian Sistem menampilkan halaman Addplace Activity lalu Actor memasukkan data nama tempat wisata, Latitude, Longitude & Deskripsi Tempat Wisata lalu menekan tombol add berlanjut Database akan menyimpan inputan actor dalam database sqlite, lalu sistem akan beralih ke halaman Activity find place dan menampilkan data tempat wisata baru yang diinputkan actor dan tersimpan di SQLite database.



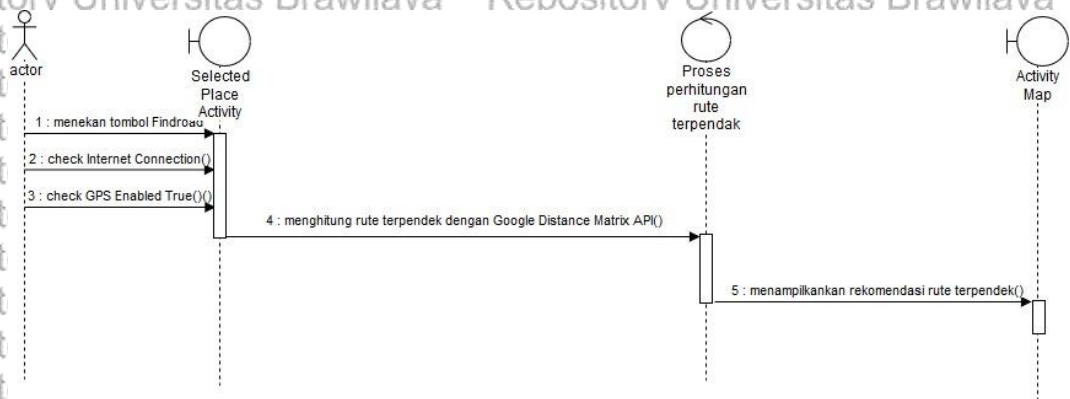
5.3 SEQUENCE DIAGRAM

Dalam Skripsi ini *sequence diagram* terdapat di Gambar 5.5 sampai Gambar 5.8.



Gambar 5.5 Sequence Diagram Merencanakan Route

Gambar 5.5 membahas mengenai *sequence diagram* merencanakan rute. Actor yang telah berada di halaman utama aplikasi menekan tombol *Plan Route* pada Halaman Utama Aplikasi kemudian Sistem beralih ke halaman *Activity find place* lalu Actor memasukkan radius jarak tempat wisata yang bisa ditemukan dari lokasi saat ini, kemudian memilih tempat wisata mana saja yang akan dikunjungi menggunakan request *Google Place API*, termasuk memilih lokasi saat ini sebagai *Starting point* kemudian Sistem akan beralih ke halaman *Selected Place Activity* setelah actor selesai menginput semua data tempat wisata yang akan dikunjungi.

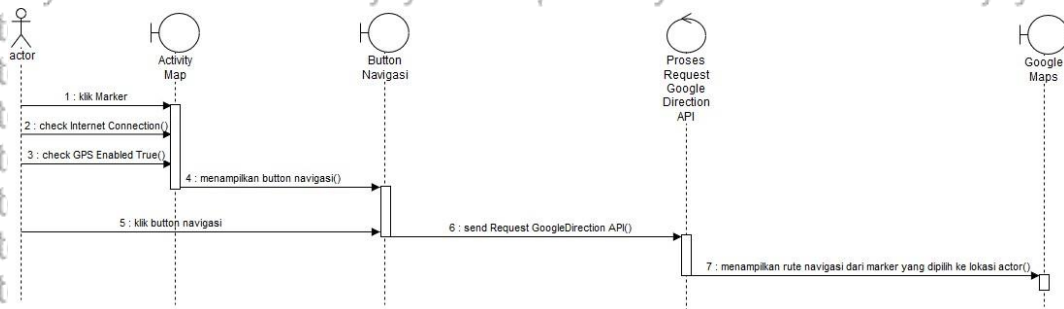


Gambar 5.6 Menampilkan Rekomendasi Rute Perjalanan Terpendek

Gambar 5.6 menjelaskan tentang use case diagram menampilkan rekomendasi rute perjalanan terpendek dimana alurnya adalah Actor menekan tombol *Find Road* pada halaman *Selected Place Activity* kemudian Sistem akan mengolah dari data-data tempat wisata yang telah dimasukkan oleh actor menggunakan layanan *Google Distance Matrix API* lalu sistem akan beralih ke halaman

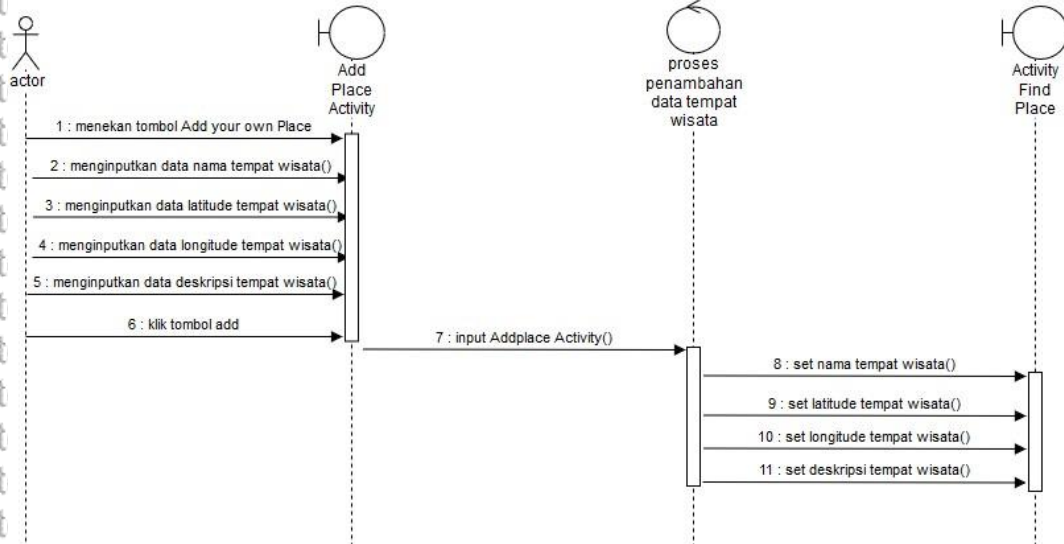


activity map dan memberikan informasi rekomendasi rute dengan jalur terpendek sebagai outputnya.



