

Implementasi Algoritma A Star Untuk Optimasi Pencarian Gym Terdekat Pada Aplikasi Android On My Gym

Dwi Rizki Darmawan^{1*}, Dede Supiyan²

^{1,2}Teknik Informatika, Universitas Pamulang, Jl. Surya Kencana No.1, Pamulang Barat, Banten, Indonesia

e-mail: ¹dwirizki164@gmail.com, ²dosen02353@unpam.ac.id

Submitted Date: April 13, 2026

Reviewed Date: April 16, 2026

Revised Date: April 21, 2026

Accepted Date: Mei 24, 2026

Abstract

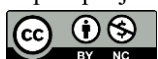
Advancements in information technology have driven the transformation of fitness services into mobile platforms; however, the utilization of such technology for gym location search and navigation remains suboptimal. The On My Gym (OMG) application, developed by PT. Airlangga Teknologi Solusi Nusantara, faces challenges related to low search accuracy and inefficient navigation routes. A pre-launch survey revealed a user satisfaction rate of 63%, indicating the need for improvements in search relevance and navigation usability. This study aims to implement the A Star algorithm in the Android-based OMG application to optimize the process of searching and determining the nearest gym routes. The A Star algorithm is applied by combining the actual cost function $g(n)$ and a distance-based heuristic function $h(n)$ to identify the optimal path. Performance evaluation is conducted using gym location data to assess search efficiency and route accuracy. The results show that the implementation of the A Star algorithm achieves efficient performance, with 251 nodes explored, 243 nodes expanded, and an execution time of 0.55 ms in a single test scenario. The system is also capable of presenting alternative gym locations at distances of 324 m, 832 m, and 1.89 km, demonstrating relevance to the user's position. These findings indicate that the A Star algorithm improves search efficiency and navigation accuracy, thereby enhancing the overall quality of mobile-based fitness service applications.

Keywords: Search Location, Gym, A Star

Abstrak

Perkembangan teknologi informasi mendorong transformasi layanan kebugaran ke platform mobile, namun pemanfaatan teknologi untuk pencarian lokasi gym dan navigasi masih belum optimal. Aplikasi On My Gym (OMG) yang dikembangkan oleh PT. Airlangga Teknologi Solusi Nusantara menghadapi permasalahan berupa rendahnya akurasi hasil pencarian dan ketidakefisienan rute navigasi. Hasil survei pra-peluncuran menunjukkan tingkat kepuasan pengguna sebesar 63%, yang mengindikasikan perlunya peningkatan pada aspek relevansi pencarian dan kenyamanan navigasi. Penelitian ini bertujuan mengimplementasikan algoritma A Star pada aplikasi OMG berbasis Android untuk mengoptimalkan proses pencarian dan penentuan rute gym terdekat. Algoritma A Star diterapkan dengan menggabungkan biaya aktual $g(n)$ dan fungsi heuristik $h(n)$ berbasis jarak untuk menentukan jalur optimal. Evaluasi dilakukan melalui pengujian performa sistem menggunakan data lokasi gym serta analisis hasil pencarian dan navigasi. Hasil pengujian menunjukkan bahwa implementasi algoritma A Star menghasilkan performa yang efisien dengan 251 node explored, 243 node expanded, dan waktu eksekusi sebesar 0,55 ms dalam satu skenario pengujian. Sistem juga mampu menampilkan alternatif lokasi gym dengan jarak 324 m, 832 m, dan 1,89 km secara relevan berdasarkan posisi pengguna. Temuan ini menunjukkan bahwa penerapan algoritma A Star meningkatkan efisiensi proses pencarian dan ketepatan rute navigasi, sehingga berkontribusi pada peningkatan kualitas layanan aplikasi kebugaran berbasis mobile.

Kata kunci: Pencarian Lokasi, Gym, A Star



1. Pendahuluan

Perkembangan teknologi informasi dan komunikasi telah mendorong adopsi aplikasi berbasis mobile di berbagai sektor, seperti transportasi, perbankan, dan layanan perjalanan. Namun, pemanfaatan teknologi serupa pada industri kebugaran, khususnya dalam pencarian lokasi gym dan navigasi, masih belum optimal. Pengguna tidak hanya membutuhkan informasi lokasi, tetapi juga mengharapkan hasil pencarian yang relevan berdasarkan posisi serta rute yang akurat dan efisien. Ketidaktepatan hasil pencarian dan rute yang tidak optimal dapat meningkatkan waktu pencarian dan menurunkan kualitas pengalaman pengguna.

Aplikasi On My Gym (OMG) dikembangkan sebagai platform mobile untuk membantu pengguna menemukan dan mengakses layanan kebugaran. Pada tahap awal pengembangan, ditemukan permasalahan pada akurasi pencarian lokasi dan efisiensi rute navigasi. Hasil survei pra-peluncuran menunjukkan tingkat kepuasan pengguna sebesar 63%, yang mengindikasikan bahwa sistem belum mampu memberikan hasil pencarian dan navigasi yang sesuai dengan ekspektasi pengguna. Selain itu, studi literatur menunjukkan bahwa sistem Location-Based Services tanpa mekanisme optimasi jalur yang memadai cenderung menghasilkan rute yang kurang efisien, terutama ketika jumlah data lokasi meningkat.

Beberapa penelitian sebelumnya telah mengkaji penggunaan algoritma pencarian jalur, namun penerapan pada konteks pencarian lokasi gym berbasis mobile dengan integrasi sistem pencarian dan navigasi masih terbatas. Hal ini menunjukkan adanya research gap dalam pengembangan sistem pencarian lokasi kebugaran yang tidak hanya menampilkan hasil terdekat, tetapi juga mempertimbangkan efisiensi rute secara komputasional.

