

**RANCANG BANGUN APLIKASI CARI JODOH MENGGUNAKAN
ALGORITMA HAVERSINE BERBASIS ANDROID**

Studi Kasus Desa Sukawera Kecamatan Ligung Kabupaten Majalengka

SKRIPSI

Diajukan untuk memenuhi sebagian dari syarat memperoleh gelar Sarjana
Komputer Program Studi Teknik Informatika



Oleh :

ADE TRIO PURWANTO

0402201021

FAKULTAS ILMU KOMPUTER

UNIVERSITAS NAHDLATUL ULAMA CIREBON

2024 M/1446 H

UNIVERSITAS NAHDLATUL ULAMA

CIREBON



PENGESAHAN STATUS SKRIPSI

JUDUL : RANCANG BANGUN APLIKASI CARI JODOH
MENGUNAKAN ALGORITMA HAVERSINE BERBASIS
ANDROID

SAYA : ADE TRIO PURWANTO

NIM : 0402201021

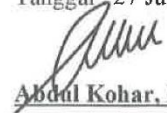
Mengijinkan Skripsi Sarjana Komputer ini disimpan di perpustakaan Universitas Nahdlatul Ulama dengan syarat-syarat kegunaan sebagai berikut :

1. Skripsi adalah hak milik Universitas Nahdlatul Ulama Cirebon
2. Perpustakaan Universitas Nahdlatul Ulama Cirebon dibenarkan membuat salinan untuk referensi saja.
3. Perpustakaan juga dibenarkan membuat salinan Skripsi ini sebagai bahan pertukaran antara institusi pendidikan tinggi
4. Berikan tanda $\checkmark$ pada kategori Skripsi
 - ☐ Sangat rahasia (Mengandung isi tentang keselamatan atau kepentingan negara Indonesia)
 - ☐ Rahasia (Mengandung isi tentang kerahasiaan dari sesuatu organisasi / badan dimana penelitian Skripsi ini dikerjakan)
 - ☒ Biasa

Disahkan oleh


Dicky Andika Sulaeman, S.T., M.Kom

Tanggal : 27 Juli 2024


Abdul Kohar, M.Kom

Tanggal : 27 Juli 2024

**UNIVERSITAS NAHDLATUL ULAMA
CIREBON**



PERSETUJUAN SKRIPSI

JUDUL : RANCANG BANGUN APLIKASI CARI JODOH
MENGGUNAKAN ALGORITMA HAVERSINE BERBASIS
ANDROID

SAYA : ADE TRIO PURWANTO

NIM : 0402201021

Skripsi ini telah diperiksa dan disetujui,
Cirebon, 27 Juli 2024

Pembimbing I

Dicky Andika Sulaeman, S.T., M.Kom
NIDN. 040906940

Pembimbing 2

Abdul Kohar, M.Kom
NIDN. 0412068704

Mengetahui,
Ketua Program Studi Teknik Informatika

Dicky Andika Sulaeman, S.T., M.Kom
NIDN. 040906940

**UNIVERSITAS NAHDLATUL ULAMA
CIREBON**



PENGESAHAN SKRIPSI

JUDUL : RANCANG BANGUN APLIKASI CARI JODOH
MENGUNAKAN ALGORITMA HAVERSINE BERBASIS
ANDROID

SAYA : ADE TRIO PURWANTO

NIM : 0402201021

Skripsi ini telah diujikan dan dipertahankan dihadapan Dewan pengujian pada sidang skripsi tanggal 24 Agustus 2024 . Menurut pandangan kami, Skripsi ini memadai untuk penganugerahan gelar Sarjana Komputer (S.Kom)

	Tanggal	Tanda Tangan
Pembimbing I <u>Dicky Andika Sulaeman, S.T., M.Kom</u> NIDN. 040906940	07 - 09 - 2024	
Pembimbing II <u>Abdul Kohar, M. Kom</u> NIDN. 0412068704	07 - 09 - 2024	
Penguji I <u>Rosidin, S. Kom., M. Kom</u> NIDN. 0415028303	07 - 09 - 2024	
Penguji II <u>H. Sukarsa, M.Kom</u> NIDN. 0418117605	07 - 09 - 2024	

Mengetahui,
Dekan Fakultas Ilmu Komputer
Universitas Nahdlatul Ulama Cirebon



Rosidin, S. Kom., M. Kom
NIDN. 0415028303

SURAT PERNYATAAN

Yang bertanda tangan dibawah ini :

Nama : Ade Trio Purwanto
NIM : 0402201021
Tempat, Tanggal Lahir : Majalengka, 02 Agustus 2003
Program Studi : Teknik Informatika
Perguruan Tinggi : Universitas Nahdlatul Ulama Cirebon

Menyatakan bahwa skripsi dengan judul sebagai berikut :

Judul Bahasa Indonesia : Rancang Bangun Aplikasi Cari Jodoh Menggunakan Algoritma Haversine Berbasis Android
Judul Bahasa Inggris : *Design and Development of a Matchmaking Application Using the Haversine Algorithm Based on Android*
Pembimbing I : Dicky Andika Sulaeman, S.T., M.Kom
Pembimbing II : Abdul Kohar, M.Kom

Adalah benar-benar **ASLI** dan **BELUM PERNAH** dibuat orang lain, kecuali yang dikembangkan dan diacu dalam Daftar Pustaka pada Skripsi/Tugas Akhir ini.

Demikian pernyataan ini **SAYA** buat, apabila dikemudian hari terbukti **SAYA** melakukan penjiplakan karya orang lain, maka **SAYA BERSEDIA** menerima **SANKSI AKADEMIK**.

Cirebon, 27 Juli 2024

Yang menyatakan



Ade Trio Purwanto
NIM. 040220121

MOTTO DAN PERSEMBAHAN

MOTO

“Kesuksesan bukanlah kunci kebahagiaan. Kebahagiaan adalah kunci kesuksesan. Jika Anda mencintai apa yang Anda lakukan, Anda akan sukses.”

-Ade Trio Purwanto-

PERSEMBAHAN

Alhamdulillahirabbil'aalamiin, segala puji dan syukur saya panjatkan kepada Allah SWT atas rahmat dan ridho-Nya. Karya ini saya persembahkan dengan penuh cinta kepada kedua orang tua saya yang sangat saya cintai, Bapak dan Ibu, yang selalu memberikan kasih sayang, perhatian, dan dukungan tanpa henti. Terima kasih yang sebesar-besarnya untuk semua keluarga dan teman-teman yang selalu mendoakan dan mendukung saya dalam segala hal. Ucapan terima kasih yang tulus juga saya sampaikan kepada para dosen Fakultas Ilmu Komputer di Universitas Nahdlatul Ulama Cirebon yang telah memberikan ilmu yang sangat berharga selama masa studi saya hingga saya dapat menyelesaikan skripsi ini.

KATA PENGANTAR



Assalamu'alaikum Wr.Wb.

Segala puji dan syukur kehadiran Allah SWT yang telah memberikan nikmat kesehatan, keimanan, dan rahmat-Nya sehingga penulis dapat menyelesaikan skripsi ini dengan baik. Shalawat serta salam semoga tetap tercurahkan kepada junjungan kita, Nabi Muhammad SAW. Amin.

Penulis menyadari bahwa tanpa bimbingan dan dukungan dari berbagai pihak, terutama dosen pembimbing I dan II yang telah memberikan arahan dan bantuan selama penulisan skripsi ini, skripsi ini tidak akan dapat terselesaikan dengan baik. Oleh karena itu, penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Dr. Agus Sugiarto, S.H., M.H., M.M., selaku Rektor Universitas Nahdlatul Ulama Cirebon
2. Bapak Rosidin, S.Kom., M.Kom., selaku Dekan Fakultas Ilmu Komputer
3. Bapak Dicky Andika Sulaeman, S.T., M.Kom., selaku Ketua Program Studi Teknik Informatika
4. Bapak Dicky Andika Sulaeman, S.T., M.Kom., selaku Pembimbing I
5. Bapak Abdul Kohar, M.Kom., selaku Pembimbing II
6. Kedua orang tua saya, Bapak Warji dan Ibu kastiem yang selalu memberikan doa dan dukungan semangat sehingga saya dapat mencapai tahap ini.
7. Pimpinan, dosen, serta seluruh staf Fakultas Ilmu Komputer Universitas Nahdlatul Ulama Cirebon.
8. Rekan-rekan mahasiswa Program Studi Teknik Informatika FILKOM - UNU Cirebon angkatan tahun akademik 2020.

Tidak lupa, penulis juga menyampaikan terima kasih kepada semua pihak yang tidak dapat disebutkan satu per satu, yang telah memberikan dukungan dan bantuan hingga skripsi ini dapat terselesaikan. Semoga Allah SWT membalasnya dengan pahala yang berlipat ganda. Amin.

Penulis menyadari bahwa dalam penulisan skripsi ini masih terdapat banyak kekurangan. Oleh karena itu, kritik dan saran yang bersifat membangun sangat diharapkan demi kesempurnaan skripsi ini. Semoga skripsi ini dapat bermanfaat bagi penulis khususnya, dan para pembaca pada umumnya. Terima kasih.

Cirebon, 27 Juli 2024

Penulis

Ade Trio Purwanto

DAFTAR ISI

HALAMAN JUDUL.....	i
PENGESAHAN STATUS SKRIPSI	ii
PERSETUJUAN SKRIPSI	iii
PENGESAHAN SKRIPSI	iv
SURAT PERNYATAAN.....	v
MOTTO DAN PERSEMBAHAN	vi
KATA PENGANTAR	vii
DAFTAR ISI.....	ix
DAFTAR TABEL	xiii
DAFTAR GAMBAR	xiv
ABSTRAK	xvi
<i>ABSTRACT</i>	xvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan dan Manfaat.....	2
1.4.1 Tujuan Penelitian	2
1.4.2 Manfaat Penelitian	3
1.5 Review Penelitian.....	3
1.6 Metode dan Teknik Penelitian.....	8
1.6.1 Metode Penelitian.....	8
1.6.2 Teknik Penelitian	9
1.7 Sistematika Penulisan.....	10
1.8 Kerangka Berpikir	11
BAB II LANDASAN TEORI	14
2.1 Pandangan Islam terhadap Pernikahan dan Teknologi.....	14

2.1.1	Pandangan Islam terhadap Pernikahan.....	14
2.1.2	Pandangan Islam terhadap Teknologi	15
2.1.3	Urgensi Rancang Bangun Aplikasi Cari Jodoh.....	16
2.2	Teknologi Android	17
2.2.1	Pengertian Android	17
2.2.2	Sejarah Android	17
2.2.3	Versi Android.....	17
2.3	Algoritma Haversine	20
2.3.1	Definisi Algoritma Haversine	20
2.3.2	Prinsip Kerja Algoritma Haversine.....	21
2.3.3	Implementasi Algoritma Haversine dalam Pemrograman	22
2.4	Algoritma Vincenty	23
2.4.1	Definisi Algoritma Vincenty.....	23
2.4.2	Prinsip Kerja Algoritma Vincenty	24
2.4.3	Implementasi Algoritma Vincenty dalam Pemrograman.....	26
2.5	Android Studio	28
2.5.1	Pengertian Android Studio.....	28
2.5.2	Spesifikasi Minimum Sistem	28
2.5.3	Fitur Utama Android Studio	29
2.6	<i>Blackbox Testing</i>	31
2.6.1	Pengertian <i>Blackbox Testing</i>	31
2.6.2	Teknik <i>Blackbox Testing</i>	31
2.7	<i>Framework</i> CodeIgniter	32
2.7.1	Pengertian CodeIgniter.....	32
2.7.2	Keunggulan CodeIgniter	34

2.8	Figma.....	36
2.8.1	Pengertian Figma	36
2.8.2	Fungsi Figma.....	36
2.9	Arsitektur Pengembangan Aplikasi.....	38
2.9.1	<i>Frontend Development</i>	38
2.9.2	<i>Backend Development</i>	38
BAB III ANALISA DAN PERANCANGAN SISTEM.....		39
3.1	Tempat dan Waktu Penelitian	39
3.1.1	Tempat Penelitian.....	39
3.1.2	Waktu Penelitian	39
3.2	Profil Desa Sukawera	39
3.2.1	Sejarah Desa Sukawera	39
3.2.2	Karakteristik Demografis Desa Sukawera	40
3.3	Analisis Masalah	41
3.4	Analisis Kebutuhan	42
3.4.1	Perangkat Keras (<i>Hardware</i>)	42
3.4.2	Perangkat Lunak (<i>Software</i>).....	42
3.5	<i>Flowchart</i> Perancangan Sistem Kerja	42
3.6	Perancangan Sistem.....	45
3.7	Perancangan <i>Database</i>	47
3.7.1	Tabel.....	47
3.7.2	Kamus Data.....	47
3.8	Perancangan <i>Interface</i> (Antarmuka).....	49
3.8.1	Tampilan Splashscreen.....	50
3.8.2	Tampilan Intro Aplikasi	50

3.8.3	Tampilan Login Aplikasi	51
3.8.4	Tampilan Daftar Aplikasi.....	52
3.8.5	Tampilan <i>Dashboard</i>	54
3.8.6	Tampilan Pencarian.....	56
3.8.7	Tampilan Profil	56
3.8.8	Tampilan <i>Map</i>	57
3.8.9	Tampilan Detail Pengguna	58
3.8.10	Tampilan Menu Pesan.....	60
3.8.11	Tampilan Menu Suka	61
3.9	Skenario Pengujian.....	61
BAB IV IMPLEMENTASI DAN ANALISA HASIL UJI COBA.....		65
4.1	Implementasi Sistem	65
4.1.1	Implementasi Tampilan <i>Interface</i>	65
4.1.2	Implementasi Algoritma Haversine	75
4.1.3	Implementasi Filter Jarak.....	76
4.2	Pengujian Sistem	78
4.2.1	Pengujian <i>Blackbox</i>	78
4.2.2	Pengujian Akurasi Jarak.....	84
4.2.3	Pengujian Filter Jarak.....	92
BAB V PENUTUP.....		96
5.1	Kesimpulan.....	96
5.2	Saran	96
DAFTAR PUSTAKA		98

DAFTAR TABEL

Tabel 1. 1 Review Penelitian	4
Tabel 1. 2 Kerangka Berpikir.....	13
Tabel 2. 1 Versi Android.....	18
Tabel 2. 2 Spesifikasi Minimum Android Studio	29
Tabel 3. 1 Waktu Penelitian	39
Tabel 3. 2 Daftar Kepala Desa Sukawera	40
Tabel 3. 3 Perbatasan Desa Sukawera.....	41
Tabel 3. 4 Jumlah Penduduk Desa Sukawera	41
Tabel 3. 5 Tabel tbl_user.....	48
Tabel 3. 6 Tabel tbl_likes.....	48
Tabel 3. 7 Tabel tbl_chat.....	49
Tabel 3. 8 Daftar Skenario Pengujian	61
Tabel 4. 1 Perangkat Pengujian Blackbox	78
Tabel 4. 2 Hasil Pengujian Blackbox.....	79
Tabel 4. 3 Pengujian Akurasi Perempuan	84
Tabel 4. 4 Pengujian Akurasi Laki-laki	88
Tabel 4. 5 Pengujian Filter Jarak Perempuan	93
Tabel 4. 6 Pengujian Filter Jarak Laki-laki	94

DAFTAR GAMBAR

Gambar 1. 1 Diagram Rapid And Development.....	9
Gambar 1. 2 Kerangka Berpikir	12
Gambar 2. 1 Logo Android Studio.....	28
Gambar 2. 2 Logo CodeIgniter	33
Gambar 3. 1 Flowchart Sistem Kerja	43
Gambar 3. 2 Use Case Aplikasi	45
Gambar 3. 3 Relasi Antar Tabel.....	47
Gambar 3. 4 Rancangan Tampilan Splashscreen	50
Gambar 3. 5 Rancangan Tampilan Intro	51
Gambar 3. 6 Rancangan Tampilan Login	51
Gambar 3. 7 Rancangan Tampilan Daftar.....	52
Gambar 3. 8 Rancangan Tampilan Dashboard	54
Gambar 3. 9 Rancangan Tampilan Pencarian	56
Gambar 3. 10 Rancangan Tampilan Profil.....	56
Gambar 3. 11 Rancangan Tampilan Map	57
Gambar 3. 12 Rancangan Tampilan Detail Akun	58
Gambar 3. 13 Rancangan Tampilan Menu Pesan	60
Gambar 3. 14 Rancangan Tampilan Menu Suka	61
Gambar 4. 1 Tampilan Splashscreen.....	65
Gambar 4. 2 Tampilan Intro.....	65
Gambar 4. 3 Tampilan Login	66
Gambar 4. 4 Tampilan Daftar	67
Gambar 4. 5 Tampilan Reset Sandi.....	68
Gambar 4. 6 Tampilan Dashboard	68
Gambar 4. 7 Tampilan Pencarian.....	69
Gambar 4. 8 Tampilan Menu Suka	70
Gambar 4. 9 Tampilan Pesan	70
Gambar 4. 10 Tampilan Menu Profil	71
Gambar 4. 11 Tampilan Detail Akun.....	72
Gambar 4. 12 Tampilan Map	73

Gambar 4. 13 Tampilan Ubah Data	73
Gambar 4. 14 Tampilan Tentang	74
Gambar 4. 15 Tampilan Keluar Aplikasi	75

ABSTRAK

Perkembangan teknologi telah membawa perubahan signifikan dalam berbagai aspek kehidupan, termasuk dalam cara orang mencari pasangan hidup. Penurunan angka pernikahan di Indonesia, seperti yang dilaporkan oleh BPS, menunjukkan adanya tantangan dalam menemukan pasangan hidup di tengah gaya hidup yang padat. Penelitian ini bertujuan untuk mengembangkan aplikasi cari jodoh berbasis Android dengan menerapkan algoritma Haversine untuk menghitung jarak antar pengguna berdasarkan koordinat geografis. Algoritma ini diharapkan dapat membantu individu menemukan calon pasangan yang berada dalam jarak yang relatif dekat, sehingga mempermudah pencarian pasangan di tengah kesibukan sehari-hari.

Metode penelitian yang digunakan adalah *Research and Development* (R&D) dengan model *Rapid And Development* (RAD). Penelitian ini melibatkan analisis kebutuhan pengguna, desain sistem, implementasi aplikasi dan pengujian fungsional serta akurasi. Evaluasi dilakukan untuk memastikan aplikasi berfungsi dengan baik dalam menentukan jarak antar pengguna.

Hasil penelitian menunjukkan bahwa aplikasi yang dikembangkan dapat mempermudah individu dalam mencari pasangan hidup, dan memberikan kontribusi positif terhadap peningkatan angka pernikahan serta perbaikan struktur sosial di komunitas. Penelitian ini juga memberikan manfaat pendidikan, sosial, dan pengembangan keterampilan teknis bagi penulis.

Kata kunci: Aplikasi Cari Jodoh, Algoritma Haversine, Android, Jarak Geografis, Rancang Bangun Aplikasi

ABSTRACT

Technological advancements have brought significant changes to various aspects of life, including the way people search for a life partner. The decline in marriage rates in Indonesia, as reported by BPS, indicates challenges in finding a partner amid busy lifestyles. This research aims to develop an Android-based dating application using the Haversine algorithm to calculate the distance between users based on geographic coordinates. The algorithm is expected to help individuals find potential partners who are relatively close in distance, thereby facilitating partner search amid daily busyness.

The research method used is Research and Development (R&D) with the Rapid Application Development (RAD) model. This study involves analyzing user needs, system design, application implementation, and functional as well as accuracy testing. Evaluation is conducted to ensure the application functions effectively in determining distances between users.

The results indicate that the developed application can ease the process of finding a life partner and positively contribute to increasing marriage rates and improving social structure within communities. The research also provides educational, social, and technical skill development benefits for the author.

Keywords: *Dating Application, Haversine Algorithm, Android, Geographic Distance, Application Development*

BAB I

PENDAHULUAN

1.1 Latar Belakang

Penggunaan teknologi di Indonesia semakin meluas dan telah mengubah berbagai aspek kehidupan, termasuk cara orang dalam mencari pasangan hidup. Pasangan hidup adalah seseorang yang dipilih untuk menjadi pendamping dalam kehidupan. Menurut laporan terbaru dari BPS yang berjudul Statistik Indonesia 2024, angka pernikahan di Indonesia terus mengalami penurunan dan mencapai titik terendah. Contohnya, pada tahun 2023, jumlah pernikahan di seluruh Indonesia mencapai 1.577.255, turun sekitar 128 ribu dibandingkan dengan tahun 2022. Namun, ini hanya merupakan perbandingan satu tahun terakhir. Jika kita melihat seluruh laporan BPS dalam kurun waktu 10 tahun terakhir, angka pernikahan di Indonesia mengalami penurunan sebesar 28,63 persen atau menyusut sebanyak 632.791. (Badan Pusat Statistik, 2024)

Salah satu penyebab utama dari penurunan angka pernikahan adalah gaya hidup yang padat. Banyak individu yang mengalami kesulitan dalam menemukan waktu untuk berinteraksi sosial secara langsung karena tuntutan pekerjaan dan aktivitas sehari-hari yang padat. Hal ini berdampak pada berkurangnya peluang untuk bertemu dengan calon pasangan potensial.