Gambar 5.7 Sequence Diagram Melihat Navigasi dari Lokasi saat ini ke Lokasi Tempat Wisata yang dipilih

Gambar 5.7 menjelaskan tentang sequence diagram Melihat Navigasi dari Lokasi saat ini ke Lokasi Tempat Wisata yang dipilih dimana alurnya adalah Actor menekan *Marker* dari sebuah tempat di *Activity Map* kemudian Sistem menampilkan tombol navigasi di bagian kanan bawah *Activity Map* lalu Actor menekan tombol navigasi di bagian kanan bawah peta berlanjut Sistem menampilkan rute navigasi dari marker yang dipilih actor ke lokasi actor saat ini menggunakan bantuan layanan *Google Direction API* dan aplikasi *Google Maps*.



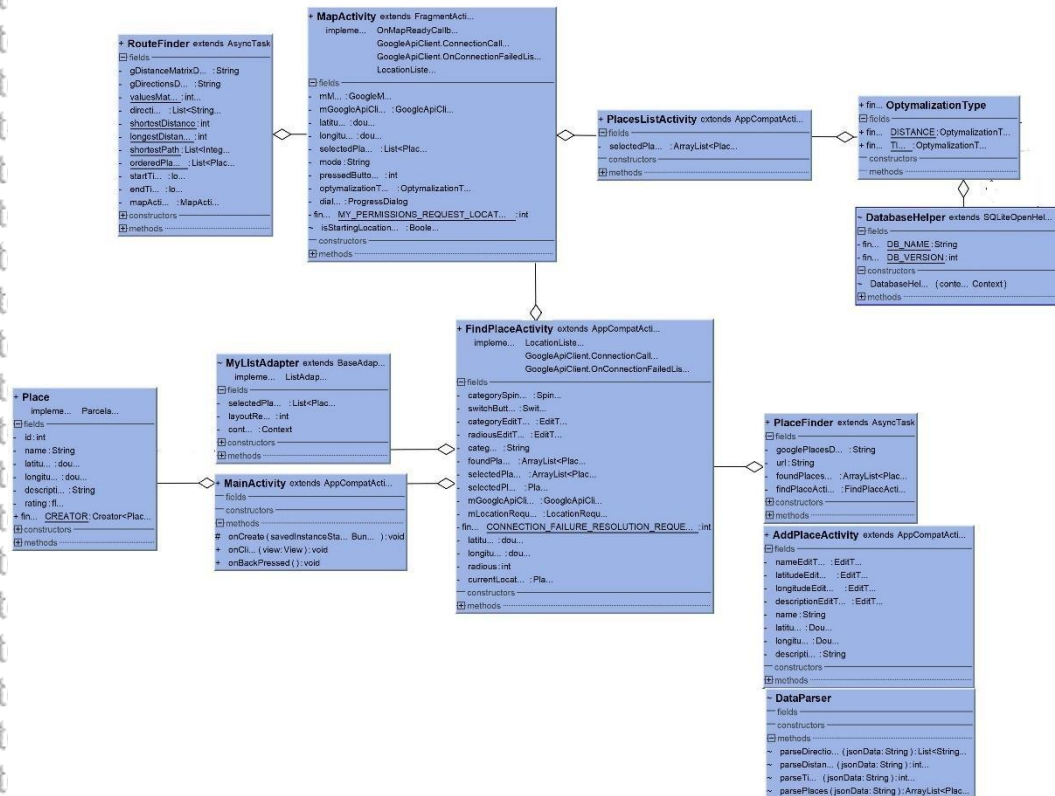
Gambar 5.8 Sequence Diagram Menambah Lokasi Tempat Wisata

Gambar 5.8 menjelaskan tentang *sequence diagram* menambah lokasi tempat wisata dimana alurnya adalah Actor menekan tombol *Add Your Own Place* kemudian Sistem menampilkan halaman *Addplace Activity* lalu Actor memasukkan data nama tempat wisata, *Latitude*, *Longitude* & Deskripsi Tempat Wisata lalu menekan tombol *add* berlanjut *Database* akan menyimpan

inputan ke database sqlite, lalu sistem akan beralih ke halaman *Activity find place* dan menampilkan data tempat yang baru saja diinputkan actor.

5.4 Class Diagram

Bagian ini menjelaskan bagaimana definisi hubungan antar class dalam rancangan aplikasi. Gambar 5.9 merupakan class diagram dari aplikasi



Gambar 5.9 Class Diagram Aplikasi Perangkat Bergerak Berbasis Lokasi Untuk Perencanaan Perjalanan Wisata Blitar Berdasarkan Rute Terpendek

Pada bagian class diagram pada aplikasi ini berisi kelas-kelas yang saling berhubungan yang membuat aplikasi ini dapat bekerja sebagaimana mestinya. Pada class inti aplikasi ini terdapat pada class *RouteFinder.java* yang berisi perhitungan untuk *brute force* untuk penyelesaian *traveling salesman problem* lalu pada kelas *findplaceActiviy.java* untuk menemukan lokasi tempat wisata yang didapat dari data *Google Place API*. Kemudian untuk penyimpanan data terletak pada kelas *place* untuk mengelola basis data pada aplikasi. Pada kelas *MapActivity.java* digunakan untuk menampilkan peta pada aplikasi menggunakan bantuan *Google.Map API*.