Untuk mengatasi permasalahan tersebut, penelitian ini mengusulkan penerapan algoritma A Star sebagai metode optimasi pencarian rute. Algoritma A Star mengombinasikan biaya aktual dari titik awal ke suatu node ($g(n)$) dan estimasi biaya menuju tujuan melalui fungsi heuristik ($h(n)$), sehingga mampu menghasilkan jalur terpendek secara lebih efisien dibandingkan metode konvensional. Karakteristik ini menjadikan A Star sesuai untuk diimplementasikan pada aplikasi berbasis Android yang membutuhkan respon cepat dan akurasi tinggi dalam proses navigasi.

Penelitian ini merupakan jenis penelitian eksperimen dengan pendekatan implementasi sistem. Algoritma A Star diintegrasikan ke dalam aplikasi OMG untuk mengoptimalkan proses pencarian dan navigasi gym terdekat. Evaluasi dilakukan menggunakan dataset lokasi gym dengan beberapa skenario pengujian berdasarkan posisi pengguna yang berbeda. Metrik evaluasi yang digunakan meliputi akurasi rute yang dihasilkan dan waktu komputasi dalam proses pencarian. Selain itu, dilakukan analisis terhadap performa algoritma untuk mengidentifikasi kelebihan dan keterbatasannya dalam konteks aplikasi berbasis lokasi.

Dengan pendekatan tersebut, penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan sistem pencarian dan navigasi berbasis mobile, khususnya pada industri kebugaran, serta menjadi referensi untuk implementasi algoritma optimasi jalur pada aplikasi sejenis.

2. Metodologi Penelitian

Metode Penelitian ini digunakan sebagai pedoman dalam melaksanakan penelitian agar hasil yang dicapai tidak menyimpang dari tujuan yang telah ditentukan. Adapun metode penelitian yang digunakan Adalah sebagai berikut:

2.1 Metode Pengumpulan Data

Metode pengumpulan data dilakukan dengan beberapa teknik, antara lain:

a. Observasi

Mengamati dan mendokumentasikan kinerja aplikasi *On My Gym* serta algoritma *A Star* selama proses *development* untuk mengidentifikasi potensi masalah dan area optimasi yang diperlukan

b. Studi Literature

Penelitian ini diawali dengan melakukan studi literatur yang mencakup penelusuran dan analisis terhadap jurnal ilmiah, buku, artikel, dan sumber akademik lainnya yang relevan. Studi literatur ini bertujuan untuk memperoleh landasan teori yang kuat mengenai algoritma *A Star*, dan penggunaan *Flutter* dalam pengembangan aplikasi berbasis *Android*.

c. Kuesioner

Kuesioner digunakan untuk memperoleh data dari responden terkait penggunaan aplikasi, khususnya dalam menilai akurasi hasil pencarian, kemudahan navigasi, dan tingkat kepuasan pengguna terhadap sistem yang dikembangkan. Kuesioner ini diberikan kepada sejumlah responden yang sesuai dengan kriteria penelitian untuk mendapatkan data penilaian secara langsung dari pengguna.

d. Perancangan Sistem

Merancang arsitektur dan struktur implementasi algoritma *A Star* yang meliputi desain *database* lokasi *gym*, dan integrasi dengan *framework Flutter*.

2.2 Metode Pengembangan Sistem

Penelitian ini menggunakan metode Research and Development (R&D) dengan pendekatan kuantitatif. Penelitian berfokus pada implementasi algoritma *A Star* untuk pencarian *gym* terdekat dan analisis performanya melalui pengukuran waktu eksekusi. Jumlah node explored, dan akurasi hasil pencarian.

1. Data yang digunakan dalam penelitian ini terdiri atas data lokasi *gym*, data jaringan jalan, dan data lokasi pengguna. Data lokasi *gym* diperoleh dari koordinat geografis yang diambil melalui Google Maps. Data jaringan jalan diperoleh dari OpenStreetMap (OSM) yang kemudian diolah menjadi graf berbobot. Adapun data lokasi pengguna diperoleh melalui layanan GPS pada perangkat Android. Ketiga jenis data tersebut disusun menjadi representasi graf yang digunakan sebagai dasar perhitungan rute oleh algoritma *A Star*.
2. Proses pengembangan sistem dilakukan melalui beberapa tahap, yaitu pengumpulan data, pengolahan data, implementasi algoritma *A Star*, pengujian sistem, dan evaluasi hasil. Pada tahap pengolahan data, koordinat *gym*, node jalan, dan posisi pengguna dipetakan ke dalam struktur graf agar dapat diproses oleh algoritma pencarian jalur terpendek. Setelah itu, algoritma *A Star* diimplementasikan untuk menentukan rute optimal berdasarkan nilai biaya aktual $g(n)$ dan nilai heuristik $h(n)$.
3. Skenario pengujian dilakukan dengan menetapkan node awal dan node tujuan pada graf jaringan jalan. Pada iterasi awal, algoritma menginisialisasi node awal **63203** dengan nilai $g(63203) = 0$ karena belum terdapat perpindahan dari titik awal, sedangkan nilai heuristik $h(63203) = 185,17$ meter menunjukkan jarak langsung menuju node tujuan. Nilai evaluasi awal diperoleh sebesar $f(63203) = 185,17$ meter, kemudian node tersebut dimasukkan ke dalam *open list*.