Berdasarkan wawancara dengan Sekretaris Desa Sukawera, Kecamatan Ligung, Kabupaten Majalengka, banyak warga desa yang lebih fokus pada karir sehingga menunda atau bahkan mengabaikan pernikahan. Menurut Sekdes, fenomena ini tidak hanya berdampak pada individu itu sendiri tetapi juga pada struktur sosial dan demografi di desa tersebut.

Untuk menjawab tantangan ini, rancang bangun aplikasi cari jodoh berbasis android menggunakan algoritma haversine diusulkan sebagai solusi inovatif. Algoritma haversine akan digunakan untuk menghitung jarak antara pengguna berdasarkan koordinat geografis, sehingga memungkinkan pengguna untuk menemukan calon pasangan yang berada dalam jarak yang relatif dekat.

Dengan adanya aplikasi ini, diharapkan dapat mempermudah individu dalam mencari dan menemukan pasangan hidup, meskipun mereka memiliki kesibukan yang padat.

Dengan solusi ini, diharapkan aplikasi ini dapat memberikan kontribusi positif dalam meningkatkan angka pernikahan, tidak hanya di Desa Sukawera, tetapi juga di berbagai daerah lain di Indonesia yang menghadapi permasalahan serupa.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah disampaikan diatas, penulis dapat merumuskan permasalahan sebagai berikut:

1. Bagaimana mengimplementasikan algoritma haversine pada aplikasi cari jodoh berbasis android untuk menghitung jarak antar pengguna?
2. Bagaimana hasil uji fungsional dan uji akurasi algoritma haversine pada aplikasi cari jodoh?

1.3 Batasan Masalah

Supaya pembahasan masalah yang dilakukan tidak menyimpang dari pokok permasalahan, maka permasalahan yang akan dibahas dibatasi sebagai berikut:

1. Aplikasi ini hanya dapat dijalankan pada sistem operasi android
2. Aplikasi ini hanya dapat berfungsi dalam mode online
3. Penelitian ini membandingkan dua algoritma, yaitu algoritma haversine dan vincenty dalam menentukan jarak terpendek antar pengguna aplikasi

1.4 Tujuan dan Manfaat

1.4.1 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah sebagai berikut:

1. Mengimplementasikan algoritma haversine pada aplikasi cari jodoh berbasis Android untuk menghitung jarak antara pengguna
2. Menguji fungsional dan akurasi algoritma haversine

1.4.2 Manfaat Penelitian

Adapun manfaat penelitian ini adalah sebagai berikut:

1. Manfaat untuk Pendidikan

Penelitian ini memberikan referensi yang berguna bagi pengembangan teknologi informasi, khususnya dalam penerapan algoritma Haversine dalam aplikasi berbasis Android. Mahasiswa dan peneliti dapat memanfaatkan hasil penelitian ini untuk memperdalam pemahaman mereka tentang implementasi algoritma dalam pemrograman Android.

2. Manfaat untuk Masyarakat

Aplikasi ini diharapkan dapat membantu individu yang memiliki gaya hidup sibuk untuk menemukan pasangan hidup dengan lebih mudah. Dengan demikian, penelitian ini dapat memberikan kontribusi dalam meningkatkan angka pernikahan dan memperbaiki struktur sosial di komunitas, khususnya di Desa Sukawera dan daerah lainnya di Indonesia.

3. Manfaat untuk Penulis

Melalui penelitian ini, penulis dapat mengembangkan keterampilan teknis dalam pemrograman android dan implementasi algoritma. Pengalaman ini akan sangat berharga dalam pengembangan karir penulis sebagai pengembang aplikasi di masa depan.

1.5 Review Penelitian

Dalam penelitian ini penulis memaparkan lima penelitian terdahulu yang relevan dengan permasalahan yang akan diteliti tentang “Rancang Bangun Aplikasi Cari Jodoh Menggunakan Algoritma Haversine Berbasis Android”. Dan berikut adalah lima penelitian terdahulu yang menjadi bahan acuan dalam mengembangkan aplikasi ini:

Tabel 1. 1 Review Penelitian

No	Judul /Peneliti /Tahun	Permasalahan	Solusi	Metode	Hasil
1	Penerapan Metode Haversine Formula Pada Aplikasi Jemput Sampah Masyarakat / Ari Amir Alkodri, Burham Isnanto, Agus Dendi, Ferry Bakti dan Erli / 2023	Penggunaan metode konvensional dalam pemrosesan data dan komunikasi yang menyulitkan interaksi dan pengelolaan layanan pengambilan sampah	Pengembangan aplikasi mobile berbasis Android menggunakan Haversine Formula untuk menemukan lokasi penjemputan sampah terdekat.	Research & Development	Penelitian ini berhasil membuat aplikasi mobile yang memberikan informasi penjemputan sampah kepada pengguna dan menampilkan lokasi penjemputan terdekat berdasarkan lokasi pengguna.
2	Penerapan Metode Haversine Formula Pada Pencarian Lokasi Fasilitas Kesehatan Terdekat /	Kesulitan dalam mengetahui informasi mengenai lokasi fasilitas kesehatan terdekat, terutama	Pengembangan aplikasi mobile yang menggunakan Haversine Formula untuk mencari dan menampilkan urutan fasilitas	Research & Development	Hasil penelitian menunjukkan bahwa pengujian fungsionalitas antarmuka aplikasi menggunakan

No	Judul /Peneliti /Tahun	Permasalahan	Solusi	Metode	Hasil
	Rahmi Hidayati dan Nurul Mutiah / 2022	dalam keadaan darurat.	kesehatan terdekat berdasarkan lokasi pengguna dan jenis fasilitas yang dipilih.		metode <i>black box</i> menghasilkan nilai 82,38% dari penilaian pengguna.
3	Penerapan Metode Haversine Formula Untuk Penentuan Titik Kumpul pada Aplikasi Tanggap Bencana / Abadi Nugroho, Rio Jumardi, Nur Fajariah Ramadhania / 2020	Tidak adanya media khusus untuk memberikan informasi yang akurat tentang bencana dan titik kumpul, serta penanganan bencana yang lambat karena perlu memverifikasi informasi terlebih dahulu.	Pengembangan aplikasi mobile menggunakan Haversine Formula dan Google Maps untuk menyediakan informasi bencana dan menemukan titik kumpul terdekat berdasarkan lokasi pengguna.	Research & Development	Hasil penelitian menunjukkan bahwa aplikasi yang dikembangkan dapat memberikan informasi tentang bencana dan mencari titik kumpul terdekat dengan sukses.

No	Judul /Peneliti /Tahun	Permasalahan	Solusi	Metode	Hasil
	Kajian Ketelitian Posisi Kiblat / Mochammad Akbar Maulana, Silvester Sari Sai dan Adkha Yuliananda / 2023	Penentuan arah kiblat yang umumnya menggunakan metode spherical trigonometry, dengan asumsi bumi berbentuk bola, yang mungkin tidak akurat karena bentuk bumi sebenarnya adalah elipsoid.	Menggunakan metode vincenty, yang mengasumsikan bentuk bumi sebagai elipsoid, untuk menghitung arah kiblat dengan hasil yang lebih akurat, serta menganalisis hasil tersebut dengan data arah kiblat dari metode spherical trigonometry sebagai alternatif penentuan arah kiblat.	Research & Development	Hasil penelitian menunjukkan bahwa penggunaan metode vincenty memberikan tingkat keakuratan yang baik dalam menentukan arah kiblat.
	Perbandingan Algoritma Vincenty Dan Hubeny Dalam Menentukan Jarak Terpendek Pada Rumah	Kesulitan dalam menentukan jarak antara dua titik antar rumah sakit di Lhokseumawe dengan akurasi dan	Merancang aplikasi berbasis Android yang menggunakan algoritma Vincenty dan Hubeny untuk menghitung jarak antar rumah sakit,	Research & Development	Perbedaan jarak antara algoritma vincenty dan hubeny kurang dari 1%, sementara hasil kedua algoritma

No	Judul /Peneliti /Tahun	Permasalahan	Solusi	Metode	Hasil
	Sakit di Lhokseumawe Berbasis Android / Ridwan Halim / 2024	efisiensi yang baik.	dan algoritma Ant Colony untuk menentukan jarak terpendek. Aplikasi ini diimplementasikan dengan framework Flutter dan library Dart (Google Maps).		tersebut dibandingkan dengan Google Maps bisa mencapai 30%. Algoritma Hubeny lebih efisien secara waktu, namun algoritma vincenty lebih akurat dengan perbedaan jarak terbaik sekitar 1,97 meter.
6	Menurut Penulis	Penelitian ini bertujuan mengembangkan aplikasi cari jodoh berbasis Android untuk Desa Sukawera guna mengatasi penurunan angka pernikahan akibat kesulitan dalam menemukan pasangan dekat lokasi. Aplikasi ini akan memanfaatkan algoritma Haversine untuk mengidentifikasi calon pasangan terdekat dari lokasi pengguna. Pengembangan dilakukan dengan metode Rapid Application Development (RAD), mencakup perencanaan, prototyping, pengujian, implementasi, dan peluncuran. Fitur utama aplikasi meliputi filter jarak terdekat, tampilan profil pengguna, dan verifikasi usia 18+. Pengujian fungsionalitas dan akurasi algoritma Haversine			

No	Judul /Peneliti /Tahun	Permasalahan	Solusi	Metode	Hasil
		akan dilakukan untuk memastikan aplikasi efektif dan sesuai dengan kebutuhan masyarakat Desa Sukawera.			

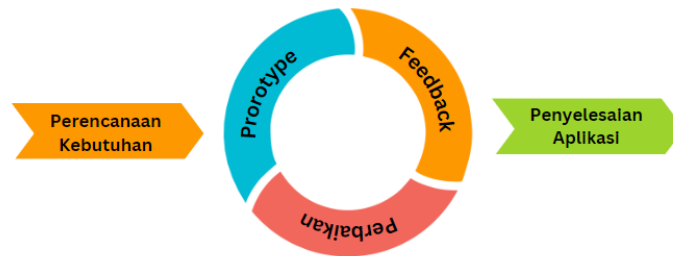
1.6 Metode dan Teknik Penelitian

1.6.1 Metode Penelitian

Dalam penelitian ini, penulis memilih untuk menggunakan metode penelitian *Research and Development* (R&D) dengan model *Rapid Android Development* (RAD). Pemilihan metode ini didasarkan pada tujuan utama penelitian, yaitu mengembangkan aplikasi cari jodoh yang efektif dan efisien, serta memastikan bahwa setiap tahap pengembangan dilakukan dengan teliti dan sistematis.

Research and Development (R&D) merupakan proses atau langkah-langkah untuk mengembangkan suatu produk baru atau menyempurnakan produk yang telah ada. R&D adalah metode yang berfokus pada inovasi dan pengembangan produk yang dapat memberikan solusi terhadap permasalahan yang ada. Metode ini sangat cocok digunakan dalam pengembangan perangkat lunak, termasuk aplikasi mobile, karena memungkinkan peneliti untuk secara iteratif mengembangkan dan menyempurnakan aplikasi berdasarkan temuan dan umpan balik yang diperoleh selama proses penelitian. (Okpatrioka, 2023)

Rapid Application Development (RAD) adalah sebuah metode pengembangan perangkat lunak yang menekankan efisiensi waktu, penghematan biaya, serta percepatan penyelesaian proyek. Metode ini memungkinkan tim pengembang untuk membuat prototipe aplikasi dengan cepat melalui iterasi yang berkelanjutan dalam waktu yang relatif singkat, sehingga kekurangan dalam aplikasi dapat segera diidentifikasi dan diperbaiki. (Faulina A. r., 2024)



Gambar 1. 1 Diagram *Rapid And Development*

Gambar di atas menggambarkan tahapan-tahapan dalam metode *Rapid Application Development* (RAD) yang digunakan dalam proses pengembangan aplikasi. Diagram ini menunjukkan proses iteratif dari RAD yang terdiri dari beberapa tahapan kunci:

- 1) Perencanaan Kebutuhan (*Requirements Planning*)
- 2) Desain Pengguna (*User Design*)
- 3) Pembangunan Cepat (*Rapid Construction*)
- 4) Penyelesaian (*Cutover*)

1.6.2 Teknik Penelitian

Untuk memahami lebih baik tentang aplikasi cari jodoh dan apa yang diinginkan penggunanya, teknik pengumpulan data yang digunakan penulis adalah dengan teknik kuesioner, studi literatur dan wawancara. Berikut adalah penjelasan poin-poin dari teknik pengumpulan data yang digunakan untuk memahami kebutuhan pengguna dalam perancangan aplikasi cari jodoh:

- 1) Kuesioner

Pada tahap ini, peneliti menyusun serangkaian pertanyaan dalam kuesioner yang dirancang untuk mengidentifikasi kebutuhan pengguna. Kuesioner ini memuat pertanyaan tentang penyebab menurunnya angka pernikahan di masyarakat dan apakah pembuatan aplikasi cari jodoh dapat menjadi solusi untuk masalah tersebut. Tujuannya adalah untuk mengumpulkan data dari pengguna potensial

mengenai konteks sosial yang mempengaruhi keputusan mereka terkait pernikahan.

2) Studi Literatur

Peneliti melakukan kajian terhadap berbagai sumber, seperti artikel dan jurnal, dengan fokus utama mencari algoritma yang berguna. Selain itu, peneliti juga mengidentifikasi elemen-elemen yang disukai oleh pengguna pada aplikasi-aplikasi tersebut dan melihat bagaimana aplikasi-aplikasi tersebut memenuhi kebutuhan penggunanya.

3) Wawancara

Wawancara dilakukan dengan Sekretaris Desa Sukawera untuk mendapatkan wawasan tentang bagaimana keadaan ekonomi, norma sosial, dan pandangan pribadi mempengaruhi keputusan untuk menikah atau menunda pernikahan.

1.7 Sistematika Penulisan

Tugas akhir ini disusun dalam lima bab dengan sistematika penulisan sebagai berikut:

BAB I PENDAHULUAN

Bab ini menjelaskan dasar-dasar yang menjadi alasan dilaksanakannya penelitian serta identifikasi masalah yang hendak dipecahkan. Bagian ini mencakup latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, dan manfaat penelitian.

BAB II LANDASAN TEORI

Bab ini menguraikan teori-teori dasar, penelitian terdahulu, kerangka pikir, dan hipotesis yang menjadi acuan dalam pelaksanaan penelitian tugas akhir ini.

BAB III RANCANGAN SISTEM INTERFACE

Bab ini membahas lingkungan pengembangan perangkat keras, alasan pemilihan *interface* yang digunakan, diagram atau rangkaian dasar perancangan *interface*, serta garis besar alur kerja *interface* yang dibangun.

BAB IV IMPLEMENTASI SISTEM DAN PENGUJIAN

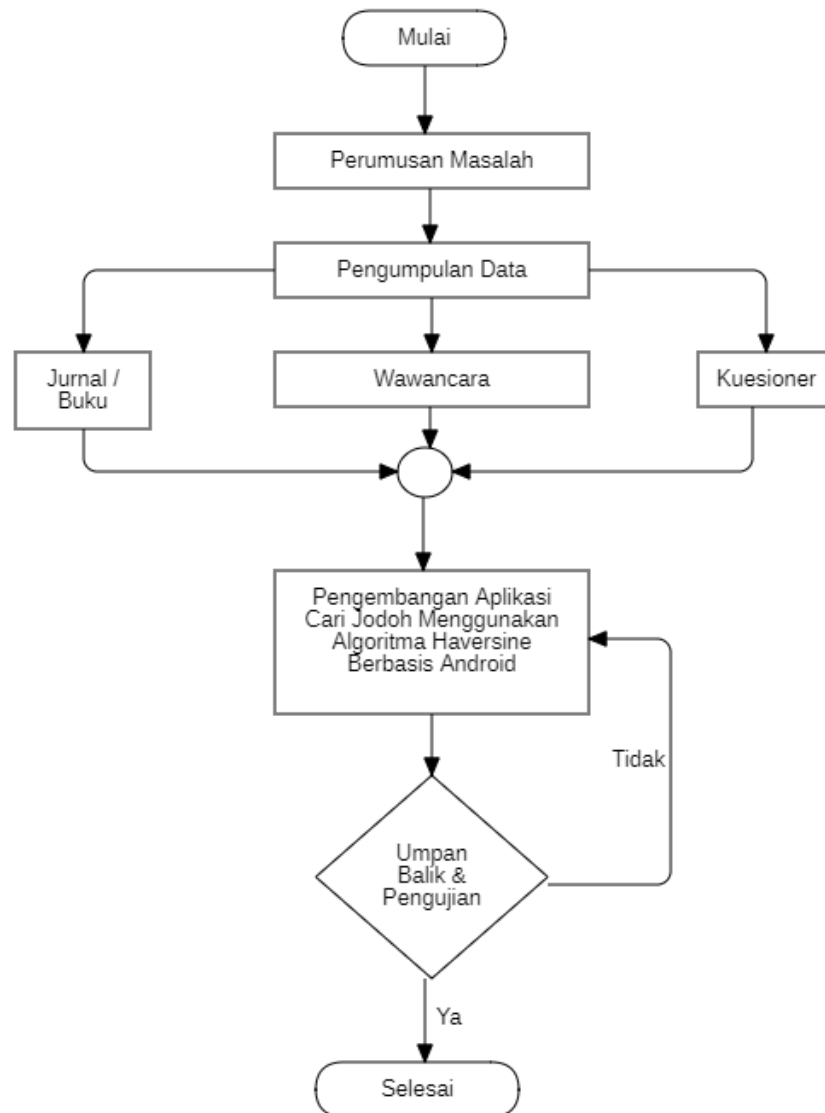
Bab ini menjelaskan hasil implementasi *interface* yang dikembangkan, pengujian melalui simulasi, pengujian dalam kondisi nyata, cara penggunaan *interface* yang telah dikembangkan, serta analisis hasil yang diperoleh dari *interface* tersebut.

BAB V PENUTUP

Bab ini menyajikan kesimpulan dan saran dari penulis terkait penelitian yang dilakukan dalam penulisan ilmiah ini.

1.8 Kerangka Berpikir

Berikut adalah gambaran dari kerangka berpikir pada penelitian ini yang ditunjukkan pada gambar berikut:



Gambar 1. 2 Kerangka Berpikir

Berikut adalah tahapan-tahapan dalam perancangan aplikasi cari jodoh menggunakan algoritma haversine berbasis android yang menggunakan model pengembangan *Rapid And Development* (RAD). Tahapan ini meliputi perumusan masalah, pengumpulan data, perancangan aplikasi, menerima masukan (*feedback*) dari responden hingga pengujian aplikasi. Penjelasan rinci dari setiap tahapannya peneliti sajikan dalam bentuk tabel di bawah ini:

Tabel 1. 2 Kerangka Berpikir

No	Tahapan	Deskripsi
1	Perumusan Masalah	Menurunnya angka pernikahan di Indonesia yang salah satu penyebabnya adalah gaya hidup yang padat.
2	Pengumpulan Data	Pengumpulan informasi dan data dari berbagai sumber.
	a. Jurnal dan Buku	Menganalisa Penelitian-penelitian terdahulu sebagai bahan referensi dan membaca buku yang memberikan informasi terkait pentingnya pernikahan.
	b. Wawancara	Melakukan wawancara dengan Sekretaris Desa Sukawera
	c. Kuesioner	Menyusun serangkaian pertanyaan dalam bentuk kuesioner yang dirancang untuk mengidentifikasi kebutuhan pengguna.
3	Perancangan Aplikasi	Merancang aplikasi sesuai dengan kebutuhan pengguna
4	<i>Feedback</i> & Pengujian	Menganalisis umpan balik (<i>feedback</i>) untuk mengidentifikasi masalah dan kekurangan Aplikasi serta Pengujian aplikasi untuk memastikan fungsionalitasnya bekerja dengan baik.

BAB II

LANDASAN TEORI

2.1 Pandangan Islam terhadap Pernikahan dan Teknologi

2.1.1 Pandangan Islam terhadap Pernikahan

Dalam Islam, pernikahan dianggap sebagai salah satu sunnah yang dianjurkan dan memiliki nilai ibadah. Pernikahan adalah cara untuk menjaga kehormatan, melanjutkan keturunan, dan memenuhi kebutuhan emosional serta biologis manusia. Pemilihan pasangan hidup dalam Islam didasarkan pada agama, akhlak, dan kecocokan. Seorang Muslim dianjurkan untuk memilih pasangan yang baik agamanya dan memiliki akhlak yang mulia, karena ini akan menjadi fondasi kehidupan rumah tangga yang harmonis dan penuh berkah.

