

BAB II TINJAUAN PUSTAKA

2.1 Algoritma A Star

Algoritma A Star digunakan untuk mencari jalur tercepat atau terpendek dari suatu titik ke titik lain. Dalam konteks LBS, algoritma ini dapat digunakan untuk mencari jalur terpendek atau tercepat dari lokasi pengguna ke lokasi yang diinginkan seperti tempat wisata atau restoran (S *et al.*, 2023).

Secara umum, algoritma A Star adalah mendefinisikan area pencarian menjadi sekumpulan node-node (tiles). Titik awal dan akhir ditentukan terlebih dahulu untuk memulai penelusuran pada tiap-tiap node yang memungkinkan untuk ditelusuri. Dari sini, akan diperoleh skor yang menunjukkan besarnya biaya untuk menempuh jalur yang ditemukan, ditambah dengan nilai heuristik yang merupakan nilai biaya estimasi dari node yang ada menuju tujuan akhir. Iterasi akan dilakukan hingga akhirnya mencapai target yang dituju (Ar, Patmanthara and Rahmatwinto, 2021).

Berikut adalah rumus perhitungan yang digunakan dalam penerapan A Star :

$$\begin{array}{l} f(n) \\ h(n) \end{array} = \begin{array}{l} g(n) \\ + \end{array} \dots\dots\dots (1)$$

Keterangan :

f(n) = fungsi dari evaluasi

g(n) = jarak tempuh antar daerah yang ditunjukkan dengan angka-angka yang terdapat pada sisi antar simpul dalam graf

h(n) = fungsi heuristik, yang mana merupakan *straight-line distance* (jarak secara garis lurus) antara daerah-daerah patokan dengan daerah yang dituju.

2.2 Android

Android pertama kali rilis pada bulan Oktober 2003 yang dikembangkan oleh Andy rubin, Rich miner, Chris White dan Nick Sears dengan sebuah perusahaan bernama android Inc di Palo Antom, California, dengan dukungan finansial dari Google, yang kemudian membelinya pada tahun 2005, tujuan awal pengembangan platform ini adalah mengembangkan sebuah system operasi yang lebih canggih dari sebuah camera digital. Karena, pasar global mengubah arus Andy dan kawan-kawan untuk membawa android inc beralih fungsi menjadi perusahaan yang bergerak pada pengembangan system smartphone.

System operasi mobile yang sangat populer pada saat ini adalah android. Pada saat ini hampir semua orang di dunia menggunakan system operasi mobile ini. Berbagai macam gadget menggunakan android sebagai system operasinya.

Android menawarkan pendekatan yang menyeluruh pada pengembangan aplikasi, dimana satu aplikasi android yang dibangun dapat berjalan di berbagai perangkat yang menggunakan system operasi android, baik yang digunakan *smartphone*, smartwatch, tablet, dan perangkat lainnya. Dalam bahasa inggris istilah android diartikan “Robot yang menyerupai manusia”, dapat dilihat dari icon android yang melambangkan robot berwarna hijau yang memiliki sepasang kaki dan sepasang tangan. Sebagai system operasi, android mempunyai fungsi sebagai penghubung antara perangkat keras dengan pengguna (user). Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat bergerak layar sentuh dari telepon pintar dan komputer tablet (Nadya Firly,2018).

Daya pikat android sebagai system operasi yang sangat populer saat ini adalah terletak pada platform opensource yang membuka peluang besar pada pengembang aplikasi untuk pengembangan teknologi terbaru (Parma et al., 2021).

2.3 Google Maps API

Google Maps API merupakan pengembangan teknologi dari google yang digunakan untuk menanamkan *Google Map* di suatu aplikasi yang tidak dibuat oleh Google. *Google Maps API* merupakan suatu *library* dengan bentuk *javascript* yang berguna untuk memodifikasi peta yang ada di *Google Maps* sesuai dengan kebutuhan. Dalam perkembangannya *Google Maps API* diberikan kemampuan untuk

mengambil gambar peta statis. Melakukan *geocoding*, dan memberikan penuntun arah. *Google Maps API* bersifat gratis untuk publik.

Penggunaan *Google Maps API* pada pengembangan aplikasi *android* dengan menggunakan *Eclipse* dan komputer menggunakan sistem operasi *windows*. *Google Maps API* terbagi menjadi 4 *platform* yaitu *web*, *Webservice*, *Android*, dan *iOS* (Setyawan and Munari, 2020).