Tabel 1. Iterasi Algoritma *A Star*

Iterasi	Current node	$g(n)$	$h(n)$	$f(n)$	Jarak edge (meter)	Action
0	63203	0.0000	185.1673	185.1673	-	Initialize (Star
1	63203	0.0000	185.1673	185.1673	-	Expand → 63
2	63202	67.3150	134.4173	201.7323	[67.3150]	Expand → 63
3	63200	85.1812	118.7764	204.0915	[17.8661]	Expand → 63
4	63204	104.1110	101.7675	186.9487	[18.9298]	Expand → 63
5	63198	122.9136	84.6134	205.527	[18.8026]	Expand → 75
6	75667	133.5148	75.2840	208.7988	[10.6012]	Expand → 63
7	63694	195.0700	56.7624	251.8324	[61.5552]	Expand → 63
8	63750	211.8997	53.3177	265.2174	[16.8297]	Expand → 63
9	63703	237.7286	37.3999	275.1285	[25.8289]	Expand → 53
10	5314	268.7713	42.2818	311.0531	[31.0427]	Expand → 33
11	3346	309.9888	14.7076	324.6963	[41.2175]	Expand → 12
12	12250	324.6963	0.0000	324.6963	[14.7076]	Goal

Pada iterasi berikutnya, node **63203** dipilih karena memiliki nilai $f(n)$ paling kecil, lalu diekspansi sehingga neighbor terdekatnya, yaitu node **63202**, dihitung kembali. Nilai $g(63202)$ diperoleh sebesar 67,3150 meter, nilai heuristik $h(63202) = 134,4173$ meter, sehingga $f(63202) = 201,73$ meter. Node **63202** kemudian dimasukkan ke dalam *open list*, sedangkan node **63203** dipindahkan ke *closed list*. Proses ini berlangsung secara iteratif hingga iterasi ke-11, ketika node tujuan **12250** berhasil dicapai dengan total jarak rute sebesar **324,69 meter**.

3. Hasil dan Diskusi

3.1 Analisa Masalah

Beberapa identifikasi masalah yang ditemukan dalam penelitian implementasi performa algoritma *a star* adalah

1. Hasil pencarian lokasi gym pada aplikasi belum sepenuhnya akurat dan belum selalu merepresentasikan lokasi terdekat berdasarkan posisi aktual pengguna, sehingga dapat menghambat proses pencarian gym yang paling sesuai.
2. Sistem pencarian dan navigasi pada aplikasi OnMyGym belum menerapkan algoritma optimasi jalur yang mampu menghasilkan rute terpendek secara efisien.
3. Kinerja sistem pencarian dan navigasi pada aplikasi OnMyGym masih dipengaruhi oleh kestabilan koneksi internet, sehingga berpotensi menyebabkan keterlambatan dalam proses pemrosesan dan penampilan rute kepada pengguna.

3.2 Perhitungan Algoritma A Star

Dan untuk menghitung algoritma *a star* pada aplikasi OMG ada beberapa hal penting yang harus dilakukan seperti:

1. Menentukan graf jaringan jalan terlebih dahulu untuk menggambarkan hubungan antar titik(*nodes*) dalam sistem navigasi, data ini berisikan koordinat dengan *format latitude* dan *longitude* yang merepresentasikan lokasi geografis setiap simpul dalam jaringan jalan. Data graf ini disusun dalam dua komponen utama, yaitu Tabel *Node*, yang berisi informasi posisi *geografis* setiap titik dalam sistem koordinat *global* dan Tabel *Edge*, mendefinisikan konektivitas antar node beserta bobot jaraknya, dimana.

jarak antar *node* dihitung menggunakan *formula haversine*, yaitu metode perhitungan jarak *geodesic* antara dua titik di permukaan bumi berdasarkan koordinat *latitude* dan *longitude*, dan berikut rumus *Formula Haversine*:

$$\text{lat} = \text{lat2} - \text{lat1}$$

$$\text{long} = \text{long2} - \text{long1}$$

$$a = \sin^2\left(\frac{\Delta \text{lat}}{2}\right) + \cos(\text{lat1}) \cdot \cos(\text{lat2}) \cdot \sin^2\left(\frac{\Delta \text{long}}{2}\right)$$

$$c = 2 \cdot a \tan^2(\sqrt{a}, \sqrt{1-a})$$

$$d = R \cdot C$$

Keterangan:

R = jari-jari bumi sebesar 6371(km)

lat = besaran koordinat latitude

long = besaran koordinat longitude

C = kalkulasi perpotongan sumbu

d = jarak (km)

I = derajat = 0.0174532925 radian

2. Algoritma *a star* menggunakan dua struktur data utama dalam proses pencarian jalur, yaitu *Open List* dan *Closed List*. Kedua struktur ini memiliki peran krusial dalam memastikan efisiensi dan keoptimalan hasil pencarian. *Open list*(daftar *node* kandidat), adalah struktur data yang menyimpan *node node* yang akan dieksplorasi, sedangkan *Closed List*(daftar *node* yang telah dikunjungi), adalah struktur data yang menyimpan *node-node* yang telah selesai dieksplorasi.



Table 2. Graf jaringan jalan lokasi pengguna ke gym terdekat

Node	Lokasi user (latitude dan longitude)
63203	106.6358694, -6.281936
63202	106.6364783, -6.2819232
63200	106.6365186, -6.2820788
63204	106.6365811, -6.2822373
63198	106.6366604, -6.2823869
75667	106.6367052, -6.2824712
63694	106.6368051, -6.2830158
63750	106.6369064, -6.2831288
63703	106.6369526, -6.2829011
5314	106.6369625, -6.2826221
3346	106.6371855, -6.2829192
12250	106.6372832, -6.2828294

Selanjutnya menentukan node mana yang akan dieksplorasi selanjutnya. Dengan rumus $f(n) = g(n) + h(n)$

Keterangan:

$f(n)$ = Total estimasi biaya dari start node ke goal node melalui node n

$g(n)$ = Biaya actual dari start node ke node n (jarak yang sudah ditempuh)

$h(n)$ = Estimasi biaya heuristic dari node n ke goal node (jarak tersisa).