أَرْبَعٌ مِنْ سُنَنِ الْمُرْسَلِينَ: الْحَيَاءُ، وَالتَّعَطُّرُ، وَالسَّوَاكُ، وَالنِّكَاحُ (رواه الترمذي)

“Ada empat perkara yang termasuk Sunnah para Rasul: rasa-malu, memakai wewangian, bersiwak, dan menikah” (HR. Tirmidzi)

Pernikahan merupakan salah satu cara untuk mencapai tujuan fundamental dari syariat Islam, yaitu menjaga keturunan. Melalui pernikahan, terbentuklah sarana penting yang membantu manusia menghindari perbuatan yang dilarang oleh Allah, seperti zina, homoseksualitas, dan tindakan serupa lainnya.

يَا مَعْشَرَ الشَّبَابِ مَنْ اسْتَطَاعَ مِنْكُمُ الْبَاءَةَ فَلْيَتَزَوَّجْ وَمَنْ لَمْ يَسْتَطِعْ فَعَلَيْهِ بِالصَّوْمِ فَإِنَّهُ لَهُ وَجَاءٌ (رواه البخاري ومسلم)

“Wahai para pemuda, siapa di antara kalian yang telah mampu, maka hendaklah ia menikah, dan siapa yang belum mampu, maka hendaklah ia berpuasa, karena sesungguhnya puasa itu menjadi benteng baginya.” (HR Al Bukhori dan Muslim)

Dengan demikian, pernikahan dalam Islam tidak hanya sekadar ikatan fisik antara dua individu, melainkan juga merupakan ikatan sosial yang lebih mendalam. Secara sosial, pernikahan bertujuan untuk membentuk keluarga yang harmonis, di mana setiap anggota keluarga saling mendukung dalam menjalankan ajaran Islam.

2.1.2 Pandangan Islam terhadap Teknologi

Dalam perspektif Islam, teknologi didefinisikan sebagai alat, metode, dan proses yang diciptakan manusia untuk memudahkan berbagai aktivitas kehidupan. Teknologi merupakan manifestasi dari kemampuan intelektual manusia yang dianugerahkan oleh Allah SWT. Al-Qur'an mengajarkan bahwa manusia diciptakan dengan kemampuan untuk belajar dan mengembangkan ilmu pengetahuan yang bisa diterapkan dalam bentuk teknologi. Firman Allah dalam Surah Al-Baqarah ayat 31:

وَعَلَّمَ آدَمَ الْأَسْمَاءَ كُلَّهَا ثُمَّ عَرَضَهُمْ عَلَى الْمَلَائِكَةِ فَقَالَ أَنْبِئُونِي بِأَسْمَاءِ هَؤُلَاءِ إِنْ كُنْتُمْ صَادِقِينَ (سورة البقرة : ٣١)

"Dan Dia mengajarkan kepada Adam nama-nama (benda-benda) seluruhnya, kemudian mengemukakannya kepada para malaikat lalu berfirman: 'Sebutkanlah kepada-Ku nama benda-benda itu jika kamu memang orang-orang yang benar!'" (QS. Al-Baqoroh : 31)

Ayat ini menunjukkan bahwa manusia diberi kemampuan untuk mengetahui dan memanfaatkan alam sekitarnya, termasuk dalam menciptakan teknologi yang bermanfaat.

Islam mendorong umatnya untuk memanfaatkan teknologi dalam berbagai aspek kehidupan sehari-hari selama penggunaannya tidak bertentangan dengan ajaran agama dan nilai-nilai moral. Teknologi dapat digunakan untuk memperlancar ibadah, memajukan pendidikan, meningkatkan ekonomi, dan menjaga kesehatan. Contoh pemanfaatan teknologi yang positif adalah penggunaan aplikasi Al-Qur'an digital, kelas online untuk pembelajaran agama, serta teknologi kesehatan yang mendukung kesejahteraan umat.

Menurut Hadis Riwayat Muslim, Rasulullah SAW bersabda:

طَلَبُ الْعِلْمِ فَرِيضَةٌ عَلَى كُلِّ مُسْلِمٍ (رواه المسلم)

“Menuntut ilmu itu wajib bagi setiap muslim.” (HR. Muslim)

Hadits ini mengindikasikan pentingnya pendidikan dan ilmu pengetahuan, termasuk teknologi, sebagai sarana untuk memajukan umat manusia dalam berbagai bidang .

2.1.3 Urgensi Rancang Bangun Aplikasi Cari Jodoh

Urgensi dalam rancang bangun aplikasi cari jodoh semakin dirasakan seiring dengan perkembangan teknologi digital yang telah mengubah banyak aspek kehidupan, termasuk dalam mencari pasangan hidup. Di era modern ini, waktu menjadi salah satu faktor paling berharga, sehingga metode tradisional untuk bertemu pasangan potensial sering kali tidak lagi relevan dengan gaya hidup yang serba cepat. Dengan memanfaatkan teknologi, aplikasi cari jodoh memberikan kemudahan akses bagi pengguna untuk menemukan pasangan berdasarkan preferensi pribadi, tanpa terikat oleh batasan geografis maupun waktu. Seiring dengan meningkatnya penggunaan internet, terutama di kalangan generasi muda, platform digital semacam ini menjadi solusi yang semakin dibutuhkan dalam memenuhi kebutuhan sosial yang cenderung mengutamakan efisiensi dan kenyamanan. (Amelia & Ayu, 2020)

Rancang bangun aplikasi ini juga menjadi sangat penting dalam menjawab tantangan privasi dan keamanan pengguna yang kerap menjadi perhatian utama dalam dunia maya. Aplikasi dengan fitur keamanan yang kuat, seperti verifikasi identitas dan pengelolaan data pribadi yang aman, dapat meningkatkan kepercayaan pengguna. Di samping itu, penerapan algoritma Haversine dalam aplikasi ini memberikan manfaat yang signifikan dalam menentukan jarak terdekat antara pengguna, yang menjadi salah satu elemen penting dalam mencocokkan pasangan berdasarkan lokasi. Algoritma Haversine memungkinkan penghitungan jarak yang akurat di permukaan bumi, membantu pengguna menemukan pasangan yang berada di wilayah terdekat, sesuai dengan preferensi jarak yang mereka inginkan. Dengan demikian, akurasi dalam pencocokan tidak hanya didasarkan pada minat dan nilai, tetapi juga pada faktor geografis yang dioptimalkan melalui algoritma ini. Pengembang aplikasi harus memastikan bahwa desain aplikasi tidak hanya

berfokus pada kemudahan penggunaan, tetapi juga memberikan pengalaman yang aman, nyaman, dan sesuai dengan nilai-nilai lokal dan budaya pengguna.

2.2 Teknologi Android

2.2.1 Pengertian Android

Android adalah sistem operasi berbasis Linux yang dirancang khusus untuk perangkat seluler seperti *smartphone*. Dikembangkan oleh Android Inc., yang kemudian diakuisisi oleh Google pada tahun 2005, Android adalah *platform open-source* yang memungkinkan pengembang untuk membuat aplikasi dan perangkat lunak untuk dijalankan pada perangkat mobile. Sistem operasi ini menawarkan berbagai fitur, termasuk kemampuan untuk menjalankan aplikasi secara *multitasking*, sinkronisasi data, dan akses ke layanan Google seperti Gmail, Maps, dan Play Store. (Coding Studio Team, 2022)

2.2.2 Sejarah Android

Sejarah Android dimulai pada Oktober 2003 ketika Android Inc. didirikan oleh Andy Rubin, Rich Miner, Nick Sears, dan Chris White dengan tujuan menciptakan sistem operasi pintar yang mampu menyesuaikan pengalaman pengguna berdasarkan preferensi mereka. Google mengakuisisi Android Inc. pada Juli 2005 dalam sebuah langkah strategis yang tidak banyak diketahui saat itu, tetapi yang kemudian terbukti sangat penting bagi perkembangan teknologi selanjutnya.

Pada November 2007, Google bersama beberapa perusahaan teknologi terkemuka membentuk *Open Handset Alliance* (OHA), sebuah konsorsium yang bertujuan untuk mengembangkan standar terbuka untuk perangkat mobile. Pada waktu bersamaan, Android secara resmi diperkenalkan kepada publik sebagai *platform open-source*, membuka jalan bagi inovasi dan pengembangan lebih lanjut dalam ekosistem perangkat mobile. (Coding Studio Team, 2022)

2.2.3 Versi Android

Perangkat Android pertama, HTC Dream, yang juga dikenal sebagai T-Mobile G1 di beberapa wilayah, diluncurkan pada Oktober 2008. Perangkat ini menjalankan Android versi 1.0 dan menawarkan fitur-fitur dasar seperti browser web, email, serta dukungan untuk aplikasi pihak ketiga. Sejak saat itu, Android

mengalami pertumbuhan pesat dan terus berkembang, dengan berbagai pembaruan sistem operasi yang meningkatkan fitur, performa, dan keamanan. (Coding Studio Team, 2022)

Hingga kini, Android tetap menjadi salah satu platform dominan di pasar perangkat mobile. Berikut adalah Versi-versi android dari masa ke masa :

Tabel 2. 1 Versi Android

No	Versi	Keterangan
1	Astro 1.0 (Alpha)	Versi pertama Android, yang dirilis pada 23 September 2008, awalnya dinamai Astro. Namun, karena masalah hak cipta, nama tersebut tidak digunakan secara komersial. Versi ini juga pertama kali digunakan pada HTC Dream.
2	Bender 1.1 (Beta)	Versi kedua, Bender 1.1, diluncurkan pada 9 Februari 2009, masih menghadapi masalah hak penamaan seperti versi sebelumnya. Pada versi ini, Android Market sekarang dikenal sebagai Google Play Store.
3	Cupcake 1.5	Dirilis pada 27 April 2009, versi ini menggunakan nama makanan secara resmi untuk pertama kalinya dengan nama Cupcake. Versi ini memperkenalkan fitur <i>on-screen keyboard</i> .
4	Donut 1.6	Versi ini dirilis pada 15 September 2009, dengan peningkatan antarmuka pengguna yang sangat ramah dan fitur pencarian yang lebih baik. Versi ini juga menambah dukungan untuk teknologi CDMA/EVDO, 802.1x, dan VPN, serta memperbaiki jaringan dan layar.
5	Éclair 2.0 – 2.1	Versi ini membawa fitur Google Maps yang menggantikan peta konvensional, memberikan manfaat besar dalam mobilitas pengguna

No	Versi	Keterangan
6	Froyo 2.2	Diluncurkan pada 20 Mei 2010, versi ini, yang singkatan dari <i>frozen yoghurt</i> , memperkenalkan fitur seperti kunci PIN dan pemolesan sistem yang sudah ada sebelumnya.
7	Gingerbread 2.3	Rilis pada 6 Desember 2010, versi ini menawarkan pembaruan signifikan dalam hiburan, termasuk dukungan format video dan kemunculan kamera depan.
8	Honeycomb 3.0/3.1	Versi ini dirilis pada 22 Februari 2011 dan dirancang khusus untuk tablet, mendukung akselerasi <i>hardware</i> dan <i>multi prosesor</i> .
9	Ice Cream sandwich 4.0	Dirilis pada 19 Oktober 2011, versi ini menggabungkan berbagai fitur dari versi tablet, termasuk <i>face unlock</i> , email, dan penggunaan data internet.
10	Jelly Bean 4.1 – 4.3	Versi Jelly Bean memperkenalkan Google Now dengan asisten suara, fitur photo sphere untuk foto panorama, dan <i>daydream</i> untuk kontrol daya serta <i>screensaver</i> .
11	KitKat 4.4	Diluncurkan pada 31 Oktober 2013, versi ini menawarkan peningkatan pengalaman pengguna dan memerlukan perangkat dengan minimal RAM 512 MB.
12	Lollipop 5.0	Rilis pada 12 November 2014, versi ini memperkenalkan perubahan antarmuka pengguna dengan desain material yang baru melalui OTA (<i>Over the Air</i>).

No	Versi	Keterangan
13	Marshmallow 6.0	Dirilis pada 5 Oktober 2015, versi ini memperkenalkan fitur-fitur canggih seperti <i>search bar</i> , sensor sidik jari, dan perizinan aplikasi.
14	Nougat 7.0	Versi ini menghadirkan perubahan besar dengan tambahan fitur seperti <i>multi-window</i> dan 63 emoji baru
15	Oreo 8.0	Rilis pada 21 Agustus 2017, versi ini menambah fitur <i>multitasking</i> dan mengubah sistem notifikasi, sehingga pengguna dapat mengatur notifikasi yang ingin ditampilkan. Antarmuka pengguna juga menjadi lebih rapi.
16	Pie 9.0	Diluncurkan pada 6 Agustus 2018, versi Pie memperkenalkan fitur-fitur baru seperti <i>smart reply</i> , navigasi berbasis gestur, <i>digital wellbeing</i> , dan <i>adaptive battery</i> .
17	Android 10	Versi ini tidak lagi menggunakan nama makanan, mengingat sulitnya menemukan nama makanan yang dimulai dengan huruf Q.

(Sumber: Coding Studio)

2.3 Algoritma Haversine

2.3.1 Definisi Algoritma Haversine

Algoritma Haversine adalah cara sederhana dan efisien untuk menghitung jarak lingkaran besar antara dua titik di permukaan bumi. Hal ini didasarkan pada asumsi bahwa Bumi berbentuk bola, dan menggunakan rumus haversine untuk menghitung jarak antara dua titik pada permukaan bola. (Hermawan, 2022)

Algoritma Haversine diperkenalkan oleh Robert W. Sinnott dalam sebuah artikel yang diterbitkan pada tahun 1984. Nama "Haversine" berasal dari fungsi "haversin", yang merupakan modifikasi dari fungsi sinus, dirancang untuk perhitungan trigonometri dalam konteks sferis. Algoritma ini semakin umum

digunakan dengan perkembangan teknologi GPS dan aplikasi peta digital, yang membutuhkan perhitungan jarak yang akurat antara koordinat geografis. (Prakarsha, Kristianty Lumbantobing, Ramadhan, & Prihandi, 2021)

2.3.2 Prinsip Kerja Algoritma Haversine

Berikut adalah rumus Haversine yang digunakan untuk menghitung jarak antara dua titik pada permukaan bumi:

$$\alpha = \sin^2\left(\frac{(\Delta lat)}{2}\right) + \cos(lat1) \times \cos(lat2) \times \sin^2\left(\frac{(\Delta lon)}{2}\right)$$

$$c = 2 \cdot \text{atan}^2(\sqrt{\alpha}, \sqrt{1 - \alpha})$$

$$d = R \cdot c$$

Keterangan :

d = Jarak

r = Jari – jari bumi (6,371 km)

Δlat = Besar perubahan latitude

Δlon = Besar perubahan longitude

Rumus ini mengukur jarak antara dua titik dengan mempertimbangkan kelengkungan bumi, memberikan hasil yang lebih akurat dibandingkan dengan metode lain yang tidak memperhitungkan bentuk sferis bumi.

Berikut adalah cara kerja Algoritma Haversine dalam Menghitung Jarak antara Dua Titik Koordinat Geografis :

- 1) Konversi koordinat lintang dan bujur dari derajat ke radian

$$lat1 = latitude1 \times \frac{180}{\pi}$$

$$lat2 = latitude2 \times \frac{180}{\pi}$$

$$lon1 = longitude1 \times \frac{180}{\pi}$$

$$lon2 = longitude2 \times \frac{180}{\pi}$$

- 2) Hitung perbedaan lintang dan bujur dalam radian

$$\Delta lat = lat2 - lat1$$

$$\Delta lon = lon2 - lon1$$

- 3) Terapkan rumus haversine untuk menghitung jarak sudut

$$\alpha = \sin^2\left(\frac{(\Delta lat)}{2}\right) + \cos(lat1) \times \cos(lat2) \times \sin^2\left(\frac{(\Delta lon)}{2}\right)$$

$$c = 2 \cdot \text{atan}^2(\sqrt{\alpha}, \sqrt{1 - \alpha})$$

- 4) Kalikan hasil jarak sudut dengan jari-jari bumi untuk mendapatkan jarak sebenarnya

$$d = R \cdot c$$

di mana R adalah jari-jari bumi, sekitar 6371 km.

2.3.3 Implementasi Algoritma Haversine dalam Pemrograman

Berikut adalah contoh implementasi algoritma Haversine dalam bahasa Python:

```
import math

def haversine(lat1, lon1, lat2, lon2):
    #Ubah lintang dan bujur menjadi radian
    lat1, lon1, lat2, lon2 = map(math.radians, [lat1, lon1, lat2, lon2])

    #Hitung selisih antara dua koordinat
    dlat = lat2 - lat1
    dlon = lon2 - lon1

    #Terapkan rumus haversine
    a = math.sin(dlat / 2)**2 + math.cos(lat1) * math.cos(lat2) * \
        math.sin(dlon / 2)**2
    c = 2 * math.atan2(math.sqrt(a), math.sqrt(1 - a))

    #Hitung jari-jari bumi
    r = 6371 #jari-jari bumi dalam kilometer

    # Kembalikan jarak
    return c * r

#Contoh penggunaan fungsi
```

```

lat1, lon1 = -6.200000, 106.816666 #Koordinat Jakarta
lat2, lon2 = -6.914744, 107.609810 #Koordinat Bandung
jarak = haversine(lat1, lon1, lat2, lon2)
print(f"Jarak antara Jakarta dan Bandung adalah {jarak:.2f}
      km")

```

2.4 Algoritma Vincenty

2.4.1 Definisi Algoritma Vincenty

Algoritma Vincenty adalah metode yang lebih akurat untuk menghitung jarak besar lingkaran antara dua titik di permukaan bumi. Algoritma ini didasarkan pada asumsi bahwa Bumi berbentuk elipsoid (bukan bola sempurna), dengan kutub yang sedikit pipih, dan menggunakan rumus yang lebih kompleks untuk menghitung jarak tersebut. (Vincenty, 1975)

Ketika membandingkan algoritma Haversine dan Vincenty, penting untuk dicatat bahwa algoritma Vincenty lebih akurat karena mempertimbangkan bentuk elipsoid Bumi, sementara algoritma Haversine menganggap Bumi sebagai bola sempurna. Meskipun demikian, algoritma Haversine tetap berguna dan efisien untuk menghitung jarak besar lingkaran dengan cepat. (Hermawan, 2022)

Perbedaan utama lainnya antara kedua algoritma tersebut adalah kompleksitas komputasinya. Algoritma Haversine relatif sederhana dan efisien, sehingga cocok untuk aplikasi yang memerlukan penghitungan cepat pada kumpulan data besar. Sebaliknya, algoritma Vincenty lebih kompleks dan membutuhkan lebih banyak daya komputasi, sehingga lebih cocok untuk aplikasi yang membutuhkan akurasi tinggi. (Hermawan, 2022)

Secara singkat, baik algoritma Haversine maupun Vincenty berguna untuk menghitung jarak besar lingkaran antara dua titik di permukaan bumi. Algoritma Haversine menawarkan kesederhanaan dan efisiensi, sementara algoritma Vincenty menawarkan akurasi yang lebih tinggi meski dengan beban komputasi yang lebih besar. Pilihan antara kedua algoritma tergantung pada kebutuhan spesifik aplikasi yang digunakan.

2.4.2 Prinsip Kerja Algoritma Vincenty

Rumus ini bersumber dari karya T. Vincenty yang dipublikasikan pada tahun 1975. Dalam makalahnya yang berjudul “*Geodetic inverse solution between antipodal points*”, Vincenty mengembangkan metode untuk menghitung jarak antara dua titik di permukaan ellipsoid bumi dengan akurasi yang tinggi. Algoritma ini telah diadopsi secara luas dalam berbagai aplikasi geodesi dan navigasi. (Vincenty, 1975)

$$U_1 = \arctan((1 - f) \tan(lat1))$$

$$U_2 = \arctan((1 - f) \tan(lat2))$$

$$L = lon2 - lon1$$

$$\lambda = L$$

Iterasi untuk λ :

$$\sin(\sigma) = \sqrt{(\cos(U_2) \sin(\lambda))^2 + (\cos(U_1) \sin(U_2) - \sin(U_1) \cos(U_2) \cos(\lambda))^2}$$

$$\cos(\sigma) = \sin(U_1) \sin(U_2) + \cos(U_1) \cos(U_2) \cos(\lambda)$$

$$\sigma = \arctan\left(\frac{\sin(\sigma)}{\cos(\sigma)}\right)$$

$$\sin(\alpha) = \frac{\cos(U_1) \cos(U_2) \sin(\lambda)}{\sin(\sigma)}$$

$$\cos^2(\alpha) = 1 - \sin^2(\alpha)$$

$$\cos(2\sigma_m) = \cos(\sigma) - \frac{2 \sin(U_1) \sin(U_2)}{\cos^2(\alpha)}$$

$$C = \frac{f}{16} \cos^2(\alpha) [4 + f(4 - 3\cos^2(\alpha))]$$

$$\lambda_{new} = L + (1 - C)f \cdot \sin(\alpha)\sigma + C \cdot \sin(\sigma) (\cos(2\sigma_m) + C \cdot \cos(\sigma) - 1 + 2\cos^2(2\sigma_m))$$

Iterasi berlanjut sampai λ_{new} konvergen.

$$u^2 = \frac{\cos^2(\alpha)(a^2 - b^2)}{b^2}$$

$$A = 1 + \frac{u^2}{16384} [4096 + u^2(-768 + u^2(320 - 175u^2))]$$

$$B = \frac{u^2}{1024} [256 + u^2(-128 + u^2(74 - 47u^2))]$$

$$\Delta\sigma = B\sin(\sigma)\left[\cos(2\sigma_m) + \frac{B}{4}\left(\cos(\sigma)(-1 + 2\cos^2(2\sigma_m))\right) - \frac{B}{6}\cos(2\sigma_m)(-3 + 4\sin^2(\sigma))(-3 + 4\cos^2(2\sigma_m))\right]$$

$$d = bA(\sigma - \Delta\sigma)$$

Berikut adalah cara kerja Algoritma Vincenty dalam Menghitung Jarak antara Dua Titik Koordinat Geografis :

- 1) Konversi koordinat lintang dan bujur dari derajat ke radian

$$lat1 = latitude1 \times \frac{180}{\pi}$$

$$lat2 = latitude2 \times \frac{180}{\pi}$$

$$lon1 = longitude1 \times \frac{180}{\pi}$$

$$lon2 = longitude2 \times \frac{180}{\pi}$$

- 2) Tentukan parameter elipsoid

Parameter ellipsoid bumi yang umum digunakan adalah dari ellipsoid WGS-84 (*World Geodetic System 1984*), yang digunakan oleh sistem GPS. Parameter-parameter ini adalah:

- Semi-major axis (a)

Ini adalah jarak dari pusat ellipsoid ke equator. Nilai ini adalah konstan dan biasanya diberikan dalam meter.

$$a = 6,378,137.0 \text{ meter}$$

- Semi-minor axis (panjang sumbu semi-minor)

Ini adalah jarak dari pusat ellipsoid ke kutub. Dapat dihitung menggunakan nilai *flattening*.

$$b = a(1 - f)$$

$$b = 6,378,137.0 \left(1 - \frac{1}{298.257223563}\right)$$

$$b = 6,356,752.3142 \text{ meter}$$

- *Flattening* (f)

Flattening adalah rasio antara perbedaan panjang semi-major axis dan semi-minor axis dengan semi-major axis.

$$f = \frac{a-b}{a}$$

Pada ellipsoid WGS-84, nilai *flattening* adalah 1 / 298.257223563.