Adapun *platform Google Maps API* adalah :

1. *Google Maps Android API*

<https://developers.google.com/maps/documentation/android-api/>

API ini digunakan untuk aplikasi yang menampilkan peta seperti GIS di Android. Fitur yang disediakan yaitu *3D Building*. *Custom Map*. *Custom Marker*, Integrasi dengan *webservice* pihak ketiga dan yang lainnya. Intinya jika dalam pembuatan aplikasi GIS seperti peta masjid atau sebaran penduduk di Suatu daerah, maka inilah APInya. Bahasa yang dipakai adalah JAVA.

2. *Google Place API for Android*

<https://developers.google.com/places/androidapi/>

Digunakan untuk aplikasi yang membutuhkan fitur mendeteksi lokasi disekitar user di *Android* dengan bahas JAVA. Misalkan daftar restoran terdekat atau pom bensin terdekat.

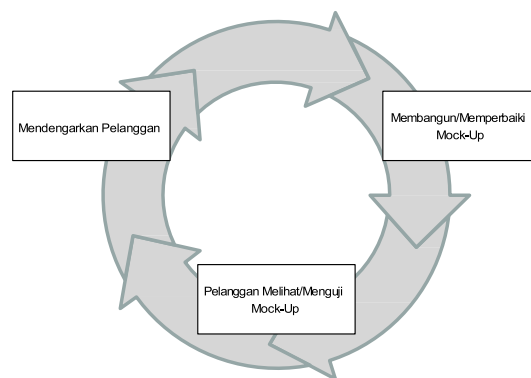
2.4 Metode *Prototype*

Prototyping merupakan teknik pengembangan sistem yang menggunakan *prototype* untuk menggambarkan sistem, sehingga pengguna atau pemilik sistem mempunyai gambaran pengembangan sistem yang akan dilakukannya. Teknik ini sering digunakan apabila pemilik sistem tidak terlalu menguasai sistem yang akan dikembangkannya, sehingga dia memerlukan gambaran dari sistem yang akan dikembangkannya tersebut. Dengan teknik *prototyping* pengembang bisa membuat *prototype* terlebih dahulu sebelum mengembangkan sistem yang sebenarnya (Mulyani, 2017).

Model pengembangan *prototyping* tidak memaksakan pengembang untuk

membuatkan prototipe dalam bentuk software, meskipun pada pelaksanaannya, sebagian besar prototipe yang dikembangkan berupa software. Terdapat beberapa jenis pemodelan *prototyping* yang dapat dipilih oleh pengembang. Berdasarkan kepada perbedaan metodologi yang digunakan di dalam pengembangan software, tujuan pengembangan software, dan area spesifik pengembangan software, maka terdapat beberapa jenis *prototyping* (Parma et al., 2021; Putu, 2023).

Metode prototipe merupakan metode pengembangan sistem dimana hasil analisa perbagian sistem langsung diterapkan kedalam sebuah model tanpa menunggu seluruh sistem selesai. *Mock-up* adalah sesuatu yang digunakan sebagai model desain yang digunakan untuk mengajar, demonstrasi, evaluasi desain, promosi, atau keperluan lain. Sebuah *mock-up* disebut sebagai prototipe perangkat lunak jika menyediakan atau mampu mendemonstrasikan sebagian besar fungsi sistem perangkat lunak dan memungkinkan pengujian desain sistem perangkat lunak. Iterasi terjadi pada pembuatan prototipe sampai sesuai dengan keinginan pelanggan atau pengguna (Rosa and Shalahuddin, 2018). Gambar 2.1 berikut merupakan pemodelan *prototype*:



Gambar 2. 1 Pemodelan Prototype

Adapun langkah atau tahapan penggunaan metode prototipe dalam penelitian adalah :

1. Tahap Mendengarkan Pelanggan

Pada tahapan ini, peneliti berkomunikasi dengan pelanggan untuk mengetahui kebutuhan sistem yang akan dibuat. Peneliti mendengarkan

keluhan pelanggan tentang pencarian lokasi JNE terdekat.