Contoh:

$$f(63203) = 0 + 185.1673 = 185.1673$$

Keterangan:

0 adalah node awal karena proses pertama

185.1673 hasil dari node 63203 ke node goal 12250

Memasukkan node ke dalam open list dan node yang sudah di eksplorasi, akan dimasukkan untuk mencegah perlooping.

Iterasi 0: Inisialisasi

Table 3 Open List iterasi 0: Inisialisasi

$g(63203) = 0$ meter (node awal)
$h(63203) = 185.1673$ meter
jarak dari 63203 ke goal 12250
$f(63203) = g(63203) + h(63203)$
$= 0 + 185.1673 = 185.1673$

Open List: [63203]

Closed List: {}

Pilih node dengan $f(n)$ terkecil dari open list, dipilih node dengan nilai $f(n)$ terkecil. Saat ini hanya ada node 63203

Lalu node 63203 dipindahkan dari openlist ke closed list karena sudah mulai dieksplorasi

Iterasi 1: proses expand node 63202

Table 4 Open List iterasi 1:Expand node 63202

Open List = []

Closed List = {63203}

$$\begin{aligned} g(63202) &= g(63202) + \text{jarak}(63203 \rightarrow 63202) \\ &= 0 + 67.3150 \text{ meter} \\ &= 67.3150 \text{ meter} \end{aligned}$$

$$h(63202) = 134.4173$$

Jarak dari 63202 ke goal 12250

$$f(63202) = g(63202) + h(63202)$$

$$f(63202) = 67.3150 + 134.4173 = 201.7323 \text{ meter}$$

Update open list dan parent

Open List = [63202]

Closed List = [63202]

Parent = {63202: 63202}

State akhir iterasi 1:

Lalu pindah ke closed list

Table 5 Closed List iterasi 1

Open List = []

Closed List: [63203, 63202]

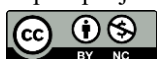
Iterasi 2: Proses expand node 63200

Table 6 Open List iterasi 2: Expand node 63200

Open List = []

Closed List = {63203, 63202}

$$\begin{aligned} g(63200) &= g(63202) + \text{jarak}(63202 \rightarrow 63200) \\ &= 67.3151 + 17.8661 \end{aligned}$$



= 85.1812 meter

$$h(63200) = 118.7764$$

Jarak dari 63200 ke goal 12250

$$f(63200) = g(63200) + h(63200)$$

$$f(63200) = 85.3151 + 118.7764 = 204.0915$$

meter

Update open list dan parent

Open List = [63200]

Closed List = [63203, 63202]

Parent = {63202: 63203, 63200: 63202}

State akhir iterasi 2:

Lalu update di closed list (pindahkan)

Table 7 Closed List iterasi 2

Open List: []

Closed List: [63203, 63202, 63200]

Dan lakukan proses perhitungan iterasi sampai node terakhir yaitu sampai proses Iterasi 11

Iterasi 11: Proses Expand node 12250

Table 8 Open List iterasi 11: Expand node 12250

Open List = []

Closed List = {63203, 63202, 63200, 63204, 63198, 75667, 63694, 63750, 63703, 5314, 3346}

$$g(12250) = g(3346) + \text{jarak}(3346 \rightarrow 12250)$$

$$= 309.9888 + 14.7076$$

$$= 324.6963 \text{ meter}$$

$$h(12250) = 0.0000$$

Jarak dari 12250 ke goal 12250

$$f(12250) = g(12250) + h(12250)$$

$$f(12250) = 324.6963 + 0.000 = 324.6963 \text{ meter}$$

Update open list dan parent

Open List = [12250]



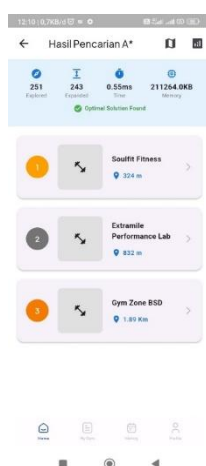
Closed List = [63203, 63202, 63200, 63204, 63198, 75667, 63694, 63750, 63703, 5314, 3346]
 Parent = {63202: 63203, 63200: 63202, 63204: 63200, 63198: 63204, 75667: 63198, 63694: 75667, 63750: 63694, 63703: 63750, 5314: 63703, 3346: 5314, 12250: 3346}

Table 9 Closed List iterasi 11

Open List: []

Closed List: [63203, 63202, 63200, 63204, 63198, 75667, 63694, 63750, 63703, 5314, 3346, 12250]

Dan hasilnya adalah menampilkan gym yang terdekat dengan pengguna dengan visual



Gambar 1 Hasil Implementasi Pencarian Gym Menggunakan Algoritma A Star

Hasil pengujian implementasi A-Star pada penelitian ini menunjukkan indikasi performa yang baik pada skenario pencarian gym terdekat: untuk satu sesi pengujian tercatat 251 node explored, 243 node expanded, dan waktu eksekusi sebesar 0,55 ms, serta aplikasi menampilkan beberapa alternatif lokasi (324 m, 832 m, dan 1,89 km). Sebagai pembandingan, Ester dan Pakpahan (2025) melaporkan hasil pengujian A-Star pada kasus pencarian rute rumah makan di Kota Medan dengan jarak rute terdekat 2,84 km, jumlah node yang dikunjungi 544, dan waktu eksekusi 6112 ms (International Multidisciplinary Journal Vol 1 No 02 November 2025).