3) Hitung U_1 dan U_2

$$U_1 = \arctan((1-f)\tan(\text{lat1}))$$

$$U_2 = \arctan((1-f)\tan(\text{lat2}))$$

4) Hitung L

$$L = \text{lon2} - \text{lon1}$$

$$\lambda = L$$

5) Iterasi untuk λ

$$\lambda_{\text{new}} = L + (1-C)f \cdot \sin(\sigma) [\sigma + C \sin(\sigma) (\cos(2\sigma_m) + C \cos(\sigma)) - 1 + 2\cos^2(2\sigma_m))]$$

6) Hitung jarak

$$u^2 = \frac{\cos^2(\alpha)(a^2-b^2)}{b^2}$$

$$A = 1 + \frac{u^2}{16384} [4096 + u^2(-768 + u^2(320 - 175u^2))]$$

$$B = \frac{u^2}{1024} [256 + u^2(-128 + u^2(74 - 47u^2))]$$

$$\Delta\sigma = B \sin(\sigma) [\cos(2\sigma_m) + \frac{B}{4} (\cos(\sigma) (-1 + 2\cos^2(2\sigma_m))) -$$

$$\frac{B}{6} \cos(2\sigma_m) (-3 + 4\sin^2(\sigma)) (-3 + 4\cos^2(2\sigma_m))]$$

$$d = bA(\sigma - \Delta\sigma)$$

2.4.3 Implementasi Algoritma Vincenty dalam Pemrograman

Berikut adalah contoh implementasi algoritma Vincenty dalam bahasa Python:

```
import math
def vincenty(lat1, lon1, lat2, lon2, a=6378137.0,
f=1/298.257223563):
    b = (1 - f) * a

    U1 = math.atan((1 - f) * math.tan(math.radians(lat1)))
```

```

U2 = math.atan((1 - f) * math.tan(math.radians(lat2)))
L = math.radians(lon2 - lon1)
Lambda = L

sinU1 = math.sin(U1)
cosU1 = math.cos(U1)
sinU2 = math.sin(U2)
cosU2 = math.cos(U2)

max_iter = 200
tol = 1e-12
for i in range(max_iter):
    sinLambda = math.sin(Lambda)
    cosLambda = math.cos(Lambda)
    sinSigma = math.sqrt((cosU2 * sinLambda) ** 2 +
        (cosU1 * sinU2 - sinU1 * cosU2 * cosLambda) ** 2)
    cosSigma = sinU1 * sinU2 + cosU1 * cosU2 * cosLambda
    sigma = math.atan2(sinSigma, cosSigma)
    sinAlpha = cosU1 * cosU2 * sinLambda / sinSigma
    cos2Alpha = 1 - sinAlpha ** 2
    cos2SigmaM = cosSigma - 2 * sinU1 * sinU2 / cos2Alpha
    C = f/16 * cos2Alpha * (4 + f * (4 - 3 * cos2Alpha))
    Lambda_prev = Lambda
    Lambda = L + (1 - C) * f * sinAlpha * (sigma + C *
        sinSigma * (cos2SigmaM + C * cosSigma * (-1 + 2 *
            cos2SigmaM ** 2)))
    if abs(Lambda - Lambda_prev) < tol:
        break

u2 = cos2Alpha * (a ** 2 - b ** 2) / (b ** 2)
A = 1 + u2 / 16384 * (4096 + u2 * (-768 + u2 * (320 -
    175 * u2)))
B = u2 / 1024 * (256 + u2 * (-128 + u2 * (74 - 47 *
    u2)))
deltaSigma = B * sinSigma * (cos2SigmaM + B / 4 *
    (cosSigma * (-1 + 2 * cos2SigmaM ** 2) - B / 6 *
        cos2SigmaM * (-3 + 4 * sinSigma ** 2) * (-3 + 4 *
            cos2SigmaM ** 2)))

s = b * A * (sigma - deltaSigma)
return s / 1000 #convert to kilometers

#Contoh penggunaan fungsi
lat1, lon1 = -6.200000, 106.816666 #Koordinat Jakarta
lat2, lon2 = -6.914744, 107.609810 #Koordinat Bandung
jarak = vincenty(lat1, lon1, lat2, lon2)

```

```
print(f"Jarak antara Jakarta dan Bandung adalah
      {jarak:.2f} km")
```

2.5 Android Studio

2.5.1 Pengertian Android Studio



Gambar 2. 1 Logo Android Studio

Android Studio adalah *Integrated Development Environment* (IDE) resmi untuk mengembangkan aplikasi Android. “Android Studio *adalah tools Integrated Development Environment* (IDE) resmi hasil kolaborasi antara Google dan JetBrains yang dibuat khusus untuk pengembangan aplikasi android.” (Regita, 2021).

Android Studio pertama kali dirilis pada tanggal 16 Mei 2013 dalam acara Google I/O 2013. Sebelumnya, Google mendukung *Eclipse* IDE dengan *plugin Eclipse Android Developer Tool* (ADT) yang memungkinkan *Eclipse* untuk mengkompilasi dan menjalankan proyek Android. Seiring dengan berkembangnya aplikasi mobile, Google memutuskan untuk menghentikan dukungan terhadap ADT pada *Eclipse* dan beralih ke Android Studio karena menawarkan fitur yang lebih lengkap. Misalnya, Android Studio mendukung berbagai bahasa pemrograman seperti Kotlin dan memudahkan pengembangan aplikasi dengan *framework* JavaScript atau Flutter.

2.5.2 Spesifikasi Minimum Sistem

Karena Android Studio merupakan IDE yang menyediakan alat lengkap untuk desain, pengujian, dan debug, perangkat ini memerlukan banyak RAM pada komputer. Oleh karena itu, untuk menggunakan Android Studio, Anda memerlukan spesifikasi sistem minimum sebagai berikut:

Tabel 2. 2 Spesifikasi Minimum Android Studio

Komponen	Spesifikasi Minimum
Sistem Operasi	Windows 8/10 64-bit, MacOS 10.14, distro Linux 64-bit yang mendukung Gnome, KDE atau UnityDE
RAM	8 GB
Ruang Penyimpanan	8 GB (IDE + Android SDK + Android Emulator)
Resolusi Layar	1280 x 800

(Sumber: Niagahoster.co.id)

2.5.3 Fitur Utama Android Studio

Berikut adalah enam fitur Android Studio yang dirancang untuk meningkatkan produktivitas Anda dalam proses pengembangan aplikasi Android. Fitur-fitur ini tidak hanya mempermudah pekerjaan Anda tetapi juga membantu Anda bekerja dengan lebih cepat dan efisien. Dengan memanfaatkan fitur-fitur ini, Anda dapat meningkatkan kualitas dan kecepatan pengembangan aplikasi Anda secara signifikan.

➤ *Visual Layout Editor*

Layout Editor memungkinkan Anda untuk membuat tata letak dengan cepat tanpa perlu menulis kode XML secara manual. Anda dapat menggunakan metode *drag* and *drop* untuk menambahkan elemen UI ke *Visual Design Editor*. Fitur ini juga memungkinkan Anda untuk melihat pratinjau tata letak pada berbagai versi dan perangkat Android dengan mudah. Jika tata letak yang Anda buat tidak sesuai, Anda dapat menyesuaikan desain dengan melakukan *resize*. *Layout Editor* memiliki tujuh bagian utama: *Palette* yang berisi elemen UI, *Component Tree* yang menampilkan struktur komponen tata letak, *Toolbar* untuk mengatur tampilan *Layout Editor*, *Design Editor* sebagai tempat mengedit tampilan desain, *Attributes* untuk mengontrol tampilan atribut yang dipilih, *View Mode* untuk mengatur tampilan tata letak, dan *Zoom and Pan Controls* untuk mengontrol ukuran pratinjau dan posisi dalam editor.

➤ *APK Analyzer*

Fitur ini menganalisis komposisi APK atau *Android App Bundle* setelah proses build, membantu mengurangi ukuran APK dan mempercepat *debug* file DEX. Selain itu, fitur ini menyediakan informasi tentang file dan ukuran, komposisi file DEX, versi *final* file dalam aplikasi, serta memungkinkan perbandingan dua APK atau *app bundle* secara berdampingan.

➤ *Android Emulator*

Tools simulasi virtual memungkinkan Anda untuk menguji aplikasi Android tanpa memerlukan perangkat fisik. Dengan menggunakan alat ini, Anda dapat memilih berbagai jenis perangkat Android serta versi sistem operasi yang diinginkan, sehingga memudahkan proses pengujian aplikasi. Keuntungan utama dari menggunakan tools simulasi ini adalah kecepatan transfer data dari Android Studio ke *Emulator* yang lebih tinggi dibandingkan dengan transfer data ke perangkat Android melalui kabel USB. Ini membuat proses pengujian menjadi lebih efisien dan cepat.

➤ *Intelligent Code Editor*

Code editor cerdas memudahkan penulisan kode dengan fitur analisis dan saran kode melalui *auto-completion*. Editor ini menyediakan saran kode dalam daftar *dropdown* saat Anda mulai menulis, dan Anda dapat menggunakan saran yang muncul dengan menekan Tab. Fitur ini mempercepat dan memperbaiki proses penulisan kode.

➤ *Flexible Build System*

Didukung oleh *Gradle*, sistem ini memungkinkan *build* otomatis, manajemen *dependensi*, dan kustomisasi konfigurasi *build* yang fleksibel. *Gradle* dapat membangun aplikasi Android dengan berbagai versi konfigurasi pada perangkat yang berbeda dalam proyek yang sama.

➤ *Realtime Profilers*

Fitur ini menyediakan statistik tentang penggunaan sumber daya seperti CPU, memori, baterai, dan aktivitas jaringan saat menggunakan Android Studio. Dengan fitur ini, Anda dapat memantau kinerja Android Studio dan mendeteksi penurunan performa yang dapat menyebabkan aplikasi menjadi tidak responsif atau hang. Ini membantu Anda mengidentifikasi dan mengatasi masalah performa secara efektif.

2.6 *Blackbox Testing*

2.6.1 *Pengertian Blackbox Testing*

Blackbox testing adalah metode pengujian yang digunakan untuk mengamati hasil dari input dan output pada perangkat lunak tanpa mengetahui struktur kode internalnya. Pengujian ini dilakukan pada tahap akhir pengembangan perangkat lunak untuk memastikan bahwa perangkat lunak berfungsi dengan baik. Untuk melakukan pengujian, penguji tidak harus memiliki kemampuan menulis kode program. Pengujian ini dapat dilakukan oleh siapa saja. (Setiawan, Black Box Testing Untuk Menguji Perangkat Lunak, 2021)

2.6.2 *Teknik Blackbox Testing*

Ada beberapa teknik yang biasanya digunakan untuk menguji perangkat lunak. Berikut ini adalah teknik-tekniknya:

- *All Pair Testing*

Pengujian ini dilakukan untuk mengevaluasi semua kemungkinan kombinasi dari seluruh pasangan input parameter. Metode ini memastikan bahwa setiap pasangan parameter diuji, meningkatkan kemungkinan menemukan interaksi yang mungkin menyebabkan masalah.

- *Boundary Value Analysis*

Teknik ini fokus pada pencarian kesalahan yang terjadi di batas-batas nilai input, baik dari sisi luar maupun dalam. Pengujian ini bertujuan untuk menemukan kesalahan yang sering terjadi pada nilai ekstrem.

- *Cause-Effect Graph*

Teknik pengujian ini menggunakan grafik untuk menggambarkan hubungan antara efek dan penyebab dari kesalahan. Grafik ini memvisualisasikan bagaimana berbagai penyebab dapat mempengaruhi hasil, membantu dalam mendeteksi kesalahan yang mungkin terjadi.

- *Equivalence Partitioning*

Teknik ini membagi data input perangkat lunak menjadi beberapa partisi yang representatif. Pengujian dilakukan pada satu nilai dalam setiap partisi untuk mewakili seluruh kelompok nilai, mengurangi jumlah pengujian yang diperlukan.

- *Fuzzing*

Fuzzing adalah metode untuk menemukan *bug* dalam perangkat lunak dengan memasukkan data yang tidak terduga atau tidak *valid*. Teknik ini bertujuan untuk mengekspos kelemahan dengan memberikan *input* yang tidak sesuai dengan format yang diharapkan.

- *Orthogonal Array Testing:*

Teknik ini digunakan untuk pengujian input yang relatif kecil dan sederhana. Meskipun sangat efektif untuk kasus dengan input kecil, teknik ini bisa menjadi berat dan kompleks jika diterapkan pada skala yang lebih besar.

- *State Transition:*

Teknik ini digunakan untuk menguji mesin status dan navigasi antarmuka pengguna dengan memetakan transisi status dalam bentuk grafik. Ini membantu dalam memverifikasi bahwa aplikasi berfungsi dengan benar saat berpindah dari satu status ke status lainnya.

2.7 **Framework CodeIgniter**

2.7.1 **Pengertian CodeIgniter**

CodeIgniter merupakan *framework* PHP *open-source* yang dirancang untuk membangun aplikasi web, pertama kali dikembangkan oleh Rick Ellis pada tahun

2006. *Framework* ini diciptakan untuk menyediakan solusi yang sederhana dan efisien dalam pengembangan aplikasi web berbasis PHP. (Anendya A. , 2024)



Gambar 2. 2 Logo CodeIgniter

Popularitas CodeIgniter terus meningkat berkat ukurannya yang kecil namun mampu menghasilkan kecepatan eksekusi yang tinggi. Dengan performa tersebut, CodeIgniter mulai mendapatkan pengakuan luas di kalangan pengembang PHP.

Pada tahun 2014, perusahaan EllisLab yang dimiliki oleh Rick Ellis mengumumkan bahwa CodeIgniter akan diakuisisi oleh British Columbia Institute of Technology (BCIT). Sejak saat itu, BCIT berkomitmen untuk terus mengembangkan framework ini dengan merilis versi-versi terbaru dan menambahkan fitur-fitur baru.

Hingga kini, CodeIgniter tetap dikenal karena kombinasi uniknya, yaitu sifatnya yang ringan namun sangat powerful. Menurut Similartech, framework ini banyak digunakan di seluruh dunia, dengan Indonesia menempati peringkat ketiga setelah India dan Amerika Serikat.

Cloudways menggambarkan CodeIgniter sebagai alat yang sangat kuat, dirancang khusus untuk menciptakan aplikasi web PHP yang berfitur lengkap namun tetap sederhana.

Pernyataan resmi dari laman CodeIgniter juga mendukung hal ini, menyebutkan bahwa framework ini "dibuat untuk pengembang yang membutuhkan toolkit sederhana dan elegan untuk menciptakan aplikasi web berfitur lengkap." Salah satu fitur unggulan CodeIgniter adalah penggunaan konsep MVC, yang memisahkan logika bisnis, tampilan, dan pengelolaan data. Hal ini memudahkan

pengembang untuk bekerja pada bagian tertentu tanpa mengganggu struktur keseluruhan aplikasi.

2.7.2 Keunggulan CodeIgniter

Framework CodeIgniter tidak hanya dikenal karena fitur-fiturnya yang unggul, tetapi juga karena berbagai kelebihan yang membuatnya menjadi pilihan utama, seperti kemudahan penggunaan, ringan, efisien, *footprint* minimal, dan masih banyak lagi. (Anendya A. , 2024)

Berikut adalah beberapa keunggulan *Framework* CodeIgniter :

- Mudah digunakan

CodeIgniter dirancang dengan antarmuka yang sederhana dan mudah dipahami, bahkan untuk pengembang pemula. *Framework* ini juga memungkinkan pembuatan pustaka khusus yang mempermudah pengembangan dan perubahan fitur.

- Ringan dan efisien

CodeIgniter dikenal sebagai salah satu *framework* yang paling ringan dan efisien, terbukti dari waktu pemuatan yang cepat dan kinerja yang tinggi. *Framework* ini cocok untuk proyek dengan berbagai ukuran dan kompleksitas, bahkan di lingkungan hosting dengan sumber daya terbatas

- Minim *footprint*

CodeIgniter memiliki *footprint* yang sangat kecil, artinya penggunaan sumber daya komputer menjadi lebih efisien. Dengan *footprint* yang kecil, respons aplikasi menjadi lebih cepat, yang meningkatkan pengalaman pengguna.

- Penanganan *bug* dan *error*

CodeIgniter dilengkapi dengan antarmuka yang membantu dalam mendeteksi dan menangani *bug* serta *error* dengan cepat. Kesalahan PHP yang muncul dalam aplikasi dapat ditampilkan dengan jelas, sehingga memudahkan pengembang dalam menemukan dan menyelesaikan *bug*. Untuk mengatasi *bug* dan menambahkan fitur baru, Anda bisa merujuk ke GitHub CodeIgniter4.

- Fitur validasi

CodeIgniter memiliki fitur validasi formulir yang memudahkan penulisan kode menjadi lebih ringkas dan minim kesalahan. Ini memungkinkan pengembang menulis kode dengan lebih efisien.

- Fleksibel dan mudah disesuaikan

Framework ini menawarkan fleksibilitas dan kustomisasi tingkat tinggi, memungkinkan pengembang untuk memuat hanya komponen dan pustaka yang dibutuhkan proyek tertentu, sehingga menghasilkan aplikasi yang lebih ringan dan efisien.

- Keamanan bawaan

CodeIgniter dilengkapi dengan fitur keamanan bawaan untuk melindungi aplikasi dari ancaman umum seperti XSS, CSRF, dan *SQL injection*, sehingga risiko keamanan dapat diminimalisir tanpa mengubah fundamental aplikasi. *Framework* ini juga mendorong pengembang untuk mengikuti praktik pengkodean yang aman dengan menyediakan panduan untuk validasi input dan sanitasi data, guna meningkatkan keamanan data pengguna.

- *Caching* halaman web dan *database*

CodeIgniter mendukung *caching* halaman web dan *database*, yang membantu mempercepat aplikasi web dengan mengurangi beban server melalui penyimpanan halaman dalam *cache*. Pengembang dapat mengontrol *cache* per halaman serta durasinya.

- Pemisahan konsep modern

Meskipun CodeIgniter tidak mewajibkan penggunaan pola *model-view-controller* (MVC), *framework* ini tetap mendorong pengembang untuk mempercepat pengembangan aplikasi web besar dan kompleks dengan implementasi MVC yang fleksibel.

- Migrasi mudah

Pengguna dapat dengan mudah melakukan migrasi server dari satu hosting ke hosting lainnya, sehingga memudahkan pengelolaan perubahan dalam basis data dan *version control*.

- URL yang *SEO-Friendly*

CodeIgniter memungkinkan pembuatan URL yang *clean* dan ramah mesin pencari melalui pendekatan berbasis segmen, serta memudahkan pengunjung untuk mengubah URL agar lebih *SEO-friendly*.

- Panduan yang jelas

CodeIgniter menyediakan dokumentasi yang komprehensif dan panduan yang mudah dipahami, membantu pengembang dalam memahami dan menggunakan fitur-fitur yang tersedia. Contoh panduan untuk CodeIgniter4 juga tersedia.