5.5 PERANCANGAN BASIS DATA

Bagian ini menjelaskan rancangan basis data Aplikasi Perangkat Bergerak Berbasis Lokasi Untuk Perencanaan Perjalanan Wisata Blitar Berdasarkan Rute Terpendek yang mana data tersebut telah diperoleh dari hasil uji coba prototyping 1 dan diskusi peneliti kepada pengguna aplikasi. Gambar 5.10 menunjukkan Perancangan basis data dari aplikasi ini yang menggunakan SQLite Database:

Place		
 Id	Integer(10)	
 name	varchar(255)	N
 latitude	varchar(20)	N
 longitude	varchar(20)	N
 description	varchar(255)	N

Gambar 5.10 Perancangan Basis Data

Gambar 5.10 menampilkan rancangan dari *database* yang digunakan dalam membangun sistem Aplikasi Perangkat Bergerak Berbasis Lokasi Untuk Perencanaan Perjalanan Wisata Blitar Berdasarkan Rute Terpendek. Dalam rancangan diatas terdapat 1 tabel, yaitu tabel Place. Tabel Place digunakan untuk menyimpan dan menampilkan semua data tempat wisata baru yang telah diinputkan actor yang berupa data nama tempat wisata, latitude tempat wisata, longitude tempat wisata dan deskripsi tempat wisata.



BAB 6 IMPLEMENTASI

Bagian ini berisi implementasi atau tahap pengerjaan dari rancangan yang telah dibuat oleh peneliti.

6.1 Spesifikasi Sistem

Bagian ini menjelaskan detail dari sistem aplikasi perencanaan perjalanan wisata blitar berdasarkan rute terpendek.

6.1.1 Spesifikasi Perangkat Keras

Bagian ini membahas bagaimana spesifikasi *hardware* dari laptop yang digunakan peneliti untuk mengimplementasikan rancangan sistem perencanaan perjalanan wisata blitar berdasarkan rute terpendek. Tabel 6.1 merupakan spesifikasi laptop yang digunakan peneliti.

Tabel 6.1 Spesifikasi Perangkat Keras Komputer

Komponen	Spesifikasi
Model sistem	TOSHIBA Satellite L745-1197U
Prosesor	Intel(R) Core(TM) Core i3, 2.2GHz
Memori	6.00 GB DDR3L
Kartu grafis	Intel HD Graphics

Tabel 6.2 merupakan spesifikasi Smartphone milik peneliti untuk menjalankan aplikasi

Tabel 6.2 Spesifikasi Perangkat Bergerak

Komponen	Spesifikasi
Model sistem	Samsung Galaxy J5
Prosesor	CPU Speed 1.2GHz. CPU Type Quad-Core.
Memori	RAM Size (GB) 2 GB
Kartu grafis	kartu grafis GPU Mali-T830MP2

6.1.2 SPESIFIKASI PERANGKAT LUNAK

Tabel 6.3 Spesifikasi Perangkat Lunak Komputer

Komponen	Spesifikasi
<i>Operating System</i>	Windows 10 Enterprise
<i>Programming Language</i>	Java, XML.
<i>Tools Development</i>	Android Studio, Sublime Text
<i>Tools Server dan DBMS</i>	SQLite
<i>Web Browser</i>	Mozilla Firefox

Berikut Spesifikasi Laptop peneliti ketika mengembangkan Aplikasi.
Tabel 6.3 merupakan spesifikasi perangkat lunak laptop peneliti.

Tabel 6.4 Spesifikasi Perangkat Lunak pada Perangkat Bergerak

Komponen	Spesifikasi
<i>Operating system</i>	Android OS 4.1 (Jelly Bean) atau lebih

Tabel 6.4 merupakan spesifikasi perangkat lunak system pada Smartphone milik peneliti dimana dalam kasus penelitian ini adalah Platform Android.

6.2 Batasan Implementasi

Batasan implementasi pada penelitian ini yaitu :

1. Untuk mengimplementasikan aplikasi perangkat bergerak dapat menggunakan android 4.4 (*KitKat*), Android 5.0 (*Lollipop*), dan Android 6.0 (*Marshmallow*).
2. Untuk mengimplementasikan nya menggunakan Bahasa pemrograman java dengan menggunakan IDE Android Studio.
3. Peneliti menggunakan layanan Google Place API untuk fitur Place Picker aplikasi ini.
4. Untuk mengimplementasikan fungsi navigasi dari suatu titik kordinat menggunakan layanan Google Directions API.
5. Untuk mengimplementasikan fungsi rekomendasi rute terpendek pada Aplikasi menggunakan layanan Google Distance Matrix API.
6. Untuk menyimpan data tempat wisata baru yang diinputkan user menggunakan SQLite Database.



6.3 IMPLEMENTASI BASIS DATA

Berikut Query atau Code dari implementasi basis data aplikasi ini

```
CREATE TABLE `Place` (
  `id` int(10) NOT NULL PRIMARY KEY,
  `name` varchar(255) NOT NULL,
  `latitude` varchar(20) NOT NULL,
  `longitude` varchar(20) NOT NULL,
  `description` varchar(255) NOT NULL,
)
```

Kode 6.1 Kode DDL SQLite Membuat Tabel Place

Kode 6.1 merupakan kode program DDL sebagai implementasi database Place. Tabel 6.5 merupakan struktur database yang diimplementasikan.

Tabel 6.5 Struktur Tabel Basis Data Place

No.	Nama Kolom	Tipe	Keterangan	Contoh
1.	id	Int(10)	Input data id	1,2,3,4, dst
2.	name	varchar(255)	Input data nama tempat wisata	Makam Bung Karno
3.	latitude	Double	input nilai latitude	-7.966574
4.	longitude	Double	input nilai longitude	112.618285
5.	description	Varchar(255)	Input data description tempat wisata.	Tempat wisata bersejarah kota Blitar.

Hasil Implementasi Database aplikasi yang menggunakan Database SQLite.

Gambar 6.1 implementasi database Place.

Place		
 id	integer(10)	
 name	varchar(255)	N
 latitude	varchar(20)	N
 longitude	varchar(20)	N
 description	varchar(255)	N

Gambar 6.1 Hasil Implementasi Basis Data



6.4 IMPLEMENTASI KODE PROGRAM

Menjelaskan bagaimana penulis menulis code atau proses coding untuk membangun aplikasi ini.