Perbedaan angka tersebut tidak serta-merta menunjukkan superioritas mutlak salah satu implementasi, melainkan lebih mencerminkan perbedaan kondisi uji. Faktor-faktor yang mempengaruhi perbedaan performa meliputi skala dan kompleksitas jaringan graf (cakupan area dan densitas node), metode dan rentang pengukuran waktu eksekusi (mis. CPU time murni vs total server processing), jenis heuristik yang digunakan (Euclidean vs Haversine), serta detail implementasi dan platform eksekusi (bahasa pemrograman, struktur data dan optimasi).

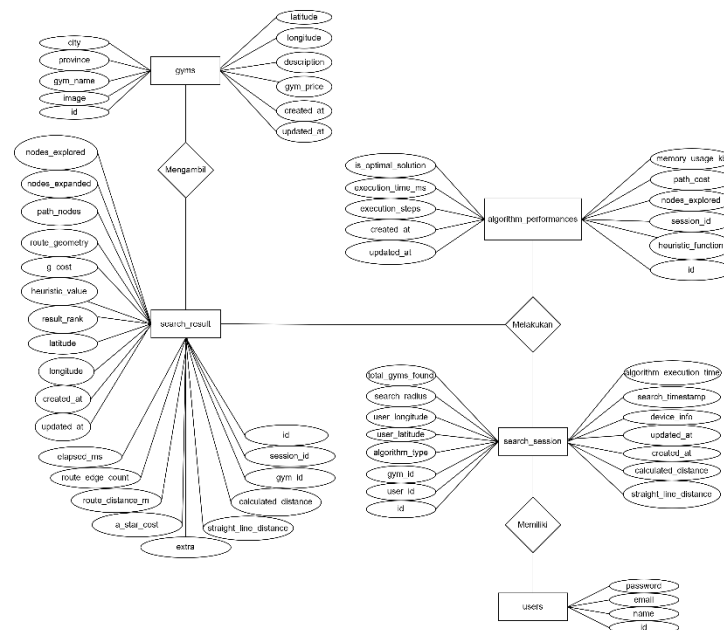
3.3 Perancangan Sistem

3.3.1 Perancangan Basis Data

Perancangan basis data dilakukan untuk menggambarkan struktur data yang digunakan.

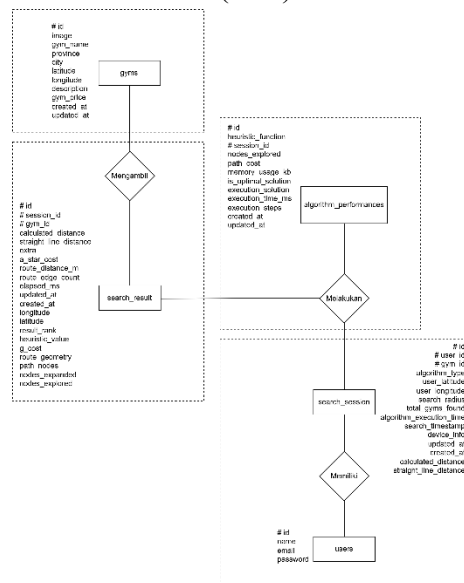
a. Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) digunakan untuk mendesain dalam basis data yang menunjukkan secara detail hubungan atau relasi antara objek dari entitas dan atributnya sehingga terbentuk secara terstruktur dan jelas menggunakan notasi dan simbol.



Gambar 2 Entity Relationship Diagram (ERD)

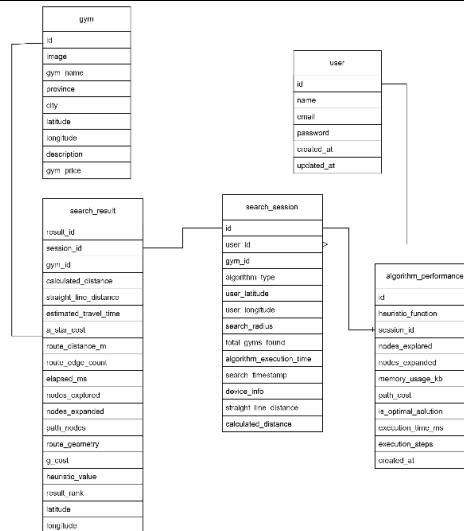
b. Transformasi ERD ke Logical Record Structure (LRS)



Gambar 3 Transformasi Entity Relationship Diagram (ERD)

c. Logical Record Structure (LRS)

Logical Record Structure (LRS) digunakan untuk merepresentasikan entitas ke dalam bentuk tabel beserta atribut dan kunci yang dimilikinya.

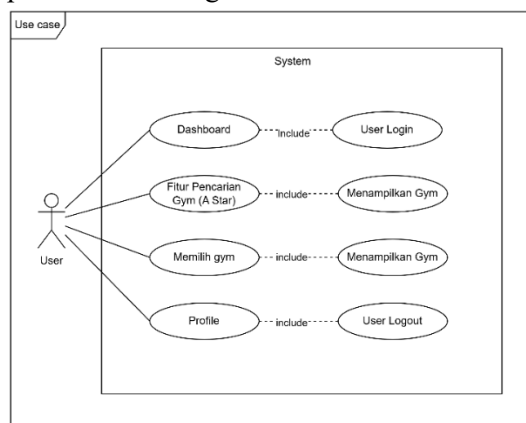


Gambar 4 Logical Record Structure (LRS)

3.4 Perancangan Aplikasi

a. Use Case Diagram

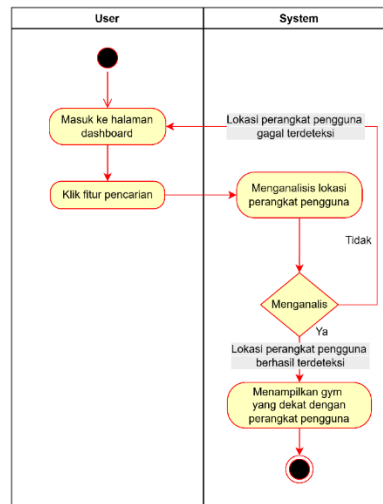
Berikut Adalah perancangan *Use Case Diagram* untuk pengembangan fitur pencarian pada aplikasi *On My Gym* berbasis *android*. Diagram ini menggambarkan interaksi antara *actor* (pengguna) dan sistem, di mana sistem tersebut mengimplementasikan algoritma *A Star*.