- Dukungan komunitas dan resmi

Dengan komunitas yang aktif dan dukungan resmi dari CodeIgniter, pengembang selalu dapat memperoleh bantuan dan pembaruan terkini. Forum CodeIgniter juga menyediakan diskusi mengenai berbagai topik terkait *framework* ini.

2.8 Figma

2.8.1 Pengertian Figma

Figma merupakan salah satu aplikasi desain yang bisa digunakan di Windows dan Mac OS untuk membuat prototipe aplikasi dan berbagai jenis desain lainnya. Sebagai alat berbasis vektor, Figma sangat ideal untuk merancang antarmuka pengguna aplikasi atau situs web. (Anendya A. , 2022)

Selain itu, Figma juga tersedia sebagai aplikasi berbasis web, sehingga kamu tidak perlu menginstalnya di komputer. Dengan koneksi internet, kamu dapat mengakses Figma kapan saja dan di mana saja. Karena platform ini berbasis online, Figma secara otomatis menyimpan pekerjaan yang sedang kamu lakukan, jadi kamu tidak perlu khawatir selama koneksi internet tetap stabil.

2.8.2 Fungsi Figma

Dengan berbagai fitur yang telah dibahas, jelas bahwa Figma menawarkan kemudahan bagi para desainer. Berikut adalah beberapa fungsi utama yang bisa kamu manfaatkan dalam aplikasi Figma:

- Wireframe Aplikasi

Sebelum memulai desain aplikasi, UI/UX Designer biasanya membuat wireframe sebagai gambaran kasar dari desain. Wireframe ini berfungsi sebagai kerangka desain yang akan diperhalus menjadi visual yang lebih lengkap. Figma memudahkan proses ini dengan menyediakan alat untuk membuat wireframe aplikasi sebelum melanjutkan ke desain UI-nya.

- UI Aplikasi dan Website

Figma dirancang terutama untuk membuat desain antarmuka aplikasi, baik untuk website maupun perangkat mobile. Aplikasi ini menawarkan berbagai pilihan frame, termasuk frame untuk ponsel seperti iPhone dan Android, desktop, serta tablet. Pilihan ini mempermudah desain dan implementasi oleh programmer.

- Prototype

Setelah desain selesai, desainer sering kali membuat prototipe untuk menunjukkan alur kerja aplikasi dari satu halaman ke halaman lainnya. Di Figma, prototyping biasanya berupa interaksi klik antara halaman yang menggambarkan bagaimana alur desain bekerja. Ini termasuk tipe prototipe klik tayang yang menampilkan interaksi antar halaman.

- Mockup Desain

Figma juga dapat digunakan untuk membuat mockup, yaitu desain yang akan diterapkan pada produk akhir. Mockup memberikan gambaran detail tentang produk yang akan dihasilkan, seperti mockup logo, kemasan, dan lainnya.

- Desain Sosial Media

Selain desain antarmuka aplikasi, Figma juga menyediakan frame untuk pembuatan desain sosial media, termasuk post untuk Twitter, Instagram, dan platform lainnya. Dengan frame yang sudah disediakan, graphic designer tidak perlu lagi menyesuaikan ukuran secara manual.

2.9 Arsitektur Pengembangan Aplikasi

Arsitektur pengembangan aplikasi modern terdiri dari dua komponen utama, yaitu *frontend* dan *backend*. Keduanya saling mendukung untuk menciptakan aplikasi yang dapat digunakan secara efektif oleh pengguna akhir. *Frontend*, yang sering disebut sebagai sisi klien, berfokus pada penyajian data dan interaksi dengan pengguna melalui antarmuka visual. Sementara itu, *backend*, yang juga dikenal sebagai sisi server, berperan dalam pemrosesan data, penyimpanan informasi, dan manajemen logika bisnis di balik aplikasi tersebut. Kombinasi yang efisien antara *frontend* dan *backend* memungkinkan terciptanya aplikasi yang fungsional, responsif, dan skalabel.

2.9.1 Frontend Development

Frontend adalah bagian dari aplikasi yang berfokus pada aspek visual dan interaktif yang dapat dilihat dan diakses langsung oleh pengguna. *Frontend* berperan dalam menciptakan pengalaman pengguna (*User Experience/UX*) yang nyaman, dengan menampilkan antarmuka pengguna (*User Interface/UI*) yang terdiri dari berbagai elemen seperti tata letak, warna, tipografi, navigasi, serta elemen interaktif seperti tombol, formulir, dan *dropdown* menu. (Huda, 2023)

2.9.2 Backend Development

Backend adalah bagian dari aplikasi yang beroperasi di sisi server dan tidak terlihat atau diakses langsung oleh pengguna. *Backend* bertanggung jawab atas pemrosesan permintaan dari *frontend*, pengelolaan dan penyimpanan data di *database*, serta implementasi logika bisnis yang kompleks. *Backend* juga berperan penting dalam menjaga keamanan, otentikasi, dan otorisasi pengguna. Teknologi yang digunakan dalam pengembangan backend meliputi bahasa pemrograman server-side seperti Node.js, Python, PHP, dan Java, serta basis data seperti MySQL, PostgreSQL, atau MongoDB. *Backend* berfungsi sebagai jembatan yang menghubungkan antarmuka pengguna dengan data, memastikan bahwa informasi yang dibutuhkan dapat dikirim dan diterima dengan benar oleh *frontend*. (Huda, 2023)

BAB III ANALISA DAN PERANCANGAN SISTEM

3.1 Tempat dan Waktu Penelitian

3.1.1 Tempat Penelitian

Penulis melakukan penelitian berdasarkan lokus penelitian dan fakta-fakta yang ada pada lapangan.

Tempat : Desa Sukawera Kecamatan Ligung Kabupaten Majalengka

Waktu : 22 Mei 2024 – 30 Juli 2024

3.1.2 Waktu Penelitian

Untuk memberikan gambaran yang jelas mengenai jadwal pelaksanaan penelitian ini, berikut adalah tabel waktu penelitian yang mencakup seluruh proses dari perencanaan hingga Penyelesaian :

Tabel 3. 1 Waktu Penelitian

No	Prosedur Penelitian	Bulan														
		Mei					Juni					Juli				
		1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
1	Perencanaan Kebutuhan															
2	Desain Pengguna															
3	Konstruksi															
4	Penyelesaian															

3.2 Profil Desa Sukawera

3.2.1 Sejarah Desa Sukawera

Berdasarkan Profil Desa Sukawera Tahun 2023, Desa Sukawera berdiri pada sekitar tahun 1920-an di sebuah hutan yang sebelumnya tidak berpenghuni. Orang pertama yang datang ke sana mendirikan sebuah gubuk dan membuka ladang untuk memenuhi kebutuhannya, namun kemudian meninggalkan tempat itu tanpa alasan yang diketahui.

Pada sekitar tahun 1927, beberapa orang lain datang dan mulai menetap di area tersebut, dengan melakukan penebangan pohon untuk memperluas ladang dan pemukiman. Empat tokoh awal, yaitu Bapak Minah, Bapak Ceyong, Bapak Sura, dan Bapak Salemi, berdiskusi dan sepakat untuk menamai tempat itu "Tegal Begog" yang berasal dari kata "tegal" berarti pemukiman baru dan "begog" yang berarti babi, hewan yang banyak ditemukan di sana.

Pada tahun 1930, Kampung Tegal Begog terbentuk dengan sekitar 6 kepala keluarga. Beberapa pendatang baru, seperti Bapak Acep, Bapak Talam, dan Bapak Eyang, juga diterima untuk tinggal di sana. Jumlah penduduk pun meningkat menjadi sekitar 13 kepala keluarga. Meski administrasinya sudah bergabung dengan Desa Bantarwaru, kampung ini belum memiliki kepala dusun resmi karena keterbatasan sarana dan prasarana.

Pada tahun 1933, jumlah penduduk semakin bertambah dan kampung ini semakin ramai. Nama "Tegal Begog" dianggap perlu direvisi untuk mencerminkan identitas yang lebih baik. Setelah diskusi, kampung ini diberi nama baru, yaitu "Sukawera," yang berarti "Suka Keindahan."

Pada tahun 1960-an, generasi tokoh masyarakat baru seperti Bapak Tayum, Bapak Karinah, dan Bapak Tolib melanjutkan perjuangan untuk membentuk desa. Akhirnya, pada tahun 1980, dengan persyaratan yang terpenuhi, Desa Sukawera resmi dibentuk. Desa ini diresmikan oleh Bupati Majalengka, H. Moch. Saleh Paindra, pada 19 Juni 1982, pukul 14.00 WIB. Daftar kepala desa Sukawera dari masa ke masa adalah sebagai berikut:

Tabel 3. 2 Daftar Kepala Desa Sukawera

No	Nama Kepala Desa	Periode
1	Radji	1982–1997
2	S. Raska	1998–2009
3	Darma	2010–2021
4	Wawan, S.Kom.I	2022–2027

3.2.2 Karakteristik Demografis Desa Sukawera

Desa Sukawera merupakan salah satu desa di Kecamatan Ligung Kabupaten Majalengka, Provinsi Jawa Barat, memiliki luas $\pm 256,461$ Ha. Secara geografis Desa Sukawera berbatasan dengan wilayah sebagai berikut :

Tabel 3. 3 Perbatasan Desa Sukawera

No	Arah	Batas Wilayah
1	Sebelah Utara	Desa Bantarwaru
2	Sebelah Timur	Desa Majasari dan Leuweung Hapit
3	Sebelah Selatan	Desa Majasuka dan Desa Cisambeng
4	Sebelah Barat	Desa Ligung dan Leuwiliang Baru

Secara Administratif, wilayah Desa Sukawera terdiri dari 4 Dusun, 8 Rukun Warga, dan 19 Rukun Tetangga. Sedangkan secara umum Tipologi Desa Sukawera Terdiri dari Areal Pesawahan. Topografis Desa Sukawera secara umum termasuk daerah landai atau dataran rendah dan berdasarkan ketinggian wilayah Desa Sukawera diklasifikasikan kepada Dataran Rendah dengan ketinggian 0-100 m dpl.

Jumlah Penduduk Desa Sukawera berdasarkan Profil Desa tahun 2022 sebanyak 2.874 jiwa yang terdiri dari 1.452 Laki-laki dan 1.422 Perempuan. Sumber penghasilan utama penduduk Sukawera Adalah Petani.

Tabel 3. 4 Jumlah Penduduk Desa Sukawera

No	Penduduk	Satuan	Tahun		
			2020	2021	2022
1	Laki-laki	Orang	1452	1410	1452
2	Perempuan	Orang	1582	1576	1422
Total		Orang	3034	2986	2874

3.3 Analisis Masalah

Angka pernikahan di Indonesia mengalami penurunan yang signifikan dalam beberapa tahun terakhir. Salah satu faktor yang mempengaruhi hal ini adalah gaya hidup masyarakat yang semakin padat, mengurangi waktu dan kesempatan untuk mencari pasangan hidup. Di Desa Sukawera, fenomena ini juga terjadi. Meskipun sudah ada beberapa platform pencarian jodoh yang tersedia, jumlah pengguna di platform tersebut sangat sedikit. Rendahnya partisipasi pengguna lokal menghambat proses pencarian pasangan yang relevan dan mudah dijangkau, sehingga pengguna kesulitan menemukan pasangan yang berada dalam jarak dekat.

Sebagai langkah kecil untuk membantu meningkatkan angka pernikahan, penelitian ini berfokus pada perancangan aplikasi pencarian jodoh berbasis Android yang dimulai dari desa. Aplikasi ini diharapkan dapat menjadi solusi efektif yang dapat menarik minat lebih banyak pengguna di Desa Sukawera dan sekitarnya, serta memudahkan mereka dalam menemukan pasangan yang sesuai dengan preferensi dan berada dalam jarak yang dekat..

3.4 Analisis Kebutuhan

Alat yang digunakan pada penelitian ini berupa perangkat keras dan perangkat lunak yang akan digunakan untuk mengembangkan aplikasi cari jodoh. Pada penelitian ini digunakan alat penelitian berupa perangkat keras dan perangkat lunak sebagai berikut:

3.4.1 Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan penulis untuk merancang dan mengembangkan aplikasi cari jodoh ini adalah Laptop Acer Aspire A515-45 dengan spesifikasi sebagai berikut :

- 1) *Processor* AMD Ryzen 5 5500U
- 2) CPU 6 inti dengan *clock speed* 2.1 GHz
- 3) RAM 16 GB DDR4
- 4) SSD NVME berkapasitas 256 GB

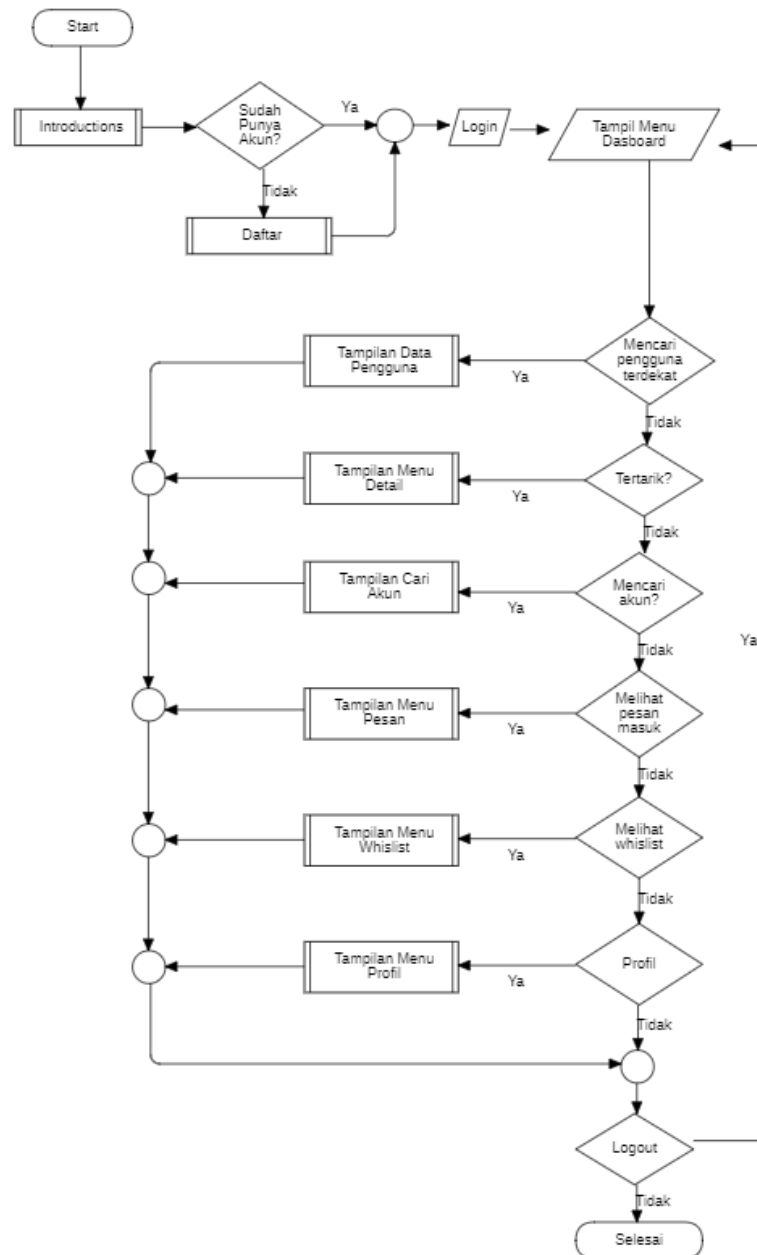
3.4.2 Perangkat Lunak (*Software*)

Dalam mengembangkan aplikasi pencarian jodoh, penulis menggunakan beberapa perangkat lunak yang dapat mendukung perancangan ini. Diantaranya adalah :

- 1) Sistem Operasi Windows 11 Pro
- 2) Android Studio
- 3) *Framework* CodeIgniter 4.3

3.5 Flowchart Perancangan Sistem Kerja

Berikut adalah *Flowchart* yang menggambarkan alur kerja aplikasi pencarian jodoh yang sedang peneliti kembangkan :



Gambar 3. 1 *Flowchart* Sistem Kerja

Proses aplikasi dimulai dengan halaman pengantar yang memberikan informasi awal tentang aplikasi dan fitur-fiturnya. Setelah itu, pengguna diarahkan ke halaman *login* untuk memasukkan kredensial mereka. Jika pengguna belum memiliki akun, mereka akan diarahkan ke halaman pendaftaran untuk membuat akun baru dengan mengisi informasi seperti nama, email, dan kata sandi.

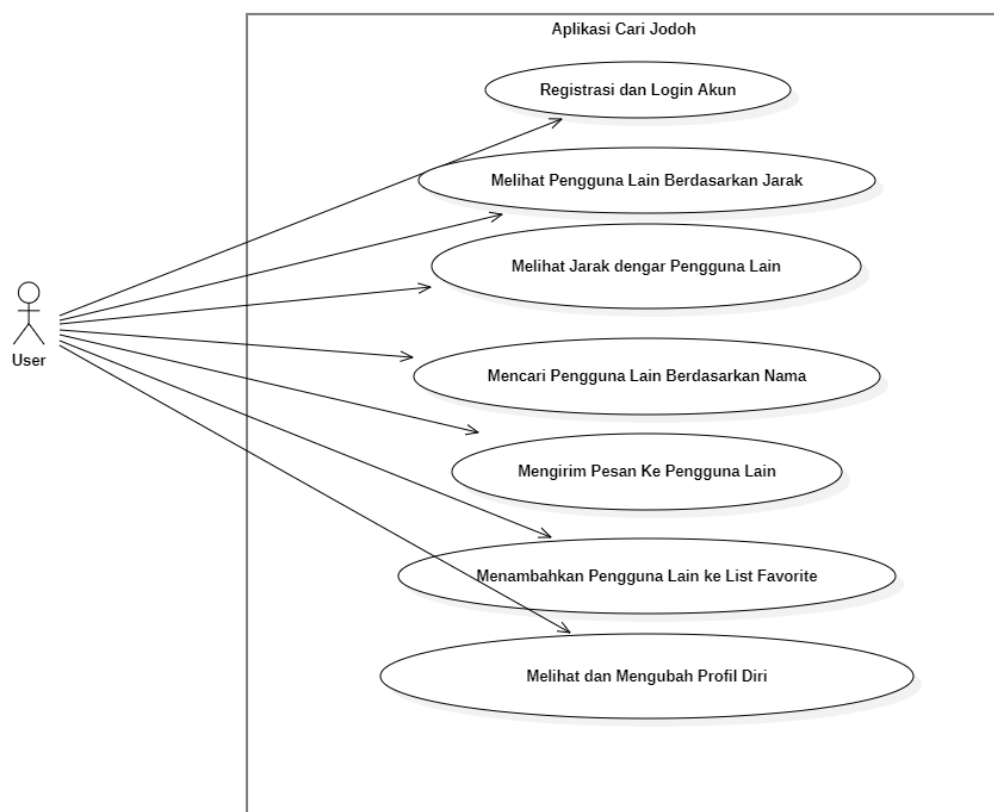
Setelah berhasil *login*, pengguna akan diarahkan ke menu utama atau *dashboard*, yang berfungsi sebagai pusat navigasi aplikasi. Di *dashboard*, pengguna dapat memilih berbagai opsi menu sesuai kebutuhan. Salah satu fitur utama adalah menampilkan data pengguna lain, termasuk nama lengkap, tempat lahir, tanggal lahir, pekerjaan, alamat, jarak, deskripsi, dan foto profil. Fitur ini membantu pengguna untuk menemukan pasangan yang sesuai dengan memberikan gambaran lengkap tentang pengguna lain. Selain itu, pengguna dapat melihat peta lokasi pengguna lain dan mengirim pesan langsung untuk berinteraksi. Selain menu utama, terdapat beberapa menu lain yang penting:

- 1) Menu Pencarian : Di menu ini, pengguna dapat mencari pengguna lain berdasarkan nama. Fitur pencarian ini mempermudah pengguna dalam menemukan orang yang spesifik yang mereka cari.
- 2) Menu Pesan : Di menu ini, pengguna dapat melihat dan mengelola pesan masuk dari pengguna lain. Fitur ini sangat penting untuk menjaga komunikasi antara pengguna yang tertarik satu sama lain.
- 3) Menu Suka : Menu ini memungkinkan pengguna untuk melihat daftar pengguna yang telah mereka masukkan ke dalam *wishlist*. Fitur ini berguna untuk menyimpan profil pengguna yang menarik dan ingin dijelajahi lebih lanjut di kemudian hari.
- 4) Menu Profil : Menu ini memungkinkan pengguna untuk melihat dan mengubah data pribadi mereka. Pengguna dapat memperbarui informasi seperti foto profil, deskripsi, dan detail pribadi lainnya kapan saja.