6.4.1 Implementasi Kode Program Aplikasi Perangkat Bergerak

Implementasi code saat coding diciptakan beberapa java class diantaranya *Activity_find_place.java*, *RouteFinder.java*, *activity_add_place.java* karena kelas-kelas tersebut merupakan implementasi berdasarkan Sequence Diagram Aplikasi Perangkat Bergerak Berbasis Lokasi Untuk Perencanaan Perjalanan Wisata Blitar Berdasarkan Rute Terpendek

```

232 private String getUrl(double latitude, double longitude, String keyword) {
233     CheckBox openNowCheckBox = findViewById(R.id.open_now_checkBox);
234     StringBuilder googlePlaceUrl = new StringBuilder("https://maps.googleapis.com/maps/api/place/nearbysearch/json?");
235     googlePlaceUrl.append("location=" + latitude + "," + longitude);
236     googlePlaceUrl.append("&radius=" + radius);
237     googlePlaceUrl.append("&keyword=" + keyword);
238     googlePlaceUrl.append("&language=pl");
239     googlePlaceUrl.append("&sensor=true");
240     if (openNowCheckBox.isChecked()) {
241         googlePlaceUrl.append("&opennow=true");
242     }
243     googlePlaceUrl.append("&key=" + "AIzaSyAD0Iq8FCv0xtj0gLyEQjXN7VN8I1QAe4");
244
245     Log.d("tag: FindPlaceActivity", "msg: url = " + googlePlaceUrl.toString());
246
247     return googlePlaceUrl.toString();
248 }

```

Kode 6.2 Kode Program Activity_find_place.java

Kode 6.2 diatas digunakan untuk menampilkan berbagai tempat-tempat dengan berbagai kategori yang ditampilkan melalui layanan Google Place API atau disebut dengan PlacePicker, tempat-tempat yang ditampilkan tersebut dapat dipilih oleh user dan akan disimpan dalam sebuah list bernama Place yang nanti akan diproses menggunakan Google Distance Matrix API untuk mendapatkan data rekomendasi rute terdekat.

```

134 static void solveTSP(int[][] valuesMatrix) {
135     shortestDistance = Integer.MAX_VALUE;
136     longestDistance = Integer.MIN_VALUE;
137     shortestPath = null;
138
139     int totalPlaces = valuesMatrix.length;
140     ArrayList<Integer> places = new ArrayList<>();
141     for(int i=0; i<totalPlaces; i++){
142         places.add(i);
143     }
144     int startPlace = places.get(0);
145     int currentDistance = 0;
146     bruteForceSearch(valuesMatrix, places, startPlace, currentDistance);
147 }
148 private static void bruteForceSearch(int[][] valuesMatrix, List<Integer> places,
149                                     int startPlace, int currentDistance) {
150     if (startPlace < places.size()-1) {
151         for (int i=startPlace; i < places.size(); i++) {
152             int tempCity = places.get(i);
153             places.set(i, places.get(startPlace));
154             places.set(startPlace, tempCity);
155             currentDistance = computeDistance(places, valuesMatrix);
156             bruteForceSearch(valuesMatrix, places, startPlace: startPlace+1,
157                             currentDistance);
158             tempCity = places.get(i);
159             places.set(i, places.get(startPlace));
160             places.set(startPlace, tempCity);
161         }
162     }

```

Kode 6.3 Kode Program RouteFinder.java



Kode 6.3 diatas digunakan untuk menampilkan hasil rekomendasi rute perjalanan terpendek berdasarkan tempat wisata yang dipilih oleh user dengan menggunakan layanan *Google Distance Matrix API* untuk dikalkulasi.

```

30 public boolean isEnteredDataValid(){
31     nameEditText = findViewById(R.id.name_edit_text);
32     latitudeEditText = findViewById(R.id.latitude_edit_text);
33     longitudeEditText = findViewById(R.id.longitude_edit_text);
34
35     return !(String.valueOf(nameEditText.getText()).isEmpty()
36             || String.valueOf(latitudeEditText.getText()).isEmpty()
37             || String.valueOf(longitudeEditText.getText()).isEmpty()
38             || (Double.valueOf(latitudeEditText.getText().toString()) > 90)
39             || (Double.valueOf(latitudeEditText.getText().toString()) < -90)
40             || (Double.valueOf(longitudeEditText.getText().toString()) > 180)
41             || (Double.valueOf(longitudeEditText.getText().toString()) < -180));
42 }
43
44 public void getValues(){
45     nameEditText = findViewById(R.id.name_edit_text);
46     latitudeEditText = findViewById(R.id.latitude_edit_text);
47     longitudeEditText = findViewById(R.id.longitude_edit_text);
48     descriptionEditText = findViewById(R.id.description_edit_text);
49
50     name = String.valueOf(nameEditText.getText());
51     latitude = Double.valueOf(latitudeEditText.getText().toString());
52     longitude = Double.valueOf(longitudeEditText.getText().toString());
53     if(String.valueOf(descriptionEditText.getText()).isEmpty()){
54         description = "";
55     }
56     else{
57         description = String.valueOf(descriptionEditText.getText());
58     }
59 }

```

Kode 6.4 Kode Program Activity_add_place.java

Kode 6.4 diatas digunakan untuk menambah lokasi tempat wisata yang ingin dikunjungi oleh user secara manual pada Aplikasi yang data-data tersebut akan diolah menggunakan Database SQLite, nama database dari sistem ini adalah database Place. Dimana nanti user akan menginputkan data nama tempat wisata, longitude tempat wisata, latitude tempat wisata dan deskripsi singkat tempat wisata.

6.5 Implementasi Antar Muka

Pada tahap menjelaskan implementasi antar muka untuk aplikasi ini. Implementasi antarmuka terdapat pada Gambar 6.2 sampai 6.6.

a. Halaman Utama Aplikasi

Halaman Utama digunakan untuk memilih opsi apakah actor ingin memilih opsi Rencanakan Rute atau Tambahkan Tempat. Gambar 6.2 menampilkan Laman Utama Aplikasi yang terdapat 2 button pada laman ini untuk mengakses 2 opsi tersebut.