Gambar 5 Use Case Diagram

b. Activity Diagram

Activity diagram merupakan sistem yang memberikan aliran kerja dalam rancangan sebuah aktivitas yang akan dijalankan. *Activity diagram* juga dapat memberikan definisi atau pengelompokkan alur tampilan pada sebuah sistem dan memiliki komponen dari bentuk tertentu yang akan dihubungkan dengan tanda panah. Proses panah tersebut memberikan arah kepada urutan aktivitas dari proses awal hingga proses akhir, *Diagram* aktivitas atau *diagram activity* ini dapat membantu dalam memetakan kegiatan-kegiatan yang dilakukan secara berurutan. Setiap kegiatan akan terhubung pada garis yang menunjukkan urutan pada proses bisnisnya.

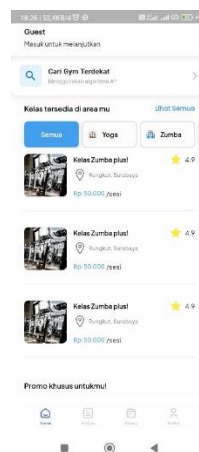


Gambar 6 Activity Diagram

3.5 Implementasi Sistem

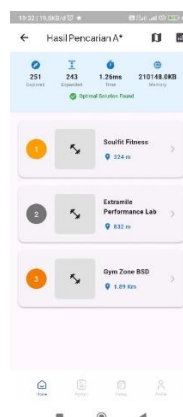
Implementasi dilakukan untuk memastikan sistem berjalan sesuai rancangan.

3.5.1 Tampilan dashboard (utama)



Gambar 7 Tampilan halaman dashboard

3.5.2 Tampilan halaman fitur pencarian gym menggunakan algoritma a star



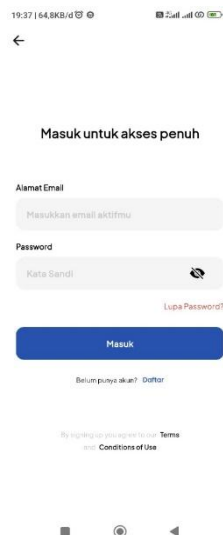
Gambar 8 Tampilan halaman saat pengguna menggunakan fitur pencarian gym menggunakan algoritma a star

3.5.3 Tampilan halaman pencarian bagian map



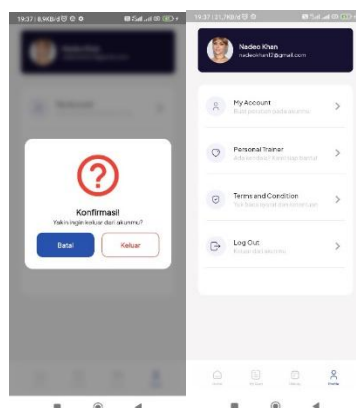
Gambar 9 Tampilan halaman saat pengguna klik icon map

3.5.4 Tampilan halaman login



Gambar 10 Tampilan halaman login, saat pengguna klik teks “Guest”

3.5.5 Tampilan halaman profile



Gambar 11 Tampilan halaman profile dan Ketika pengguna klik “Log Out”

3.6 Hasil Quesioner

Penulis menyebarkan kuesioner kepada pengguna untuk mengetahui sejauh mana sistem yang dikembangkan telah memenuhi kebutuhan mereka. Kuesioner terdiri dari 10 pertanyaan dengan 2 form yaitu sebelum aplikasi On My Gym diimplementasikan algoritma a star nya dan sesudah aplikasi On My Gym diimplementasikan algoritma a star nya, dengan pilihan jawaban Sangat Setuju (SS), Setuju (S), Cukup(C), Tidak Setuju (TS), Sangat Tidak Setuju (STS) yang diberi bobot masing-masing 5, 4, 3, 2, dan 1. Jumlah responden adalah sebanyak 30 orang, berikut adalah rekapitulasi hasil kuesioner yang telah didapatkan.

Table 10 Kuesioner Sebelum A Implementasi Algoritma A Star

No	Pernyataan	SS	S	C	TS	STS	Skor Total	Nilai Max	Presentase
1	Aplikasi OnMyGym mudah digunakan untuk mencari gym terdekat	1	-	10	17	2	71	150	47,33%
2	Tampilan antarmuka aplikasi jelas dan mudah dipahami	3	17	9	1	-	112	150	74,66%
3	Informasi gym yang ditampilkan aplikasi mudah untuk dibaca dan diakses	2	19	8	-	1	111	150	74%
4	Pengguna aplikasi tidak membutuhkan waktu lama untuk dipahami	1	17	10	1	1	88	150	58,66%
5	Rute yang diberikan oleh aplikasi terasa akurat	-	1	6	22	1	67	150	44,66%
6	Waktu pencarian rute pada aplikasi terasa cepat	-	14	15	1	-	103	150	68,66%
7	Aplikasi tidak mengalami lag Ketika menampilkan rute	1	14	14	1	-	105	150	70%
8	Aplikasi menampilkan gym yang	1	2	5	22	-	72	150	48%