2. Tahap Membangun/Memperbaiki *Mock-Up*

Setelah pengumpulan data yang dibutuhkan didapatkan, maka selanjutnya merancang pencarian lokasi JNE terdekat berbasis Android dengan menerapkan algoritma A Star. Perancangan sistem terdiri dari *use case diagram*, *activity diagram*, *database*, dan antarmuka aplikasi berbasis *Android*. Setelah dilakukan perancangan, maka rancangan tersebut diimplementasikan ke dalam pemrograman aplikasi *Android*. Bahasa pemrograman yang digunakan adalah *Kotlin* dan penyimpanan data menggunakan *MySQL*. Aplikasi yang digunakan dalam media pembuatan aplikasi *Android* adalah *Android Studio*.

3. Tahap Pelanggan Melihat/Menguji *Mock-Up*

Tahap ini dilakukan pengujian aplikasi dengan menguji pemasangan aplikasi dengan tujuan apakah aplikasi mampu berjalan pada *platform smartphone* Android dengan sebagaimana semestinya yang mengacu pada fungsional sistem.

2.5 UML (Unified Modelling Language)

Grady Booch, Jim Rumbaugh, dan Ivar Jacobson mengembangkan *UML* pada pertengahan 1990-an dengan banyak umpan balik dari komunitas pengembangan perangkat lunak.

UML (Unified Modelling Language) adalah “bahasa standar untuk penulisan cetak biru perangkat lunak. *UML* dapat digunakan untuk memartifak-artifak suatu sistem software-intensive”. Dengan kata lain, samaan konstruksi, arsitek perangkat lunak membuat diagram *UML* untuk membantu pengembangan perangkat lunak membangun perangkat lunak. (Roger S. Pressman, ph.D 2012).

Berikut merupakan beberapa alat bantu dalam pengerjaan pembuatan sistem :




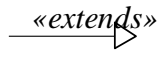

1. *Use Case Diagram*

Use Case mendiskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case*

digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Tujuan utama pemodelan *use case* adalah untuk memutuskan dan mendiskripsikan kebutuhan-kebutuhan fungsional sistem dan juga menyediakan basis untuk melakukan pengujian sistem yang memverifikasi sistem. Syarat penamaan *use case* adalah nama didefinisikan sesederhana mungkin dan dapat dipahami. Tabel 2.2 berikut merupakan simbol dan keterangan use case diagram :

Tabel 2. 1 Simbol dan Keterangan *Use Case Diagram*

Simbol	Deskripsi
<p><i>Use case</i></p> 	<p>Fungsionalitas yang disediakan sistem sebagai unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja diawal frase nama <i>use case</i></p>
<p>Aktor</p> 	<p>Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar itu sendiri.</p>
<p>Asosiasi</p> 	<p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.</p>
<p>Ekstensi</p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i>, dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan.</p>
<p>Generalisasi</p> 	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.</p>





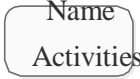
<p><i>Include</i></p> <p><<Include>></p>	<p><i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan</p>
--	---


2. Activity Diagram

Activity Diagram menggunakan aliran fungsionalitas sistem. Dapat juga digunakan untuk menggambarkan aliran kejadian (*flow of events*) dalam *use case*. Aktivitas dalam digram dipersentasikan dengan bentuk bujur sangkar bersudut tidak lancip, yang didalamnya berisi langkah-langkah apa saja yang terjadi dalam aliran kerja.

Diagram aktivitas diagram ini tidak perlu dibuat untuk setiap aliran kerja, tetapi diagram ini akan sangat berguna untuk aliran kerja yang melebar dan kompleks. Tabel 2.3 berikut merupakan simbol dan keterangan activity diagram :

Tabel 2. 2 Simbol dan Keterangan Activity Diagram

Simbol	Deskripsi
<p>Status awal</p> 	<p>Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.</p>
<p>Aktivitas</p> 	<p>Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.</p>
<p>Percabangan</p> 	<p>Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.</p>
<p>Penggabungan</p> 	<p>Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.</p>
<p><i>Swimlane</i></p> 	<p>Memisahkan Organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.</p>

Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
---	--

2.6 Black Box Testing

Black box testing merupakan pengujian untuk mengetahui fungsi perangkat lunak yang telah berjalan sesuai dengan kebutuhannya. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan pengeluaran perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Pengujian kotak hitam (*black box*) juga disebut pengujian perilaku yang berfokus pada persyaratan fungsional perangkat lunak, artinya teknik pengujian *black box* memungkinkan untuk membuat beberapa kumpulan kondisi masukan yang sepenuhnya akan melakukan semua kebutuhan fungsional untuk program (Rosa and Shalahuddin, 2018). *Black box testing* berupaya untuk menemukan kesalahan dalam kategori berikut :

1. Fungsi yang salah atau hilang.
2. Kesalahan antarmuka.
3. Kesalahan dalam struktur data atau akses basis data eksternal.
4. Kesalahan perilaku atau kinerja.
5. Kesalahan instalasi dan penghentian.