Abidin Route Apps



Gambar 6.2 Halaman Utama Aplikasi

b. Halaman Find Place Activity

Halaman Find Place Activity digunakan untuk menemukan dan memilih beberapa lokasi tempat wisata yang akan dikunjungi actor dalam radius yang ditentukan oleh input actor dari lokasi mereka saat ini. Gambar 6.3 menunjukkan implementasi Halaman Find Place Activity



Radius [meter] :

Radius Tidak Boleh Kosong

Pilih / Masukkan Kategori Tempat

Kategori :

My places

☐ Parsing Data yang didapat

TEMUKAN TEMPAT

Tempat yang ditemukan :

Starting Location

Add this place if you want the current location to be included in the route planning

Alun Alun Blitar

Alun Alun Kota Blitar

Kampung Coklat

Kampung Wisata Coklat Blitar

Kebon Rojo

Taman Kota Kebon Rojo Blitar

Makam Bung Karno

Makam Presiden Bung Karno

TAMBAHKAN TEMPAT LAGI

Gambar 6.3 Halaman Find Place Activity

c. Halaman Selected Place Activity

Halaman Selected Place Activity digunakan untuk memberikan data daftar lokasi tempat wisata sesuai pilihan pengguna. Mereka juga dapat menghapus salah satu list item tempat wisata sesuai kehendak mereka, terdapat 2 tombol pada halaman ini yaitu tombol add place untuk menambah daftar baru tempat wisata yang akan dikunjungi dan button find road untuk beralih ke halaman map activity dan mendapatkan rekomendasi rute perjalanan terpendek berdasarkan tempat yang telah dipilih. Gambar 6.4 menunjukkan implementasi antar muka halaman Selected Place Activity

← Selected places :

Tempat yang telah anda pilih :

1. Starting Location

Add this place if you want the current location to be included in the route planning



2. Alun Alun Blitar

Alun Alun Kota Blitar



3. Kampung Coklat

Kampung Wisata Coklat Blitar



4. Kebon Rojo

Taman Kota Kebon Rojo Blitar



5. Makam Bung Karno

Makam Presiden Bung Karno



6. Sumber Udel

Kolam Renang Umum Kota Blitar



TAMBAHKAN TEMPAT LAGI



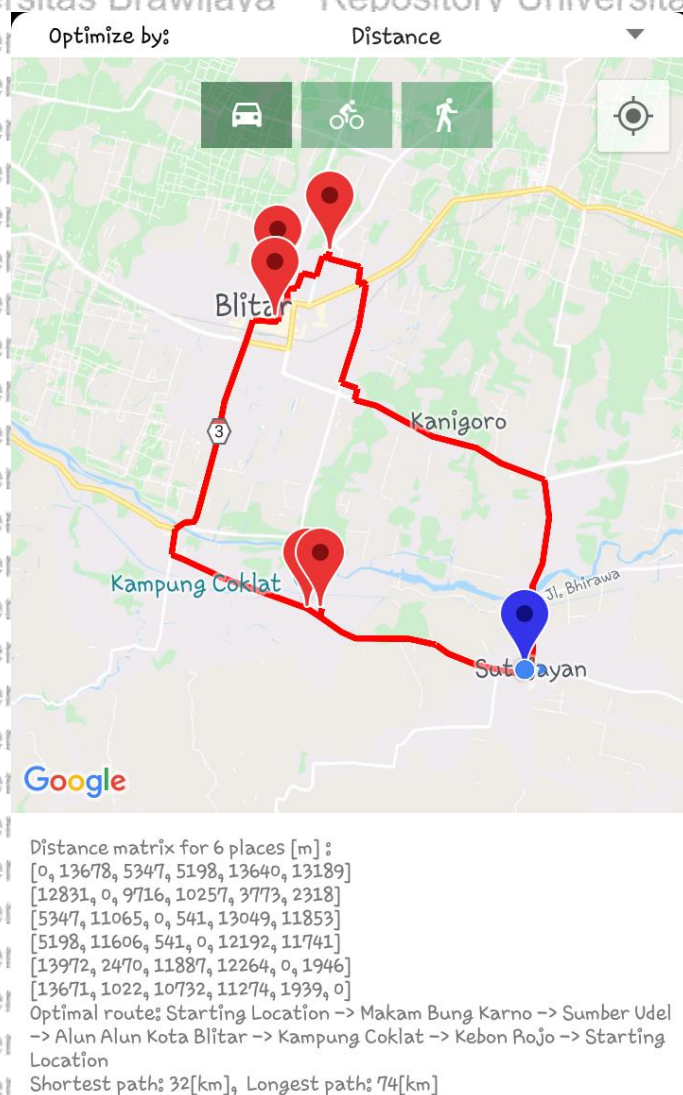
TEMUKAN RUTE



Gambar 6.4 Halaman Selected Place Activity

d. Halaman Map Activity

Halaman Map Activity digunakan untuk menampilkan rekomendasi rute terpendek berdasarkan daftar lokasi tempat wisata yang telah dipilih oleh actor, pada halaman ini terdapat marker-marker yang merepresentasikan lokasi tiap tempat wisata dan polyline yang menghubungkan marker tersebut yang merepresentasikan rute perjalanan terpendek dari daftar lokasi tempat wisata tersebut. Pada Halaman ini juga memberikan informasi rekomendasi rute dengan jalur terpendek yang telah ditemukan dan juga informasi tentang urutan tempat-tempat yang direkomendasikan dikunjungi terlebih dahulu sehingga rute perjalanan akan optimal. Di bagian ini, ada juga matriks jarak atau matriks waktu dari mana rute terbaik ditemukan. Matriks ini dibuat menggunakan layanan web Google Distance Matrix. Gambar 6.5 menunjukkan implementasi antar muka halaman Map Activity



Gambar 6.5 Halaman Map Activity

e. Halaman Add Place Activity

Halaman Add Place Activity digunakan untuk menambahkan data tempat wisata baru berdasarkan data nama tempat wisata, latitude, longitude & deskripsi tempat wisata. Data tempat wisata ini akan otomatis ditambahkan ke database sqlite aplikasi dan dapat ditampilkan pada daftar tempat yang dapat dikunjungi aktor dalam aplikasi ini. Terdapat tombol Add pada halaman ini yang berfungsi menambahkan data yang diinputkan dan ditampilkan di halaman Find Place Activity. Gambar 6.6 menunjukkan laman Add Place Activity.