1. Rumus skor per pertanyaan untuk questioner sebelum diimplementasikan Total = (Jumlah SS x 5) + (Jumlah S x 4) + (Jumlah C x 3) + (Jumlah TS x 2) + (Jumlah STS x 1)
 2. Rumus skor maksimal
Skor Max = (Jumlah Responden x 5)
 3. Rumus persentase (%) per pertanyaan Presentase skor = (Skor Total/Skor Max) x 100%
 4. Rumus skor total pertanyaan Total skor seluruh = (Total Q1 + Total Q2 + Total Q3 + Total Q4 + Total Q5 + Total Q6 + Total Q7 + Total Q8 + Total Q9 + Total Q10) Diperoleh Nilai Maksimal 1500
 5. Rumus rata-rata presentase Rata-rata persentase = (Total skor seluruh/total skor maksimal) x 100%
Rata-rata persentase = (945/1500) x 100% = 63%
- Jadi, rata-rata nilai keseluruhan adalah 63% dan termasuk ke dalam kategori Cukup. Hal ini menunjukkan bahwa aplikasi OMG belum memenuhi ekspektasi pengguna dalam hal kemudahan penggunaan, informasi yang disajikan, tampilan, dan fitur pencarian.

Table 11 Kuesioner Sesudah Implementasi Algoritma A Star

No	Pernyataan	SS	S	C	TS	STS	Skor Total	Nilai Max	Presentase
1	Aplikasi OnMyGym mudah digunakan untuk mencari gym terdekat	2	28	-	-	-	122	150	81,33%
2	Tampilan antarmuka aplikasi jelas dan mudah dipahami	4	26	-	-	-	124	150	82,66%
3	Informasi gym yang ditampilkan aplikasi mudah untuk dibaca dan diakses	1	27	2	-	-	119	150	79,33%
4	Pengguna aplikasi tidak membutuhkan waktu lama untuk dipahami	2	28	-	-	-	122	150	81,33%
5	Rute yang diberikan oleh aplikasi terasa akurat	1	22	7	-	-	114	150	76%
6	Waktu pencarian rute pada aplikasi terasa cepat	1	24	5	-	-	116	150	77,33%
7	Aplikasi tidak mengalami lag Ketika menampilkan rute	-	24	6	-	-	114	150	76%
8	Aplikasi menampilkan gym yang relevan	-	30	-	-	-	120	150	80%

1. Rumus skor per pertanyaan untuk questioner sesudah diimplementasikan $\text{Total} = (\text{Jumlah SS} \times 5) + (\text{Jumlah S} \times 4) + (\text{Jumlah C} \times 3) + (\text{Jumlah TS} \times 2) + (\text{Jumlah STS} \times 1)$
2. Rumus skor maksimal $\text{Skor Max} = (\text{Jumlah Responden} \times 5)$
3. Rumus persentase (%) per pertanyaan $\text{Presentase skor} = (\text{Skor Total} / \text{Skor Max}) \times 100\%$
4. Rumus skor total pertanyaan $\text{Total skor seluruh} = (\text{Total Q1} + \text{Total Q2} + \text{Total Q3} + \text{Total Q4} + \text{Total Q5} + \text{Total Q6} + \text{Total Q7} + \text{Total Q8} + \text{Total Q9} + \text{Total Q10})$ Diperoleh Nilai Maksimal 1500
5. Rumus rata-rata presentase $\text{Rata-rata persentase} = (\text{Total skor seluruh} / \text{total skor maksimal}) \times 100\%$
 $\text{Rata-rata persentase} = (1194 / 1500) \times 100\% = 79,6\%$

Jadi, rata-rata nilai keseluruhan adalah 79,6% dan termasuk ke dalam kategori Baik. Hal ini menunjukkan bahwa aplikasi OMG telah memenuhi ekspektasi pengguna dalam hal kemudahan penggunaan, informasi yang disajikan, tampilan, dan fitur pencarian menggunakan algoritma a star.

Berdasarkan hasil analisis yang dilakukan, respons pengguna terhadap aplikasi OnMyGym setelah diimplementasikan algoritma A menunjukkan persentase sebesar 79,6%, yang termasuk dalam kategori baik dibanding sebelum diimplementasikan algoritma A Star.

4. Kesimpulan

Dapat disimpulkan bahwa pengembangan sistem pencarian *gym* pada aplikasi *OnMyGym* berhasil diimplementasikan dan mampu menghasilkan rekomendasi yang lebih relevan dengan mempertimbangkan kedekatan lokasi pengguna sehingga mengurangi kebutuhan pencarian ulang atau penyaringan manual. Integrasi algoritma *A Star* dengan bobot jarak berbasis *formula Haversine* dan data jaringan jalan nyata dari *OpenStreetMap* yang diproses menggunakan *osm2pgroute* membentuk *graf* jalan yang representatif sehingga pencarian rute dan pemeringkatan *gym* berdasarkan jarak berjalan secara akurat. Dalam studi kasus yang diuji, sistem menghitung jarak total sekitar 324.6963 meter melalui 11 *edge* pada *graf* OSM dari *node* sumber ke *node* tujuan, menandakan presisi perhitungan jarak *geometris* pada implementasi ini. Secara keseluruhan, kombinasi *A Star*, *Haversine*, *OpenStreetMap*, dan *osm2pgroute* terbukti menjadi solusi yang efektif dan efisien untuk menyajikan hasil pencarian *gym* terdekat dalam aplikasi *OnMyGym*,



meskipun kinerja akhir bergantung pada bagaimana data peta diproses dan dimana beban komputasi ditempatkan.