Jika pengguna selesai menggunakan aplikasi, mereka dapat memilih untuk *logout*, yang memungkinkan mereka keluar dari akun dengan aman. Setelah *logout*, pengguna akan diarahkan kembali ke halaman *login*. Fitur ini melindungi data dan privasi pengguna, memastikan bahwa tidak ada orang lain yang dapat mengakses akun tanpa izin. Dengan berbagai menu dan fitur yang lengkap, aplikasi ini dirancang untuk memberikan pengalaman pengguna yang mudah dan intuitif, memudahkan pencarian dan interaksi dengan pengguna lain, serta mengelola informasi pribadi dengan efisien.

3.6 Perancangan Sistem

Diagram di bawah ini menunjukkan *Use Case* dari aplikasi pencarian jodoh yang sedang dikembangkan. Diagram ini menggambarkan berbagai tindakan yang dapat dilakukan oleh pengguna dalam aplikasi, seperti *registrasi* dan *login* akun, melihat pengguna lain berdasarkan jarak, mengirim pesan, serta mengelola profil dan daftar favorit. Setiap garis panah mengindikasikan interaksi langsung antara pengguna dengan fitur-fitur utama dalam aplikasi.



Gambar 3. 2 *Use Case* Aplikasi

Pada *Use Case* Diagram aplikasi cari jodoh, terdapat satu aktor utama yaitu pengguna. Aktor pengguna dapat melakukan berbagai aktivitas dalam aplikasi, yang dirinci sebagai berikut:

- 1) *Registrasi* dan *login* akun : Pengguna dapat mendaftarkan akun baru jika belum memiliki akun atau *login* ke akun yang sudah ada. Proses ini adalah langkah awal yang harus dilakukan untuk mengakses fitur-fitur lain dalam aplikasi.

- 2) Melihat pengguna lain berdasarkan jarak : Setelah *login*, pengguna dapat melihat profil pengguna lain yang berada dalam jarak tertentu dari lokasi mereka. Fitur ini memungkinkan pengguna untuk menemukan calon pasangan yang berada di dekat mereka.
- 3) Melihat jarak dengan pengguna lain : Pengguna dapat melihat seberapa jauh jarak mereka dengan pengguna lain yang profilnya sedang mereka lihat. Informasi jarak ini berguna untuk merencanakan pertemuan atau sekadar mengetahui kedekatan lokasi.
- 4) Mencari pengguna lain berdasarkan nama : Pengguna dapat mencari pengguna lain dengan memasukkan nama tertentu. Fitur ini memudahkan pengguna untuk menemukan orang yang mereka kenal atau orang yang spesifik yang ingin mereka temui.
- 5) Mengirim pesan ke pengguna lain : Pengguna dapat mengirim pesan pribadi ke pengguna lain. Fitur ini memungkinkan komunikasi langsung antara pengguna, sehingga mereka bisa lebih mengenal satu sama lain.
- 6) Menambahkan pengguna lain ke daftar suka : Pengguna dapat menambahkan profil pengguna lain ke daftar suka mereka. Fitur ini berguna untuk menyimpan profil pengguna yang menarik bagi mereka sehingga mudah diakses kembali di kemudian hari.
- 7) Melihat dan mengubah profil diri : Pengguna dapat melihat dan mengedit informasi profil mereka sendiri, termasuk foto, deskripsi, dan detail lainnya. Fitur ini memastikan bahwa profil pengguna selalu terbaru dan mencerminkan informasi yang akurat tentang diri mereka.

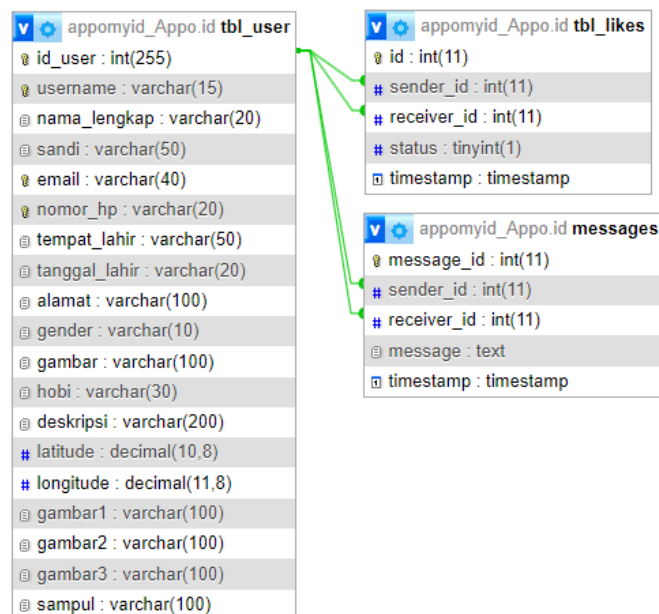
Dengan berbagai fitur ini, aplikasi memberikan pengguna kemampuan untuk mencari, berinteraksi, dan mengelola informasi pengguna lain serta informasi pribadi mereka sendiri. Setiap aktivitas yang dapat dilakukan pengguna dirancang untuk mempermudah dan memperkaya pengalaman mereka dalam menggunakan aplikasi cari jodoh.

3.7 Perancangan Database

Database atau basis data adalah kumpulan data yang dikelola sedemikian rupa berdasarkan ketentuan tertentu yang saling berhubungan sehingga mudah dalam pengelolaannya. Melalui pengelolaan tersebut pengguna dapat memperoleh kemudahan dalam mencari informasi, menyimpan informasi dan membuang informasi. (Kusuma, 2020)

3.7.1 Tabel

Dalam perancangan aplikasi ini, penulis menggunakan 3 tabel dalam *database*, diantaranya adalah *tbl_user*, *tbl_likes* dan *messages*.



Gambar 3. 3 Relasi Antar Tabel

3.7.2 Kamus Data

Berikut ini adalah kamus data atau dokumentasi yang berfungsi untuk mendeskripsikan elemen-elemen data yang digunakan dari aplikasi cari jodoh yang sedang penulis kembangkan :

Database : Appo.id

Jumlah Tabel : 3 Tabel

Tabel 3. 5 Tabel tbl_user

tbl_user			
No	Nama Kolom	Tipe	Keterangan
1	id_user	Integer	Kode unik tiap pengguna
2	username	Varchar	Nama akun pengguna
3	nama_lengkap	Varchar	Nama pengguna
4	sandi	Varchar	Kata sandi pengguna
5	email	Varchar	Alamat email pengguna
6	nomor_hp	Varchar	Nomor telepon pengguna
7	tempat_lahir	Varchar	Tempat lahir pengguna
8	tanggal_lahir	Varchar	Tanggal lahir pengguna
9	alamat	Varchar	Alamat pengguna
10	gender	Varchar	Jenis kelamin pengguna
11	gambar	Varchar	Nama file foto profil pengguna
12	hobi	Varchar	Kegemaran pengguna
13	deskripsi	Varchar	Keterangan tentang pengguna
14	latitude	Decimal	Nilai lokasi latitude pengguna
15	longitude	Decimal	Nilai lokasi longitude pengguna
16	gambar1	Varchar	Nama file gambar pengguna
17	gambar2	Varchar	Nama file gambar pengguna
18	gambar3	Varchar	Nama file gambar pengguna
19	sampul	Varchar	Nama file sampul pengguna

Tabel 3. 6 Tabel tbl_likes

tbl_likes			
No	Nama Kolom	Tipe	Keterangan
1	id	Integer	Kode unik tbl_likes tiap pengguna
2	sender_id	Integer	Pengirim berupa kode unik tiap pengguna dari tabel id_user

tbl_likes			
No	Nama Kolom	Tipe	Keterangan
3	receiver_id	Integer	Penerima berupa kode unik tiap pengguna dari tabel id_user
4	status	Boolean	Status berupa true / false
5	timestamp	Timestamp	Waktu memberikan like / unlike

Tabel 3. 7 Tabel tbl_chat

tbl_chat			
No	Nama Kolom	Tipe	Keterangan
1	id	Integer	Kode unik tbl_chat tiap pengguna
2	sender_id	Integer	Pengirim berupa kode unik tiap pengguna dari tabel id_user
3	receiver_id	Integer	Penerima berupa kode unik tiap pengguna dari tabel id_user
4	message	Boolean	Isi teks pesan
5	timestamp	Timestamp	Waktu mengirim pesan

3.8 Perancangan *Interface* (Antarmuka)

Dalam proses perancangan desain aplikasi pencarian jodoh, penulis menggunakan Figma sebuah platform desain populer yang digunakan untuk membuat prototipe interaktif. Figma dipilih karena kemampuannya yang unggul dalam memvisualisasikan antarmuka pengguna secara detail. Melalui Figma, penulis berhasil merancang tampilan dan alur kerja aplikasi yang sesuai dengan kebutuhan pengguna serta tujuan dari aplikasi tersebut. Hasil dari perancangan ini mencakup berbagai aspek desain antarmuka, mulai dari halaman utama hingga fitur-fitur khusus yang mendukung pencarian jodoh. Berikut ini adalah hasil rancangan prototipe desain aplikasi yang telah disusun oleh penulis untuk mengilustrasikan konsep dan fungsi aplikasi yang akan dikembangkan :

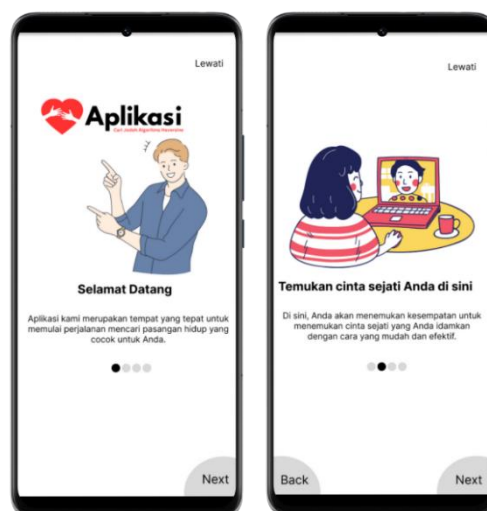
3.8.1 Tampilan Splashscreen

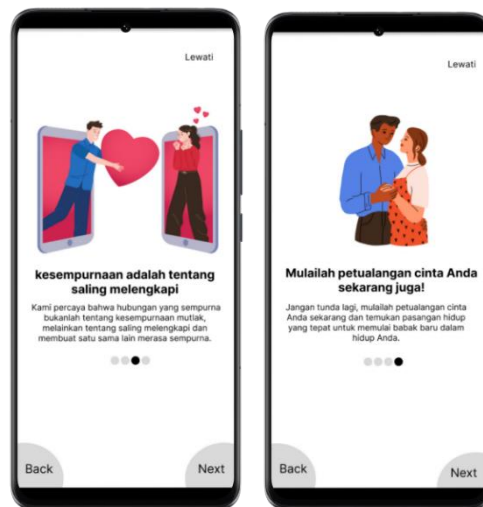


Gambar 3. 4 Rancangan Tampilan *Splashscreen*

Splashscreen adalah tampilan awal yang muncul saat membuka aplikasi. Pada Aplikasi ini, *splashscreen* menampilkan logo dan *progressbar loading* sebelum masuk ke halaman selanjutnya. Fungsinya adalah memberikan waktu bagi aplikasi untuk memuat sumber daya yang diperlukan sebelum pengguna mulai menggunakan aplikasi secara aktif.

3.8.2 Tampilan Intro Aplikasi

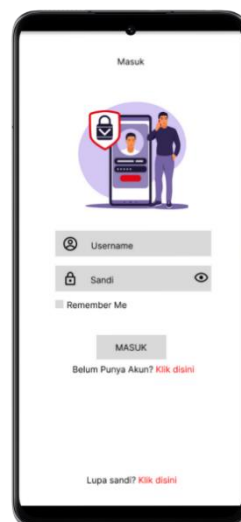




Gambar 3. 5 Rancangan Tampilan Intro

Intro aplikasi adalah bagian yang hanya muncul saat pengguna membuka aplikasi untuk pertama kalinya. Fitur ini dirancang untuk memberikan gambaran singkat tentang apa yang dapat dilakukan dengan aplikasi tersebut, sehingga membantu pengguna memahami cara menggunakan aplikasi serta mengenali fitur-fitur utamanya. Intro ini berperan penting dalam membimbing pengguna baru, memastikan mereka dapat memanfaatkan aplikasi secara maksimal sejak awal.

3.8.3 Tampilan Login Aplikasi

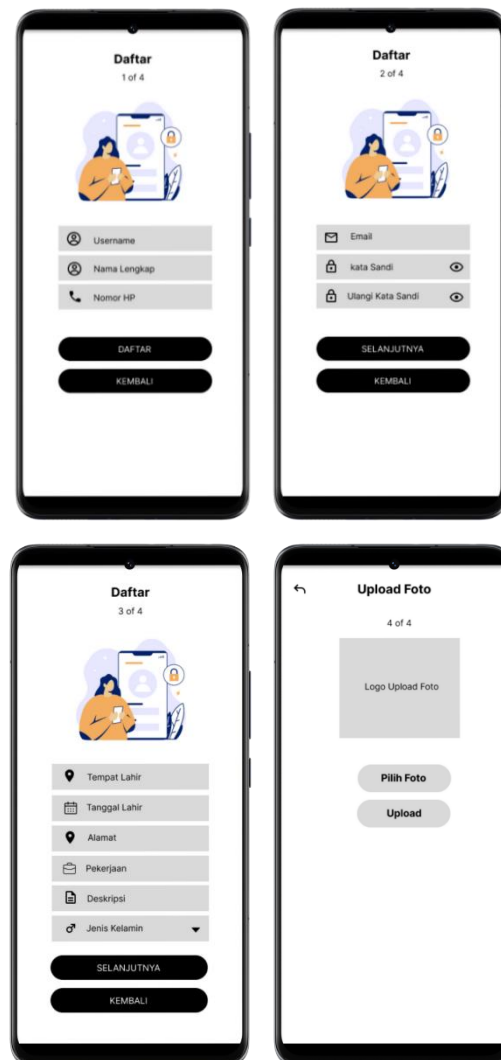


Gambar 3. 6 Rancangan Tampilan Login

Proses *login* aplikasi terjadi ketika pengguna memasukkan nama pengguna dan kata sandi untuk mengakses akun mereka di dalam aplikasi. Selain

menggunakan nama pengguna, aplikasi ini juga menyediakan opsi *login* menggunakan nomor *handphone* atau alamat email yang telah terdaftar sebelumnya, hal ini dapat memberikan fleksibilitas lebih kepada pengguna. Pada halaman *login*, pengguna juga memiliki opsi untuk tetap masuk ke aplikasi dengan mencentang kolom "*remember me*". Jika opsi ini dipilih, pengguna tidak perlu lagi memasukkan nama pengguna dan kata sandi saat membuka aplikasi di lain waktu, sehingga pengalaman penggunaan menjadi lebih nyaman dan efisien.

3.8.4 Tampilan Daftar Aplikasi



Gambar 3. 7 Rancangan Tampilan Daftar

Proses pendaftaran akun pada aplikasi ini dimulai ketika pengguna membuat akun baru di dalam aplikasi. Tahap pendaftaran akun dirancang secara bertahap

untuk memastikan bahwa semua informasi yang diperlukan terkumpul dengan baik. Ada empat tahap utama dalam alur pendaftaran akun pada aplikasi ini, yang masing-masing harus dilalui oleh pengguna untuk menyelesaikan proses pendaftaran dengan sukses. Berikut adalah rincian dari keempat tahap tersebut :

- Pengisian *Username*, Nama Lengkap, dan Nomor *Handphone*

Pengguna pertama-tama diminta untuk mengisi *username*, nama lengkap, dan nomor *handphone* mereka. Beberapa validasi akan dilakukan untuk memastikan data yang dimasukkan memenuhi syarat. Jika *username* yang diinput sudah terdaftar, akan muncul pesan *error* yang berbunyi "*Username* sudah terdaftar". Selain itu, *username* tidak diperbolehkan mengandung spasi atau angka. Jika pengguna memasukkan *username* dengan spasi atau angka, akan muncul pesan *error* "*Username* tidak boleh mengandung spasi dan angka". Untuk nomor *handphone*, validasi dilakukan untuk memastikan bahwa nomor tersebut belum terdaftar di sistem. Jika nomor sudah terdaftar, pengguna akan menerima pesan *error* yang sesuai, yang menunjukkan bahwa nomor tersebut tidak dapat digunakan lagi..

- Pengisian Alamat Email dan Kata Sandi

Pengguna kemudian harus mengisi alamat email dan kata sandi. Untuk memastikan bahwa pengguna dapat mengingat dan menggunakan kata sandi dengan benar, sistem meminta pengguna untuk memasukkan kata sandi dua kali. Jika kedua input kata sandi tidak cocok, sistem akan memberikan pesan *error* yang menginformasikan bahwa kata sandi yang dimasukkan tidak sesuai. Hal ini bertujuan untuk mencegah kesalahan dalam pembuatan kata sandi yang dapat menyebabkan pengguna tidak dapat login di kemudian hari.

- Pengisian Data Pribadi dan Izin Lokasi

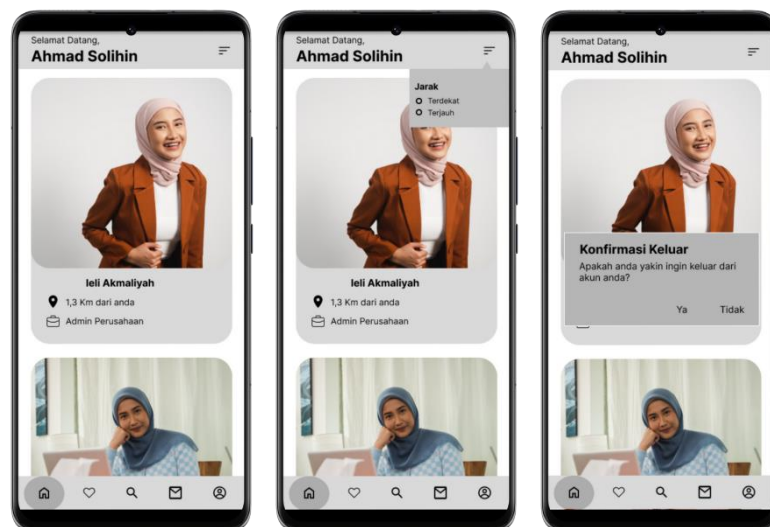
Pada tahap ini, pengguna diminta untuk mengisi informasi pribadi lainnya, seperti tempat lahir, tanggal lahir, alamat, pekerjaan, deskripsi diri, dan jenis kelamin. Selain itu, aplikasi akan meminta izin untuk mengakses lokasi pengguna. Data lokasi ini akan digunakan untuk fitur-

fitur dalam aplikasi dan akan disimpan di database bersama dengan data-data lain yang telah diisi oleh pengguna. Izin lokasi ini penting untuk mengoptimalkan fungsi pencarian dan penyaringan pasangan berdasarkan jarak.

- Pengisian Foto Profil

Terakhir, pengguna diminta untuk mengunggah setidaknya satu foto pribadi sebagai foto profil. Ini dilakukan untuk memastikan bahwa semua akun yang terdaftar memiliki identitas visual yang jelas, yang dapat meningkatkan kepercayaan dan kenyamanan dalam interaksi antar pengguna. Setelah foto profil diunggah, pengguna akan diarahkan ke halaman login untuk memasukkan username dan kata sandi mereka. Pada tahap ini, pengguna sudah dapat mengakses semua fitur aplikasi dan mulai menggunakan layanannya.

3.8.5 Tampilan *Dashboard*



Gambar 3. 8 Rancangan Tampilan *Dashboard*

Dashboard adalah halaman utama dalam aplikasi yang berfungsi sebagai pusat informasi bagi pengguna. Halaman ini dirancang untuk memberikan gambaran menyeluruh tentang berbagai data dan aktivitas pengguna lain yang relevan, serta memudahkan pengguna dalam pengambilan keputusan atau navigasi di dalam aplikasi. *Dashboard* memainkan peran penting dalam memastikan bahwa

informasi yang dibutuhkan pengguna tersedia dengan cepat dan mudah diakses, sehingga pengalaman pengguna menjadi lebih efisien dan intuitif. Pada halaman *dashboard* ini, pengguna dapat melakukan tiga hal utama:

- Memfilter Pengguna Lain Berdasarkan Jarak

Pengguna dapat memanfaatkan fitur *filter* yang tersedia di *dashboard* untuk menyaring daftar pengguna lain berdasarkan jarak geografis. Fitur ini memungkinkan pengguna menemukan calon pasangan yang berada dalam radius lokasi tertentu, meningkatkan relevansi dan efisiensi dalam pencarian. Dengan memfokuskan hasil pencarian pada orang-orang yang berada di area terdekat, fitur ini membantu memperbesar peluang interaksi dan koneksi yang lebih realistis dan praktis.

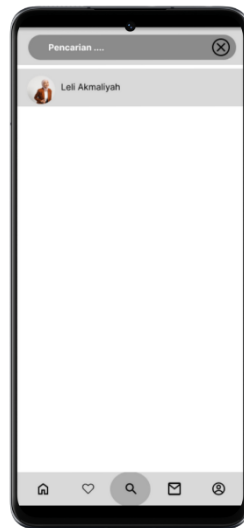
- Melihat Detail dari Pengguna Lain

Di *dashboard*, pengguna memiliki kemampuan untuk mengklik atau memilih profil pengguna lain untuk melihat informasi lebih mendalam. Dalam konteks aplikasi ini, pengguna lain yang dimaksud adalah calon pasangan potensial. Fitur ini memberikan akses ke detail penting seperti bio, minat, dan foto profil, yang membantu pengguna untuk mengenal calon pasangan lebih baik sebelum memutuskan untuk berinteraksi lebih lanjut. Kemudahan dalam mengeksplorasi informasi ini mempercepat proses pencarian dan seleksi pasangan yang sesuai dengan preferensi.