←

Tambahkan Tempat Wisata

Masukkan Nama Tempat Wisata

✚

Makam Bung Karno

Lat [Contoh : 8,084819]

Lng [Contoh : 112,176346]

🌐

8,123456

🌐

112,211345

Masukkan Deskripsi Singkat Tempat Wisata

📝

Makam Presiden Bung Karno

TAMBAHKAN KE DATABASE

+

📍

Gambar 6.6 Halaman Add Place Activity

6.6 Implementasi Penyelesaian Traveling Salesman Problem

Traveling Salesman Problem merupakan permasalahan dimana seorang salesman harus mengunjungi setiap kota, salesman tersebut berangkat dari kota awal dan berakhir pada kota yang sama, serta setiap kota harus dikunjungi tidak lebih dari satu kali. Fungsi tujuan dari TSP adalah untuk mencari rute terpendek yang bisa dilalui oleh salesman sehingga dihasilkan total jarak terpendek dimana dalam penelitian kali ini peneliti menggunakan layanan Google Distance Matrix API. Pada Layanan API tersebut menyelesaikan permasalahan TSP dari beberapa titik lokasi tempat yang dipilih menggunakan algoritma Brute Force dengan Teknik Exhaustive Search dimana Exhaustive search adalah teknik pencarian solusi secara brute force untuk masalah yang melibatkan pencarian elemen dengan sifat khusus. Biasanya di antara objek-objek kombinatorik seperti permutasi, kombinasi, atau himpunan bagian dari sebuah himpunan. Penggunaan Brute force pada permasalahan TSP dapat diilustrasikan sebagai berikut:

Diberikan n buah kota serta diketahui jarak antara setiap kota satu sama lain. Temukan perjalanan (tour) terpendek yang melalui setiap kota lainnya hanya sekali dan kembali lagi ke kota asal keberangkatan. Permasalahan TSP



ini tidak lain adalah menemukan sirkuit Hamilton dengan bobot minimum. Langkah-langkah metode exhaustive search yaitu Buat list (Enumerasi) setiap solusi yang mungkin dengan cara yang sistematis. Lalu Evaluasi setiap kemungkinan solusi satu per satu, terlebih solusi yang mirror, bisa di keluarkan. Kemudian simpan solusi terbaik yang ada sampai saat ini. Kemudian Setelah pencarian berakhir, umumkan solusi terbaik. Terakhir Sumber daya yang dibutuhkan dalam pencarian solusi menggunakan Exhaustive Search sangat besar, meskipun algoritma exhaustive secara teoritis menghasilkan solusi. Implementasi Brute force pada permasalahan TSP pada aplikasi ini di implementasikan pada potongan code berikut :

```

134  @ static void solveTSP(int[][] valuesMatrix) {
135      shortestDistance = Integer.MAX_VALUE;
136      longestDistance = Integer.MIN_VALUE;
137      shortestPath = null;
138
139      int totalPlaces = valuesMatrix.length;
140      ArrayList<Integer> places = new ArrayList<>();
141      for(int i=0; i<totalPlaces; i++){
142          places.add(i);
143      }
144      int startPlace = places.get(0);
145      int currentDistance = 0;
146      bruteForceSearch(valuesMatrix, places, startPlace, currentDistance);
147  }
148  @ private static void bruteForceSearch(int[][] valuesMatrix, List<Integer> places,
149      int startPlace, int currentDistance) {
150      if(startPlace < places.size()-1){
151          for(int i=startPlace; i < places.size(); i++){
152              int tempCity = places.get(i);
153              places.set(i, places.get(startPlace));
154              places.set(startPlace, tempCity);
155              currentDistance = computeDistance(places, valuesMatrix);
156              bruteForceSearch(valuesMatrix, places, startPlace+1,
157                  currentDistance);
158              tempCity = places.get(i);
159              places.set(i, places.get(startPlace));
160              places.set(startPlace, tempCity);
161          }
162      }

```

Kode 6.5 Kode Program Penyelesaian TSP



BAB 7 PENGUJIAN

Pada tahap ini memberikan informasi hasil pengetesan Aplikasi Perangkat Bergerak Berbasis Lokasi Untuk Perencanaan Perjalanan Wisata Berdasarkan Rute Terpendek.

7.1 Pengujian Validasi

Merupakan pengujian untuk memperoleh informasi mengenai valid atau tidaknya *test case* yang telah direncanakan terdapat pada aplikasi. Statusnya ada untuk pengujian ini yaitu valid atau tidak valid.

7.1.1 Pengujian Validasi Kebutuhan Fungsional

Hasil pengecekan validasi aplikasi terdapat pada tabel 7.1.

Tabel 7.1 Pengujian Validasi Kebutuhan Fungsional

No	Kebutuhan Fungsional	Kasus Pengujian	Hasil yang Diharapkan	Hasil	Status
1	Merencanakan Rute	a. Tidak memilih lokasi wisata b. Memilih Lokasi tempat wisata yang sama lebih dari 1 kali c. Memilih beberapa tempat wisata dengan lokasi yang berbeda.	a. Menampilkan pesan "Select Place First" b. Menampilkan toast "This Place has already been Selected" c. Berhasil Menyimpan data lokasi tempat wisata yang telah dipilih oleh user	a. Menampilkan toast "Select Place First" b. Menampilkan toast "This Place has already been Selected" c. Berhasil Menyimpan data lokasi tempat wisata yang telah dipilih oleh user	Valid
2	Menampilkan Rekomendasi Rute Perjalanan Terpendek	a. Melihat Rekomendasi Rute Perjalanan Terpendek tanpa terkoneksi internet	a. Menampilkan pesan "Aplikasi Terhenti" b. Menampilkan rekomendasi rute perjalanan terpendek	a. Menampilkan pesan "Aplikasi Terhenti" b. Menampilkan	Valid



		b. Melihat Rekomendasi Route Perjalanan Terpendek dengan terkoneksi internet	dengan urutan tempat mana saja yang harus dikunjungi terlebih dahulu.	rekomendasi rute perjalanan terpendek dengan urutan tempat mana saja yang harus dikunjungi terlebih dahulu.	
3	Menambah Lokasi Tempat Wisata	a. Tidak mengisikan Data yang diperlukan secara lengkap. b. mengisikan Data yang diperlukan secara lengkap	a. Menampilkan pesan "Enter a valid data" b. Berhasil menyimpan data lokasi tempat wisata baru dan menampilkan toast "Added to database"	a. Menampilkan pesan "Enter a valid data" b. Berhasil menyimpan data lokasi tempat wisata baru dan menampilkan toast "Added to database"	Valid
4	Melihat Navigasi dari Lokasi saat ini ke Lokasi Tempat Wisata yang dipilih	a. Tidak terkoneksi dengan internet b. Terkoneksi dengan jaringan internet	a. Menampilkan pesan "Tidak terhubung internet" b. Memberi informasi Navigasi dari Lokasi saat ini ke Lokasi Tempat Wisata yang dipilih	a. Menampakka pesan "Tidak terhubung internet" b. Memberi informasi Navigasi dari Lokasi saat ini ke Lokasi Tempat Wisata yang dipilih	Valid