2.7 Penelitian terdahulu

Penelitian terdahulu yang terkait dengan penelitian yang dilakukan saat ini adalah seperti pada tabel.

Tabel 2. 3 berikut merupakan penelitian terdahulu :

NO	PENULIS	ARTIKEL DAN TAHUN	KESIMPULAN
1	Ativ Mutsaqov, Fernando and Ayu Megawaty	Penerapan Algoritma A Star Pada Aplikasi Pencarian Lokasi Foto Berbasis Android(2020)	Hasil dari penelitian ini berupa sebuah aplikasi yang dapat memberikan informasi mengenai spot atau lokasi foto yang ada di Kota Bandar Lampung,

			<p>dilengkapi dengan penunjuk jalan berupa maps yang telah diberikan algoritma A Star untuk menentukan rute terdekat menuju lokasi atau spot foto.</p> <p>Sedangkan penelitian yang sedang dilakukan adalah memberikan petunjuk pengguna rute tercepat ke kantor cabang JNE provinsi lampung.</p>
2	Hadi Nuryoso, Pradjoko and Lelah	Penerapan Algoritma A* pada Pencarian Rute Terpendek pada Rute Angkot Di Kota Sukabumi(2020)	<p>Berdasarkan implementasi dari Algoritma A Star yang diterapkan pada sistem yang berbentuk halaman situs web dengan hasil yang diperoleh dari sistem yang dibuat yaitu dapat melakukan pencarian rute terdekat pada rute angkot dan juga dapat menghitung jarak tempuh dari titik awal ke titik tujuan. Perbedaan Penelitian pada penjelasan berikut dengan penelitian yang sedang di lakukan adalah penelitian tersebut menggunakan sistem yang berbentuk web sedangkan saya menggunakan android dalam pencarian lokasi kantor JNE</p>
3	Pasaribu et al., 2022	Perancangan Aplikasi Objek Wisata di Kabupaten Toba dengan Algoritma A* Berbasis Mobile dan SIG(2020)	<p>Berdasarkan implementasi dari Algoritma AStar yang diterapkan pada sistem berbentuk sebuah aplikasi android dengan hasil diperoleh dari sistem dibuat yaitu dapat melakukan pencarian rute terdekat objek wisata. Didukung oleh sumber daya alam dan keindahan Danau Toba, sektor pariwisata</p>

			merupakan sektor potensial yang menjadi andalan di Kabupaten Toba. Sedangkan penelitian yang sedang di lakukan adalah memudahkan pengguna dalam menemukan cabang kantor JNE pada Provinsi Lampung dalam Pengambilan suatu Barang atau paket pesanan online
4	Arfida, Wibowo and Tarwiyati	Aplikasi Pemetaan Terhadap Penyebaran Lokasi Sekolah Agama Di Provinsi Lampung(2020)	Hasil dari penelitian ini berupa Sistem yang dapat digunakan untuk Mengetahui letak penyebaran sekolah agama di Provinsi Lampung. Perbedaan penelitian pada artikel tersebut dengan yang sedang di lakukan adalah penelitian tersebut dilakukan untuk mengetahui suatu titik tata letak suatu lokasi sekolah agama di Provinsi Lampung sedangkan yang sedang dilakukan mencari suatu lokasi kantor cabang JNE terdekat
5	Susanto and Handayani	Aplikasi Pencarian Jasa Pangkas Rambut Di Bandar Lampung Menggunakan Algoritma A-Star Berbasis Android(2022)	Aplikasi pencarian pangkas rambut berbasis Android dengan menerapkan metode A Star dapat membantu masyarakat dalam mencari rute pangkas rambut terdekat. Perbedaan penelitian yang di lakukan dengan artikel tersebut adalah memudahkan pengguna dalam pencarian lokasi kantor cabang JNE dengan jalur yang tercepat sehingga menghemat waktu dan biaya