- Keluar dari Aplikasi

Fitur ini memungkinkan pengguna untuk menutup aplikasi dengan mudah dan cepat setelah selesai digunakan. Jika sebelumnya pengguna telah mencentang opsi "*remember me*" saat *login*, maka saat aplikasi dibuka kembali, mereka tidak perlu melakukan *login* ulang. Hal ini mempermudah akses ke aplikasi dan memastikan pengalaman pengguna tetap lancar, tanpa perlu memasukkan kembali nama pengguna dan kata sandi setiap kali ingin menggunakan aplikasi.

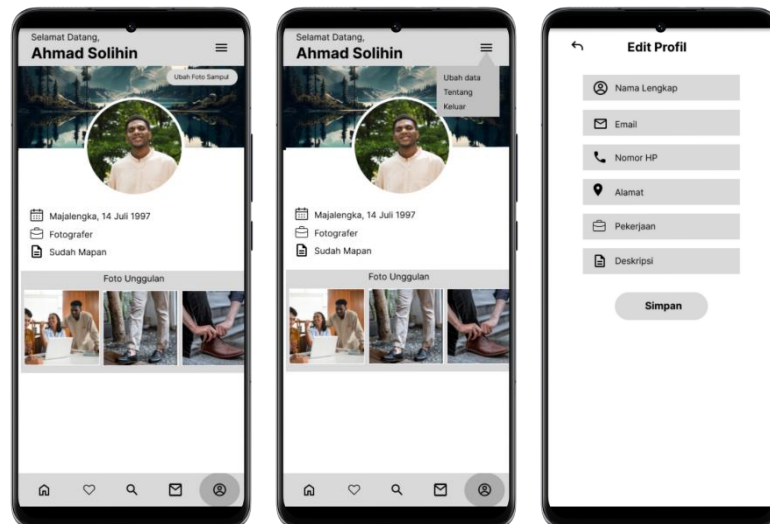
3.8.6 Tampilan Pencarian



Gambar 3. 9 Rancangan Tampilan Pencarian

Dalam aplikasi cari jodoh, fitur pencarian adalah salah satu komponen kunci yang membantu pengguna menemukan akun pengguna lain. Pada halaman ini pengguna dapat mencari dan menampilkan detail akun dari pengguna lain.

3.8.7 Tampilan Profil



Gambar 3. 10 Rancangan Tampilan Profil

Profil dalam aplikasi cari jodoh merupakan halaman yang sangat penting karena memuat informasi tentang pengguna secara lengkap. Di halaman ini, pengguna memiliki kesempatan untuk memperbarui data diri mereka dengan cara

yang mudah dan intuitif. Mereka dapat mengubah data pribadi seperti nama, usia, dan informasi kontak untuk memastikan bahwa profil mereka selalu akurat dan *up-to-date*. Selain itu, pengguna juga dapat mengganti foto profil mereka, yang sering kali merupakan gambar utama yang dilihat oleh calon pasangan. Tidak hanya itu, pengguna juga dapat mengubah foto sampul, yang memberikan konteks tambahan tentang diri mereka dan minat mereka.

Halaman profil juga menyediakan opsi bagi pengguna untuk keluar dari akun mereka dengan mudah. Setelah memilih opsi *logout*, pengguna akan diarahkan kembali ke halaman *login*. Ini memastikan bahwa sesi mereka ditutup dengan benar dan menjaga keamanan akun mereka. Proses keluar ini sangat penting untuk menjaga privasi pengguna, terutama jika mereka menggunakan perangkat bersama atau perangkat umum. Dengan cara ini, aplikasi memastikan bahwa hanya pengguna yang berwenang yang dapat mengakses akun mereka setiap kali mereka melakukan *login*.

3.8.8 Tampilan Map



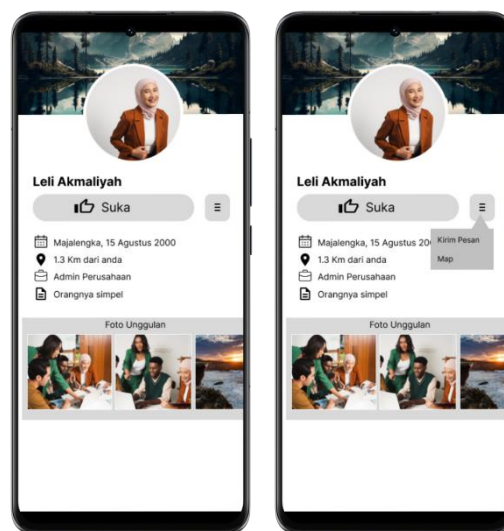
Gambar 3. 11 Rancangan Tampilan Map

Halaman *Map* dalam aplikasi cari jodoh adalah fitur yang sangat membantu pengguna dalam mencari calon pasangan berdasarkan lokasi geografis mereka. Dengan adanya fitur ini, pengguna dapat dengan mudah menemukan calon pasangan yang berada di sekitar mereka, sehingga memperbesar kemungkinan

untuk bertemu langsung. Fitur ini memanfaatkan teknologi GPS untuk secara akurat menentukan lokasi pengguna dan menampilkan calon pasangan yang berada dalam radius tertentu dari posisi mereka. Ini memudahkan pengguna untuk melihat siapa saja yang berada di dekatnya dan mempercepat proses pencarian pasangan.

Selain membantu pengguna menemukan calon pasangan yang dekat, fitur *Map* juga memberikan fleksibilitas dalam menjelajahi kemungkinan hubungan di daerah lain. Dengan cara ini, pengguna tidak hanya terbatas pada pencarian di lingkungan sekitar mereka, tetapi juga dapat memperluas jangkauan pencarian untuk menemukan pasangan yang cocok di lokasi yang berbeda. Fitur ini memberikan kemudahan dan opsi tambahan dalam proses pencarian pasangan, membuat pengalaman mencari jodoh menjadi lebih dinamis dan bermanfaat.

3.8.9 Tampilan Detail Pengguna



Gambar 3. 12 Rancangan Tampilan Detail Akun

Halaman detail pada aplikasi cari jodoh dirancang secara komprehensif untuk menampilkan informasi lengkap dan mendalam tentang pengguna lain, memberikan pengalaman yang lebih informatif dan personal. Informasi yang ditampilkan mencakup:

- **Nama Lengkap:** Menampilkan nama lengkap dari pengguna lain yang memungkinkan identifikasi yang lebih jelas dan personal.

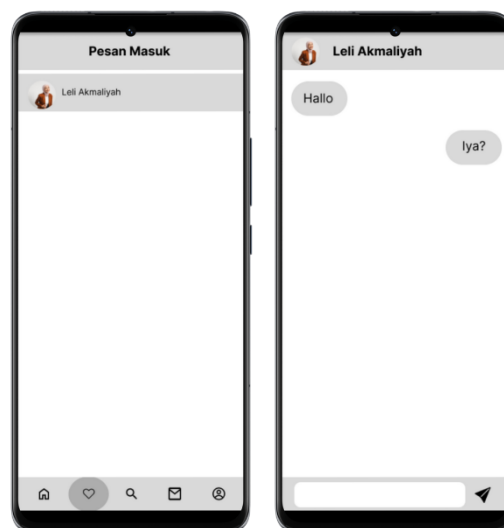
- Tempat Lahir: Menyediakan informasi mengenai tempat kelahiran pengguna, memberikan konteks mengenai latar belakang geografis mereka.
- Tanggal Lahir: Menyertakan data mengenai tanggal lahir pengguna untuk memberikan gambaran mengenai usia mereka, yang dapat membantu dalam pencocokan berdasarkan usia.
- Jarak dari Lokasi Anda: Menampilkan informasi mengenai jarak antara lokasi pengguna Anda dengan lokasi pengguna lain secara real-time, memungkinkan pengguna untuk memahami seberapa dekat mereka secara fisik dengan calon pasangan mereka.
- Pekerjaan: Memberikan deskripsi mengenai pekerjaan pengguna lain untuk memberikan informasi tambahan tentang latar belakang profesional mereka dan membantu dalam pencocokan berdasarkan minat dan tujuan karir.
- Deskripsi: Menyediakan penjelasan singkat mengenai diri pengguna, termasuk hobi, minat, dan kepribadian mereka, sehingga pengguna dapat lebih memahami kepribadian dan ketertarikan calon pasangan.
- Foto Profil: Menampilkan foto utama pengguna yang akan menjadi gambar pertama yang dilihat oleh pengguna lain, memberikan kesan pertama yang visual dan memudahkan identifikasi.
- Foto Sampul: Menggunakan gambar latar belakang yang memperindah tampilan halaman detail, memberikan nuansa visual yang lebih menarik dan personal.
- Foto Unggahan Lainnya: Menyediakan koleksi foto tambahan yang diunggah oleh pengguna untuk memberikan gambaran visual yang lebih lengkap tentang kehidupan dan aktivitas mereka.

Selain menampilkan informasi detail ini, halaman detail pengguna juga dilengkapi dengan fitur pengiriman pesan, yang memungkinkan pengguna untuk berkomunikasi secara langsung dengan pengguna lain. Fitur ini sangat penting untuk membangun interaksi dan hubungan yang lebih dekat, karena memungkinkan pengguna untuk mengirim pesan teks dengan cepat dan mudah. Fitur ini

mendukung komunikasi *real-time*, yang memfasilitasi pertukaran informasi dan dialog yang lancar antara pengguna.

Selain itu, halaman detail juga memiliki fitur melihat lokasi pengguna lain, yang dirancang untuk membantu pengguna mengetahui lokasi calon pasangan mereka dengan lebih jelas. Fitur ini memudahkan pengguna dalam merencanakan pertemuan, sehingga mereka dapat menentukan waktu dan tempat yang sesuai berdasarkan lokasi fisik calon pasangan. Dengan fitur ini, aplikasi tidak hanya memfasilitasi interaksi tetapi juga mendukung perencanaan pertemuan yang lebih efektif..

3.8.10 Tampilan Menu Pesan



Gambar 3. 13 Rancangan Tampilan Menu Pesan

Halaman pesan pada aplikasi cari jodoh adalah salah satu fitur inti yang memungkinkan pengguna untuk berkomunikasi secara langsung dengan pengguna lain. Fitur ini dirancang untuk mendukung interaksi yang efisien dan personal antara pengguna, dengan tujuan memperkuat hubungan yang sedang dibangun. Berikut adalah penjelasan detail mengenai halaman pesan:

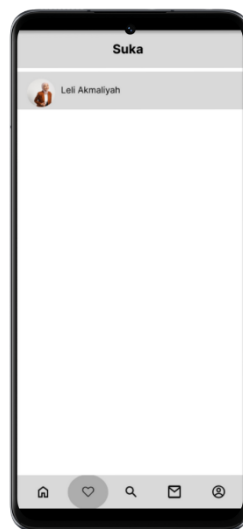
- Daftar Percakapan

Menampilkan daftar percakapan terbaru dengan pengguna lain. Setiap item dalam daftar ini mencakup foto profil, nama pengguna, dan waktu pesan terakhir untuk memudahkan identifikasi percakapan.

- Ruang Percakapan

Menampilkan percakapan antara pengguna dengan pengguna lain dalam bentuk urutan pesan. Di menu ini, pengguna dapat mengirim dan menerima pesan kepada pengguna lain.

3.8.11 Tampilan Menu Suka



Gambar 3. 14 Rancangan Tampilan Menu Suka

Halaman suka pada aplikasi cari jodoh adalah fitur yang memungkinkan pengguna untuk melihat daftar pengguna lain yang telah mereka sukai. Fitur ini memberikan cara yang mudah untuk mengakses kembali profil-profil yang menarik minat pengguna dan mempermudah dalam mengelola interaksi lebih lanjut.

3.9 Skenario Pengujian

Berikut ini adalah skenario pengujian *blackbox*, di mana pengujian dilakukan tanpa mengetahui struktur internal atau kode dari aplikasi yang diuji, fokus pada input dan output untuk memastikan fungsi sesuai dengan spesifikasi yang diharapkan:

Tabel 3. 8 Daftar Skenario Pengujian

Jenis Pengujian	Hasil yang diharapkan
	Halaman ditampilkan dengan benar

Jenis Pengujian	Hasil yang diharapkan
Halaman <i>Splash Screen</i>	Tampil beberapa detik dan beralih ke halaman berikutnya
Halaman Intro	Halaman ditampilkan dengan benar
	Pengguna dapat melewati halaman intro
	Pengguna dapat mengikuti <i>tour</i> pengenalan aplikasi dengan menekan <i>next</i> / <i>back</i>
	Halaman intro hanya tampil jika pengguna baru membuka aplikasi pertama kali
Halaman <i>Login</i>	Halaman ditampilkan dengan benar
	Validasi jika kolom <i>username</i> / sandi belum diisi maka akan muncul <i>error</i> “kolom tidak boleh kosong”
	Validasi jika <i>username</i> / sandi salah maka akan muncul <i>error</i> “Kredensial tidak cocok”
	Jika email dan sandi cocok dengan yang terdaftar pada <i>database</i> , maka akan diarahkan ke halaman <i>dashboard</i>
	Jika “ <i>remember me</i> ” dicentang, apabila email dan sandi betul maka saat membuka aplikasi dilain waktu tidak perlu <i>login</i> lagi
Halaman Daftar	Halaman ditampilkan dengan benar
	Validasi Jika pengguna mendaftar menggunakan email yang sudah terdaftar pada aplikasi maka akan <i>error</i> “Alamat email telah digunakan”
	Validasi Jika pengguna mendaftar menggunakan Nomor HP yang sudah terdaftar pada aplikasi maka akan <i>error</i> “Nomor HP telah digunakan”
	Validasi Jika pengguna mendaftar menggunakan <i>username</i> yang sudah terdaftar pada aplikasi maka akan <i>error</i> “ <i>Username</i> telah digunakan”

Jenis Pengujian	Hasil yang diharapkan
	Validasi jika ada kolom yang belum diisi maka pada kolom yang kosong akan muncul <i>error</i> “Kolom tidak boleh kosong”
	Validasi kolom kata sandi, jika keduanya tidak cocok maka akan muncul <i>error</i> “Kata sandi tidak cocok”
	Saat unggah foto profil, pengguna bisa memilih dan memotong foto kemudian diunggah ke server
Halaman Dashboard	Halaman ditampilkan dengan benar
	Pengguna dapat menyortir pengguna terdekat / terjauh
	Nama pengguna ditampilkan dengan benar
	Pengguna dapat merefresh halaman yang akan memperbarui lokasi pengguna tersebut
	pengguna dapat memilih pengguna yang disukai, kemudian akan diarahkan ke halaman detail
Halaman Pencarian	Halaman ditampilkan dengan benar
	Pengguna dapat mencari pengguna lain berdasarkan nama
	Jika pengguna telah menemukan akun yang dicari, maka akan diarahkan ke halaman detail
Halaman Suka	Halaman ditampilkan dengan benar
	Saat pengguna menyukai pengguna lain maka akan tampil di halaman suka
	Saat pengguna batal menyukai pengguna lain, maka list pengguna tersebut akan hilang di halaman suka
Halaman Pesan	Halaman ditampilkan dengan benar
	Pengguna dapat melihat pesan masuk berdasarkan nama
	Pengguna dapat menerima dan mengirim pesan teks
	Saat menekan <i>refresh</i> , maka halaman akan diperbarui
Halaman Profil	Halaman ditampilkan dengan benar

Jenis Pengujian	Hasil yang diharapkan
	Pengguna dapat mengubah profil berikut fotonya
	Pengguna dapat melakukan keluar akun
	Pengguna dapat mengakses halaman “ <i>about</i> ” yang menjelaskan versi dan <i>developer</i> aplikasi
Halaman Map	Peta ditampilkan dengan benar
	Pengguna dapat melihat lokasi dia dan pengguna lain
Halaman Detail	Halaman ditampilkan dengan benar
	Pengguna dapat melihat profil lengkap pengguna lain
	Pengguna dapat melakukan aksi suka pada pengguna lain
	Pengguna dapat melakukan aksi batal suka pada pengguna lain
	Pengguna dapat mengirim pesan ke pengguna lain
	Pengguna dapat mengakses <i>map</i>
	Pengguna dapat memperbesar foto pengguna lain

BAB IV

IMPLEMENTASI DAN ANALISA HASIL UJI COBA

4.1 Implementasi Sistem

4.1.1 Implementasi Tampilan *Interface*

1) Tampilan *Splashscreen*



Gambar 4. 1 Tampilan *Splashscreen*

Splashscreen menampilkan logo dan *progressbar loading* sebelum masuk ke halaman utama aplikasi. Ini memberikan waktu bagi aplikasi untuk memuat sumber daya yang diperlukan sehingga pengguna dapat menggunakan aplikasi dengan lancar setelah halaman utama terbuka.

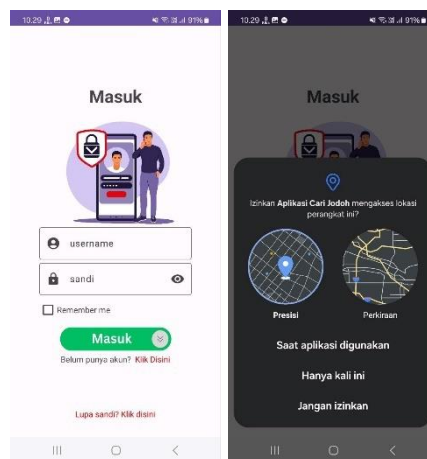
2) Tampilan Intro Aplikasi



Gambar 4. 2 Tampilan Intro

Intro aplikasi ditampilkan hanya saat pertama kali pengguna membuka aplikasi, memberikan panduan singkat tentang cara menggunakan berbagai fitur utama dengan jelas dan mudah dipahami. Intro ini mencakup tutorial langkah demi langkah yang memandu pengguna baru melalui proses pendaftaran, navigasi aplikasi, dan penggunaan fitur-fitur penting seperti pencarian calon pasangan dan pengiriman pesan. Dengan cara ini, pengguna dapat dengan cepat memahami bagaimana memanfaatkan aplikasi secara efektif.

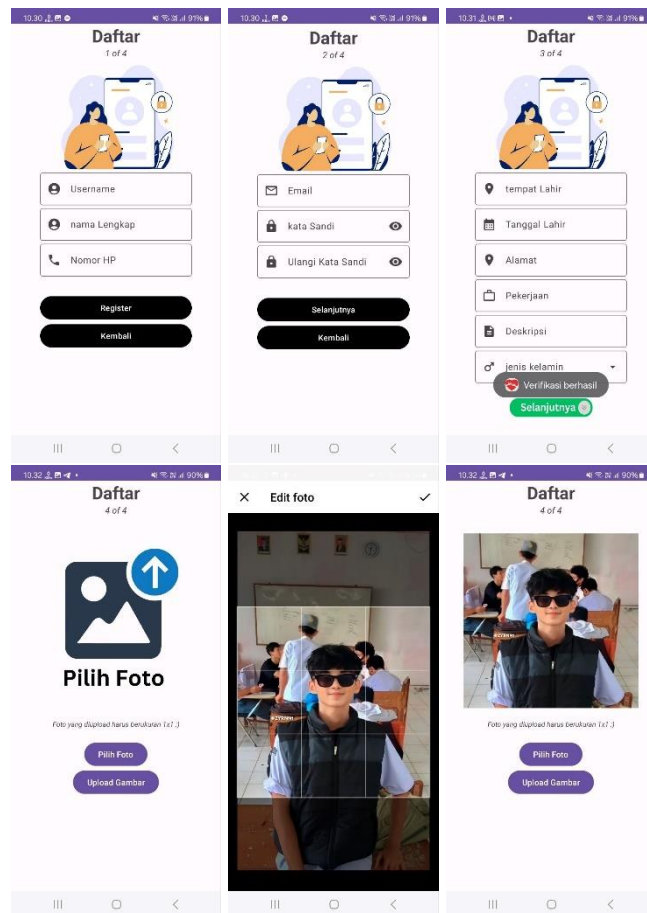
3) Tampilan *Login*



Gambar 4. 3 Tampilan *Login*

Halaman *login* dirancang untuk memungkinkan pengguna mengakses akun mereka dengan berbagai metode autentikasi. Pengguna dapat masuk menggunakan *username* dan kata sandi mereka, atau memilih opsi *login* menggunakan nomor telepon atau alamat email yang telah terdaftar. Halaman ini juga menyertakan opsi "*remember me*" yang memungkinkan pengguna untuk tetap *login* saat kembali membuka aplikasi di kemudian hari. Dengan fitur ini, pengalaman *login* menjadi lebih mudah dan cepat, memberikan kenyamanan tambahan bagi pengguna yang sering mengakses aplikasi.