7.2 PENGUJIAN USABILITAS

Pada pengujian aplikasi penulis menggunakan kuisioner SUS dengan hasil yang dapat dilihat pada tabel 7.3

Tabel 7.2 Hasil Pengujian Usabilitas Sistem

	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13	R14	R15
P1	5	4	3	4	5	4	4	5	5	4	4	5	5	3	3
P2	3	2	2	3	2	1	2	3	2	2	3	2	2	3	2
P3	4	5	5	4	5	3	5	5	4	4	3	5	4	4	3
P4	2	1	3	3	1	2	3	2	1	3	4	2	2	2	3
P5	4	4	4	3	2	5	3	5	3	5	3	3	5	5	5
P6	1	3	2	3	1	2	2	3	3	3	1	2	2	3	2
P7	5	5	4	4	4	3	4	3	3	4	4	3	5	5	5
P8	1	2	3	1	3	3	2	3	1	2	3	2	2	2	1
P9	3	4	3	5	4	4	3	4	2	3	2	4	5	3	3
P10	2	2	3	2	3	1	2	1	3	2	3	3	2	1	2
	80	80	65	70	75	75	70	75	67,5	70	55	72,5	85	72,5	72,5
Total Skor SUS															1085
Rata - rata SUS															72,33333

Penjelasan

P berarti pernyataan

Angka 4 diartikan
Setuju

R berarti responden

Angka 5 diartikan
Sangat setuju

Angka 1 diartikan
Sangat tidak setuju

Angka 2 diartikan Tidak
setuju

Angka 3 diartikan
Netral

Rumus Skor *System Usability Scale* = $((P1 - 1) + (5 - P2) + (P3 - 1) + (5 - P4) + (P5 - 1) + (5 - P6) + (P7 - 1) + (5 - P8) + (P9 - 1) + (5 - P10)) \times 2,5$

Diperoleh hasil dengan total skor SUS 1085, didapatkan rata-rata 72,33 *System Usability Scale* dengan kategori tersebut digolongkan pada berstatus *acceptable*, dan mempunyai kategori *good*, dan tergolong memiliki predikat cukup.



BAB 8 PENUTUP

8.1 Kesimpulan

Bab berikut berisi kesimpulan dari hasil penelitian pengembangan aplikasi perangkat bergerak berbasis lokasi untuk perencanaan perjalanan wisata blitar berdasarkan rute terpendek, dan berikut kesimpulannya:

1. Analisis kebutuhan pada Aplikasi Perencanaan Perjalanan Wisata Blitar Berdasarkan Rute Terpendek mempunyai empat kebutuhan fungsional yaitu merencanakan rute, Menampilkan rekomendasi rute perjalanan terpendek, menambah lokasi tempat wisata, melihat navigasi lokasi terkini ke destinasi tempat wisata yang dipilih. Untuk kebutuhan non-fungsional berjumlah satu, yaitu User Acceptance.
2. Pada bagian bab perancangan Aplikasi Perencanaan Perjalanan Wisata Blitar Berdasarkan Rute Terpendek menggunakan *sequence diagram*, rancangan basisdata, rancangan Activity Diagram dan rancangan antarmuka.
3. Pada bagian implementasi rancangan dari aplikasi ini menggunakan menggunakan android studio dalam proses Coding, Database SQLite, Layanan Google Map Api, Google Place Api, Google Direction Api, dan Google Distance Matrix Api.
4. Pada Pengujian Aplikasi Perencanaan Perjalanan Wisata Blitar Berdasarkan Rute Terpendek didapatkan hasil kategori good pada kuisioner SUS dan pada pengujian validasi semua berstatus valid pada kebutuhan fungsional maupun non fungsional.

8.2 Saran

Pada pengembangan Aplikasi Perencanaan Perjalanan Wisata Blitar Berdasarkan Rute Terpendek perlu ditemukan alternatif lain dari layanan Google Distance Matrix Api hal tersebut dikarenakan layanan tersebut sebenarnya berbayar dan ada kuota penggunaan untuk yang versi gratis, diharapkan dapat menemukan tools atau layanan lain yang bersifat Open Source dan gratis agar semakin memudahkan dalam pengembangan Aplikasi yang serupa dengan tajuk yang hampir sama untuk kedepannya.



DAFTAR REFERENSI

Abdurrahman Albes Fajri, Endina Putri Purwandari, Funny Farady Coastera, 2018, Implementasi Sistem Informasi Geografis Untuk Pencarian Indekos Dengan Jarak Terpendek Menggunakan Metode Algoritma Dijkstra. Universitas Bengkulu.

Arifianto, Teguh. (2011). Membuat Interface Aplikasi Android Lebih Keren dengan LWUIT. Yogyakarta: Andi Publisher

Daniel, Galin. (2004). Software Quality Assurance. 90 Tottenham Court Road, London.

Darytamo, Budi, MT, dkk, 2007, "Pemrograman Berorientasi Obyek dengan Java 2 Platform Micro Edition (J2ME)", INFORMATIKA, Bandung.

Devlopers Android(2019, 27 Desember) Arsiterktur Platform. Diakses pada 27 Desember 2019, dari <https://developer.android.com/guide/platform?hl=id>

Fuady, Fahri Hikmawan 2017. Aplikasi Pencarian Rute Terpendek Bengkel Mobil Menggunakan Algoritma Dijkstra. STMIK AKAKOM, YOGYAKARTA. Abdurrahman Albes Fajri, Endina Putri Purwandari, Funny Farady Coastera, 2018, Implementasi Sistem Informasi Geografis Untuk Pencarian Indekos Dengan Jarak Terpendek Menggunakan Metode Algoritma Dijkstra. Universitas Bengkulu.