Referensi

- Annisa. (2025). *Haversine Formula: Menghitung Jarak Akurat Antar Lokasi di Permukaan Bumi*. Retrieved from fikti.umsu: <https://fikti.umsu.ac.id/haversine-formula-menghitung-jarak-akurat-antar-lokasi-di-permukaan-bumi/>
- Aslami, J. A. (2015). IMPLEMENTASI ALGORITMA A STAR (A STAR) SEBAGAI DASAR PERGERAKAN NPC (NON PLAYER CHARACTER) MENDEKATI PLAYER UNTUK MENINGKATKAN REALITAS GAME PEMBELAJARAN KOSAKATA BAHASA ARAB. *etheses.uin-malang*.
- Aswandi, L. M. (2023). Implementation of the A Star Heuristic Search Algorithm in Determining the Shortest Path. *International Journal of Computer Sciences and Mathematics Engineering*, 29-35.
- Azwar, S. (2013). Metode Penelitian. *etheses.uin-malang*.
- Dyah Ayu Megawaty, M. Y. (2020). Penerapan Algoritma A-Star Pada Aplikasi Pencarian Lokasi Foto Berbasis Android. *journalbalitbangdalamampung*, 39-52.
- Erno Sumantri, S. H. (2023). Penerapan Algoritma A STARStar Untuk Mencari Rute Terpendek Dari Kemayoran Ke Destinasi Monumen Nasional (MONAS). *Jurnal Sains dan Teknologi*, 673-680.
- Firmansyah, I. (2025, July 5). *Class Diagram Adalah: Pengertian, Fungsi, dan Komponen dalam Pemodelan UML*. Retrieved from softwareseni: <https://www.softwareseni.co.id/blog/class-diagram-adalah>
- Geograf. (2023, Oktober 23). *Pengertian Android: Definisi dan Penjelasan Lengkap Menurut Ahli*. Retrieved from geograf.id: <https://geograf.id/jelaskan/pengertian-android/>
- Hasrudy Tanjung, E. (2018). Pengaruh Pelatihan Kompetensi Lingkungan Kerja Terhadap Kinerja Pegawai. *Maneggio: Jurnal Ilmiah Magister Manajemen*.
- Hendrik. (2022). *Pengertian Produk: Klasifikasi, Jenis, dan Ciri-Ciri Produk yang Disukai Konsumen*. Retrieved from gramedia: <https://www.gramedia.com/literasi/pengertian-produk/>
- Kamus Besar Bahasa Indonesia (KBBI)*. (n.d.). Retrieved from Kamus Besar Bahasa Indonesia (KBBI): <https://kbbi.web.id/navigasi>
- Komputer, W. (2002). *Apa & Bagaimana E-Commerce*. Retrieved from onesearch.id: <https://onesearch.id/Record/IOS4100.slims-1254/Details>
- Kumar, R. (2024, November 7). *The A STAR Algorithm: A Complete Guide*. Retrieved from datacamp: <https://www.datacamp.com/tutorial/a-star-algorithm>
- Kusmeiyanto Feri Purna, M. V. (2022). Penerapan passage plan dalam sistem navigasi kapal serta pengaruh sistem navigasi ketika berada pada alur pelayaran Karang Jamuang, Surabaya. *Prosiding Seminar Bidang Nautika*.
- Maiyana, E. (2018). Pemanfaatan Android Dalam Perencanaan Aplikasi Kumpulan Doa. *Jurnal Sains dan Informatika*.
- Muhammad Farid, R. R. (2020).
- MUQHITARAHIMA. (2025, August 21). *Penggunaan Use Case Diagram dalam Pengembangan Sistem*. Retrieved from dac.telkomuniversity: <https://dac.telkomuniversity.ac.id/>
- onerosolutions. (2025, July 2). *Activity Diagram: Pengertian, Tujuan, Komponen, dan Cara Membuatnya*. Retrieved from onerosolutions: <https://onero.id/insight/detail/activity-diagram/>
- Prasatya. (2025, January 27). *Apa Itu Sequence Diagram, Pengertian, Fungsi, dan Cara Membuatnya*. Retrieved from CODEPOLITAN: <https://www.codepolitan.com/blog/sequence-diagram-adalah/>
- Prof.Dr.Sugiyono. (2013). *Metode Penelitian Kuantitatif Kualitatif dan R&D*. Bandung: Alfabeta Bandung.
- Simeramisna, B. S. (2023, Juni 1). Optimasi Algoritma Pencarian Dalam Aplikasi E-Commerce Untuk Pengalaman Pengguna Yang Lebih Baik. *jurnalkomputerindonesia*, 19-24.
- Sukroni, A. F. (2025). Modifikasai Algoritma A-Star Pada Optimalisasi Rute Terpendek. *etheses-uin-malang*.
- Trivusi. (2023, January 20). *Algoritma A STAR (A Star): Pengertian, Cara Kerja, dan Kegunaannya*. Retrieved from trivusi: <https://www.trivusi.web.id/2023/01/algoritma-a-star.html>
- Visual Studio Code*. (2025). Retrieved from Visual Studio Code: <https://code.visualstudio.com/>
- Woltmann, S. (2024, November 27). *Implementing a Priority Queue Using a Heap*. Retrieved from HappyCoders.eu: <https://www.happycoders.eu/algorithms/implement-priority-queue-using-heap/>
- Yusuf, Y. (2024, December 16). *Apa Itu MySQL*. Retrieved from bif.telkomuniversity: <https://bif.telkomuniversity.ac.id/apa-itu-mysql/>
- Zainuddin Iba, A. (2024, July). *Landasan Teori, Kerangka Pemikiran, Penelitian Terdahulu, & Hipotesis*. Retrieved from ResearchGate: https://www.researchgate.net/publication/382060756_Landasan_Teori_Kerangka_Pemikiran_Penelitian_Terdahulu_Hipotesis