Google Cloud Platform Documentation(2008, 1 April) Get started with Google Cloud. Diakses pada 27 Desember 2019 dari <https://cloud.google.com/docs>

Helga, Aditya Rizqi Geovani, 2016, Implementasi Algoritma Dijkstra Untuk Mengetahui Lokasi Tempat Ibadah Umat Muslim Di Kota Malang Pada Aplikasi Mobile Phone (Studi Kasus Tempat Ibadah Di Wilayah Kecamatan Lowok Waru), Skripsi, Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Küpper, Axel. (2005). Location-Based Services: Fundamentals and Operation. 10.1002/0470092335.ch2.

Munir, Rinaldi, Matematika Diskrit, Penerbit Informatika, 2005.



Raymond Mcleod, Jr., Sistem Informasi Management, Jilid Dua, Edisi Bahasa Indonesia, PT. Bhuana Ilmu Populer, Jakarta, 2001.



LAMPIRAN

Kuisisioner SUS dapat dilihat pada Lampiran 1

Lampiran 1 Kuisisioner Pertanyaan SUS

1.	Saya pikir bahwa saya akan ingin lebih sering menggunakan aplikasi ini				
	☐	☐	☐	☐	☐
	Sangat Tidak Setuju	Tidak Setuju	Netral	Setuju	Sangat Setuju
2.	Saya menemukan bahwa aplikasi ini, tidak harus dibuat serumit ini				
	☐	☐	☐	☐	☐
	Sangat Tidak Setuju	Tidak Setuju	Netral	Setuju	Sangat Setuju
3.	Saya pikir aplikasi mudah untuk digunakan				
	☐	☐	☐	☐	☐
	Sangat Tidak Setuju	Tidak Setuju	Netral	Setuju	Sangat Setuju
4.	Saya pikir bahwa saya akan membutuhkan asisten untuk dapat menggunakan aplikasi ini				
	☐	☐	☐	☐	☐
	Sangat Tidak Setuju	Tidak Setuju	Netral	Setuju	Sangat Setuju
5.	Saya menemukan berbagai fungsi pada aplikasi ini diintegrasikan dengan baik				
	☐	☐	☐	☐	☐
	Sangat Tidak Setuju	Tidak Setuju	Netral	Setuju	Sangat Setuju
6.	Saya pikir ada terlalu banyak ketidakkonsistenan dalam aplikasi ini				
	☐	☐	☐	☐	☐
	Sangat Tidak Setuju	Tidak Setuju	Netral	Setuju	Sangat Setuju
7.	Saya bayangkan bahwa kebanyakan orang akan mudah untuk mempelajari aplikasi ini dengan sangat cepat				
	☐	☐	☐	☐	☐
	Sangat Tidak Setuju	Tidak Setuju	Netral	Setuju	Sangat Setuju
8.	Saya menemukan, aplikasi ini sangat rumit untuk digunakan				
	☐	☐	☐	☐	☐
	Sangat Tidak Setuju	Tidak Setuju	Netral	Setuju	Sangat Setuju
9.	Saya merasa sangat percaya diri untuk menggunakan aplikasi ini				
	☐	☐	☐	☐	☐
	Sangat Tidak Setuju	Tidak Setuju	Netral	Setuju	Sangat Setuju



10.	Saya perlu belajar banyak hal sebelum saya bisa memulai menggunakan aplikasi				
	☐	☐	☐	☐	☐
	Sangat Tidak Setuju	Tidak Setuju	Netral	Setuju	Sangat Setuju

Transkrip Wawancara :

Nama : Dhanang Dzuhri Rahmatullah

Profesi : Tour Guide

Hari / Tanggal : Minggu, 19 Mei 2019

Tempat : Kota Wlingi di Kediaman Peneliti

Data hasil wawancara dapat dilihat pada Lampiran 2

Lampiran 2 Data Hasil Wawancara

Peneliti : Bagaimana pendapat anda sebagai Tour Guide jika di kembangkan sebuah Aplikasi pada Platform Perangkat Bergerak yang dimana fungsinya untuk memperoleh suatu rute dengan jarak tempuh terpendek berdasarkan lebih dari satu destinasi tempat wisata yang akan dikunjungi?
Dhanang : Menurut saya hal tersebut merupakan ide yang bagus karena sebagai Tour Guide saya melakukan perjalanan untuk mengunjungi beberapa destinasi tempat wisata dalam sekali rute perjalanan dan mengingat di Kota Blitar terdapat banyak tempat wisata yang membuat orang-orang secara umum ingin mengunjungi lebih dari 1 lokasi tempat wisata ketika mereka liburan.
Peneliti : Menurut anda apa fitur utama yang penting dari aplikasi tersebut selain menampilkan rekomendasi rute terdekat?
Dhanang : Pengguna harus bisa memilih tempat-tempat wisata yang nanti akan mereka kunjungi secara manual sehingga aplikasi tersebut nantinya dapat membuat sebuah rute berdasarkan tempat-tempat yang dipilih tersebut lai
Peneliti : Apakah suatu ide yang bagus untuk memanfaatkan layanan API Google Place dimana fungsinya agar aplikasi tersebut dapat menyediakan data semua lokasi tempat wisata agar Pengguna tinggal mudah memilih tempat wisata sesuai keinginan mereka?
Dhanang : Pendapat saya itu merupakan ide yang bagus untuk memanfaatkan API Google agar menyediakan semua data tempat wisata mengingat Google pasti mempunyai data terlengkap untuk hal tersebut seperti hal nya pada aplikasi Google Map.
Peneliti : Apa kiranya penambahan fitur lain pada Aplikasi ini yang perlu ditambahkan?
Dhanang : Opini saya alangkah lebih baik jika ditambahkan lagi fitur aplikasi supaya pengguna dapat menambahkan lokasi tempat wisata secara manual



pada aplikasi untuk solusi jika terdapat kasus sebuah tempat wisata yang ingin dikunjungi oleh pengguna data nya tidak ditemukan atau belum teridentifikasi sebagai sebuah tempat wisata oleh Google dengan alasan mungkin karena tempat tersebut masih baru atau masih belum banyak diketahui, dsb