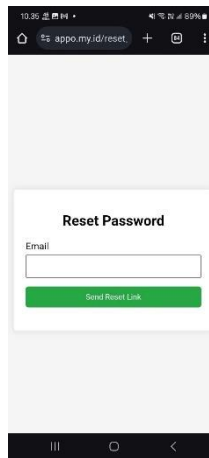
4) Tampilan Daftar



Gambar 4. 4 Tampilan Daftar

Proses pendaftaran akun baru terdiri dari empat tahap. Tahap pertama adalah mengisi nama pengguna, nama lengkap, dan nomor handphone. Selanjutnya, pengguna harus mengisi alamat email dan kata sandi, dengan pengisian kata sandi dilakukan dua kali untuk konfirmasi agar tidak lupa. Pada tahap ketiga, pengguna diminta mengisi tempat lahir, tanggal lahir, alamat, pekerjaan, deskripsi, dan jenis kelamin, serta memberikan izin lokasi untuk menyimpan data lokasi pengguna. Terakhir, pengguna harus mengunggah setidaknya satu foto pribadi sebagai foto profil. Setelah semua tahap ini selesai, pengguna akan diarahkan ke halaman *login*.

5) Tampilan Lupa Sandi



Gambar 4. 5 Tampilan *Reset* Sandi

Halaman lupa sandi memungkinkan pengguna mereset kata sandi mereka dengan memasukkan email terdaftar. Jika email valid, token *reset* akan dikirimkan ke email tersebut. Pengguna kemudian memasukkan token di halaman *reset* sandi dan membuat kata sandi baru. Setelah diverifikasi, kata sandi baru dapat digunakan untuk *login* ke aplikasi.

6) Tampilan Menu *Dashboard*



Gambar 4. 6 Tampilan *Dashboard*

Dashboard dalam aplikasi cari jodoh adalah halaman utama yang dirancang untuk menampilkan data pengguna lain. Pada halaman ini, pengguna dapat dengan mudah melihat profil calon pasangan

potensial yang terdaftar di aplikasi. Untuk meningkatkan efisiensi pencarian, pengguna dapat memanfaatkan fitur filter berdasarkan jarak, yang memungkinkan mereka menemukan calon pasangan yang terdekat maupun yang terjauh. Selain itu, pengguna dapat mengakses detail lebih lanjut dari setiap calon pasangan, seperti informasi pribadi, minat, dan foto.

7) Tampilan Menu Pencarian



Gambar 4. 7 Tampilan Pencarian

Fitur pencarian membantu pengguna menemukan dan menjelajahi akun pengguna lain dalam aplikasi. Dengan menggunakan fitur ini, pengguna dapat mencari akun lain dengan memasukkan nama pengguna pada kolom pencarian.

Setelah melakukan pencarian, pengguna dapat melihat hasil yang relevan dan mengakses halaman detail untuk setiap akun pengguna yang ditemukan. Halaman detail ini memungkinkan pengguna untuk melihat informasi lengkap tentang pengguna lain, termasuk nama lengkap, tempat lahir, tanggal lahir, pekerjaan, deskripsi, serta foto profil, foto sampul, dan foto unggahan lainnya. Fitur pencarian ini dirancang untuk memudahkan pengguna dalam menemukan dan mengeksplorasi profil yang menarik minat mereka dengan efisien..

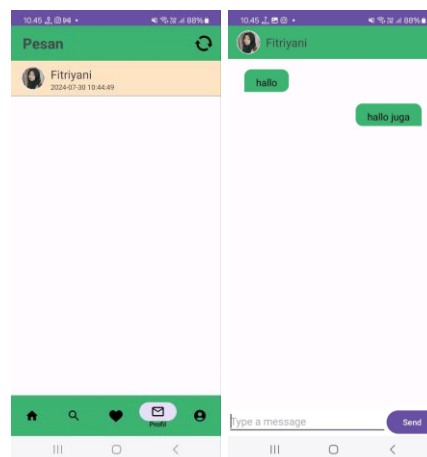
8) Tampilan Menu Suka



Gambar 4. 8 Tampilan Menu Suka

Halaman suka memudahkan pengguna untuk melihat dan mengelola daftar profil yang telah mereka sukai, termasuk foto profil dan nama lengkap. Pengguna dapat mengklik profil untuk melihat detail lebih lanjut seperti nama lengkap, tempat lahir, tanggal lahir, pekerjaan, deskripsi, serta berbagai foto, yang memudahkan mereka dalam mengelola interaksi dan komunikasi di aplikasi.

9) Tampilan Menu Pesan

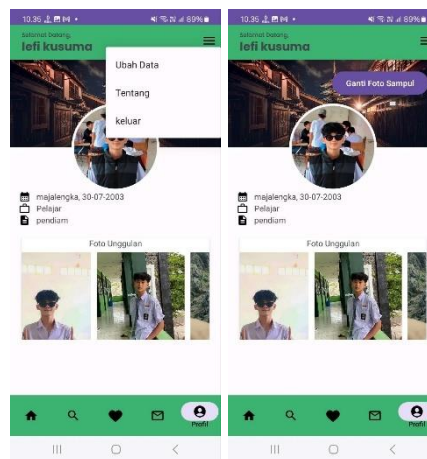


Gambar 4. 9 Tampilan Pesan

Halaman chat dalam aplikasi cari jodoh memungkinkan komunikasi langsung antara pengguna, mendukung interaksi yang efisien dan personal. Fitur ini terdiri dari dua bagian utama, yaitu daftar

percakapan dan ruang percakapan. Daftar percakapan menampilkan percakapan terbaru, lengkap dengan foto profil, nama pengguna, dan potongan pesan terakhir, sehingga pengguna dapat dengan mudah melihat dan mengakses percakapan terkini mereka. Sementara itu, ruang percakapan menampilkan urutan pesan secara mendetail, mencakup teks, gambar, video, atau file lainnya, lengkap dengan tanda waktu pengiriman. Dengan fitur ini, pengguna dapat menikmati pengalaman komunikasi yang lancar dan terorganisir, memastikan bahwa semua interaksi mereka terdokumentasi dengan baik dan mudah diakses kapan saja.

10) Tampilan Menu Profil



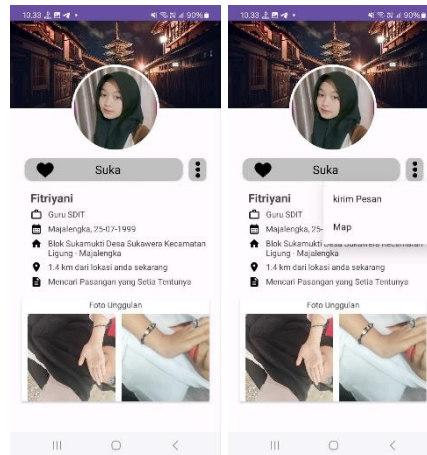
Gambar 4. 10 Tampilan Menu Profil

Halaman profil menyajikan berbagai informasi tentang pengguna, termasuk data pribadi, foto profil, dan foto sampul. Pengguna memiliki opsi untuk memperbarui data diri mereka, mengganti foto profil, dan mengubah foto sampul sesuai keinginan. Fitur ini memungkinkan pengguna untuk menyesuaikan tampilan dan informasi mereka agar sesuai dengan preferensi pribadi.

Selain itu, halaman profil juga menyediakan opsi untuk keluar dari akun. Setelah pengguna memilih untuk *logout*, mereka akan diarahkan kembali ke halaman *login*, di mana mereka dapat masuk kembali jika diperlukan. Fitur ini memastikan keamanan akun dan

memberikan akses yang mudah untuk perubahan data serta keluar dari sistem.

11) Tampilan Detail Akun



Gambar 4. 11 Tampilan Detail Akun

Halaman ini menyajikan informasi lengkap tentang pengguna lain, mencakup berbagai detail yang membantu pengguna memahami lebih dalam tentang profil tersebut. Informasi yang ditampilkan meliputi nama lengkap, tempat lahir, tanggal lahir, jarak dari lokasi pengguna, pekerjaan, deskripsi singkat mengenai kepribadian dan minat, serta berbagai foto yang mencakup foto profil, foto sampul, dan foto unggahan lainnya.

Selain itu, halaman ini juga menyediakan fitur tambahan yang memungkinkan pengguna untuk mengirim pesan langsung kepada pengguna lain, memfasilitasi komunikasi yang cepat dan mudah. Pengguna juga dapat melihat lokasi pengguna lain, yang membantu dalam merencanakan pertemuan dengan lebih efisien. Dengan fitur-fitur ini, halaman detail pengguna memberikan alat yang lengkap untuk berinteraksi dan mengelola hubungan dalam aplikasi.

12) Tampilan *Map*

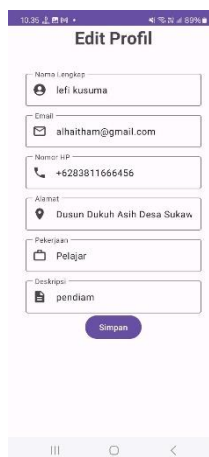


Gambar 4. 12 Tampilan *Map*

Halaman *Map* dalam aplikasi cari jodoh memungkinkan pengguna mencari calon pasangan berdasarkan lokasi geografis. Dengan fitur ini, pengguna dapat melihat pengguna lain yang berada di sekitar area tertentu, memperluas jangkauan pencarian mereka, atau menemukan calon pasangan yang dekat dengan lokasi mereka.

Fitur ini memanfaatkan teknologi GPS untuk menampilkan calon pasangan dalam radius yang dapat disesuaikan oleh pengguna, memberikan fleksibilitas dalam menjelajahi kemungkinan hubungan di berbagai daerah.

13) Tampilan Ubah Data



Gambar 4. 13 Tampilan Ubah Data

Tampilan ubah data memungkinkan pengguna untuk mengubah informasi pribadi mereka yang telah terdaftar di aplikasi. Fitur ini dirancang untuk memberikan fleksibilitas kepada pengguna dalam memperbarui data mereka sesuai kebutuhan. Pengguna dapat mengakses halaman ini dari menu profil dan melakukan perubahan yang diperlukan, seperti nama pengguna, nama lengkap, alamat email, nomor *handphone*, alamat, pekerjaan dan deskripsi. Selain itu, pengguna juga dapat mengganti atau memperbarui foto profil mereka dengan mengunggah gambar baru.

14) Tampilan Tentang

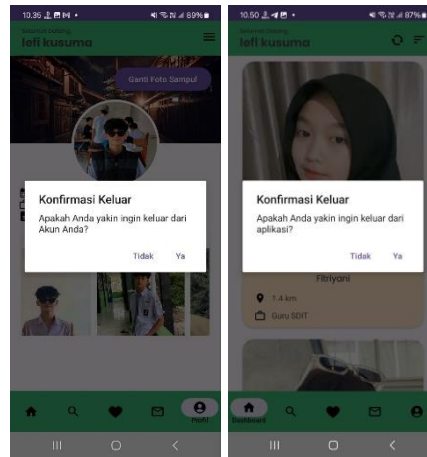


Gambar 4. 14 Tampilan Tentang

Halaman Tentang pada aplikasi menyediakan informasi penting mengenai aplikasi dan pengembangnya. Di halaman ini, pengguna dapat melihat nomor versi aplikasi saat ini, yang membantu mereka mengetahui apakah mereka menggunakan versi terbaru dan mendapatkan pembaruan terbaru. Selain itu, halaman ini juga menyertakan informasi tentang pengembang aplikasi, seperti nama pengembang atau perusahaan, serta kontak yang dapat dihubungi. Biasanya, informasi kontak mencakup alamat email, nomor telepon, atau tautan ke situs web dan media sosial. Halaman ini bertujuan untuk memberikan transparansi mengenai aplikasi dan memudahkan

pengguna untuk menghubungi pengembang jika mereka memiliki pertanyaan, masalah, atau umpan balik.

15) Tampilan Keluar Akun Dan Keluar Aplikasi



Gambar 4. 15 Tampilan Keluar Aplikasi

Tampilan keluar akun dan keluar aplikasi menyediakan opsi bagi pengguna untuk mengelola status masuk mereka dan keluar dari aplikasi sepenuhnya. Fitur ini dirancang untuk memberikan pengguna kontrol penuh atas penggunaan aplikasi dan keamanan akun mereka. Terdapat dua opsi utama, yaitu keluar akun dan keluar aplikasi.

Tombol keluar akun memungkinkan pengguna untuk keluar dari akun mereka dan akan mengarahkan mereka kembali ke halaman *login*, berguna untuk mengamankan akun saat tidak digunakan atau saat ingin beralih ke akun lain. Tombol keluar aplikasi memungkinkan pengguna untuk menutup aplikasi sepenuhnya, menghentikan semua aktivitas aplikasi, dan memastikan aplikasi tidak berjalan di latar belakang.

Dengan fitur ini, pengguna dapat dengan mudah mengelola status masuk mereka, menjaga keamanan akun, dan memastikan aplikasi hanya berjalan saat diperlukan.

4.1.2 Implementasi Algoritma Haversine

1) Rumus Haversine

Rumus Haversine yang digunakan untuk menghitung jarak antara dua titik pada permukaan bumi adalah:

$$\alpha = \sin^2\left(\frac{(\Delta lat)}{2}\right) + \cos(lat1) \times \cos(lat2) \times \sin^2\left(\frac{(\Delta lon)}{2}\right)$$

$$c = 2 \cdot \text{atan}^2(\sqrt{\alpha}, \sqrt{1-\alpha})$$

$$d = R \cdot c$$

Keterangan :

d = Jarak

r = Jari – jari bumi (6,371 km)

Δlat = Besar perubahan latitude

Δlon = Besar perubahan longitude

2) Penerapan

Berikut adalah script algoritma Haversine dalam bahasa Java yang dapat digunakan untuk menghitung jarak antara dua titik berdasarkan koordinat geografis mereka.

```
public Double getDistance(double lttarget, double lgtarget,
    double ltsaya, double lgsaya) {
    Double pi = 3.14159265358979;
    Double lat1 = lttarget;
    Double lon1 = lgtarget;
    Double lat2 = ltsaya;
    Double lon2 = lgsaya;
    Double R = Double.valueOf(6371);
    Double latRad1 = lat1 * (pi / 180);
    Double latRad2 = lat2 * (pi / 180);
    Double deltaLatRad = (lat2 - lat1) * (pi / 180);
    Double deltaLonRad = (lon2 - lon1) * (pi / 180);
    Double a = Math.sin(deltaLatRad / 2) *
        Math.sin(deltaLatRad / 2) + Math.cos(latRad1) *
        Math.cos(latRad2) * Math.sin(deltaLonRad / 2) *
        Math.sin(deltaLonRad / 2);
    Double c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1 -
        a));
    Double s = R * c; // hasil jarak dalam meter
    Double mm = BigDecimal.valueOf(s).setScale(1,
        BigDecimal.ROUND_HALF_DOWN).doubleValue();
    return mm;
}
```

4.1.3 Implementasi Filter Jarak

1) Rumus Filter Jarak

Pada penelitian ini, filter jarak diimplementasikan dengan menggunakan metode pengurutan objek berdasarkan jarak dari lokasi pengguna. Proses ini melibatkan pengurutan daftar objek lokasi

berdasarkan jarak masing-masing objek dari lokasi pengguna. Kode Java berikut menggambarkan cara pengurutan ini dilakukan:

```
Collections.sort(locations, new Comparator<Location>()
{
    @Override
    public int compare(Location o1, Location o2) {
        return Double.compare(o1.distance,
            o2.distance);
    }
});
```

Penjelasan:

- Collections.sort
Metode dari Java Collections Framework yang digunakan untuk mengurutkan elemen dalam koleksi.
- Comparator<Location>
Antarmuka yang digunakan untuk menentukan urutan objek Location. Pada implementasi ini, Comparator digunakan untuk membandingkan dua objek Location.
- Double.compare(o1.distance, o2.distance)
Membandingkan jarak dari dua objek Location (o1 dan o2) dan mengembalikan nilai yang menunjukkan urutan mereka berdasarkan jarak. Jika o1 lebih dekat daripada o2, maka o1 akan ditempatkan sebelum o2.

2) Penerapan

Untuk penerapan filter jarak pada data yang sudah disaring, kode berikut digunakan untuk mengurutkan data berdasarkan jarak yang telah disimpan dalam atribut dari objek ModelClass:

```
Collections.sort(filteredData, new
Comparator<ModelClass>() {
    @Override
    public int compare(ModelClass o1, ModelClass o2) {
        double distance1 = Double.parseDouble(o1.getV());
        double distance2 = Double.parseDouble(o2.getV());
        return Double.compare(distance1, distance2);
    }
});
```

Penjelasan:

- `filteredData`

Daftar data yang telah difilter sebelumnya berdasarkan kriteria tertentu.

- `ModelClass`

Kelas yang digunakan untuk menyimpan data yang relevan. Atribut `V` dalam kelas ini menyimpan informasi jarak dalam format string.

- `Double.parseDouble(o1.getV())`

Mengonversi nilai jarak yang disimpan sebagai string dalam atribut `V` menjadi tipe data double untuk perbandingan.

- `Double.compare(distance1, distance2)`

Membandingkan dua nilai jarak dan menentukan urutan mereka dalam daftar berdasarkan jarak yang lebih kecil akan ditempatkan lebih dulu.

4.2 Pengujian Sistem

4.2.1 Pengujian *Blackbox*

Berikut adalah hasil pengujian *blackbox* yang telah diuji oleh salah satu responden penelitian. Rincian perangkat yang digunakan dalam pengujian adalah sebagai berikut:

Tabel 4. 1 Perangkat Pengujian *Blackbox*

Nama	Yusuf Nomi Syaripudin
Perangkat yang digunakan	SmartPhone
Merk dan Model	Samsung Galaxy A50
Sistem Operasi	Android 11
Versi Aplikasi	1.0
Koneksi Internet	Wi-Fi 30 MBps
Kondisi Baterai	80%
Mode Tampilan	Mode Terang
Kapasitas RAM	4 GB

Penyimpanan	64 GB
Tanggal Pengujian	29 Juli 2024
Nomor HP	083825525333

Hasil pengujian meliputi berbagai aspek fungsionalitas aplikasi sesuai dengan skenario yang telah ditetapkan dalam metode pengujian *blackbox*. Berikut adalah hasil dari pengujian *blackbox*:

Tabel 4. 2 Hasil Pengujian *Blackbox*

Jenis Pengujian	Hasil yang diharapkan	Keterangan	
		Sesuai	Tidak
Halaman <i>Splash Screen</i>	Halaman ditampilkan dengan benar	√	
	Tampil beberapa detik dan beralih ke halaman berikutnya	√	
Halaman Intro	Halaman ditampilkan dengan benar	√	
	Pengguna dapat melewati halaman intro	√	
	Pengguna dapat mengikuti <i>tour</i> pengenalan aplikasi dengan menekan <i>next</i> / <i>back</i>	√	
	Halaman intro hanya tampil jika pengguna baru membuka aplikasi pertama kali	√	
Halaman <i>Login</i>	Halaman ditampilkan dengan benar	√	
	Validasi jika kolom <i>username</i> / sandi belum diisi maka akan muncul <i>error</i> “kolom tidak boleh kosong”	√	

Jenis Pengujian	Hasil yang diharapkan	Keterangan	
		Sesuai	Tidak
	Validasi jika <i>username</i> / sandi salah maka akan muncul <i>error</i> “Kredensial tidak cocok”	√	
	Jika email dan sandi cocok dengan yang terdaftar pada <i>database</i> , maka akan diarahkan ke halaman <i>dashboard</i>	√	
	Jika “ <i>remember me</i> ” dicentang, apabila email dan sandi betul maka saat membuka aplikasi dilain waktu tidak perlu <i>login</i> lagi	√	
Halaman Daftar	Halaman ditampilkan dengan benar	√	
	Validasi Jika pengguna mendaftar menggunakan email yang sudah terdaftar pada aplikasi maka akan <i>error</i> “Alamat email telah digunakan”	√	
	Validasi Jika pengguna mendaftar menggunakan Nomor HP yang sudah terdaftar pada aplikasi maka akan <i>error</i> “Nomor HP telah digunakan”	√	
	Validasi Jika pengguna mendaftar menggunakan <i>username</i> yang sudah terdaftar pada aplikasi maka akan <i>error</i> “ <i>Username</i> telah digunakan”	√	

Jenis Pengujian	Hasil yang diharapkan	Keterangan	
		Sesuai	Tidak
	Validasi jika ada kolom yang belum diisi maka pada kolom yang kosong akan muncul <i>error</i> “Kolom tidak boleh kosong”	√	
	Validasi kolom kata sandi, jika keduanya tidak cocok maka akan muncul <i>error</i> “Kata sandi tidak cocok”	√	
	Saat unggah foto profil, pengguna bisa memilih dan memotong foto kemudian diunggah ke server	√	
Halaman Dashboard	Halaman ditampilkan dengan benar	√	
	Pengguna dapat menyortir pengguna terdekat / terjauh	√	
	Nama pengguna ditampilkan dengan benar	√	
	Pengguna dapat merefresh halaman yang akan memperbarui lokasi pengguna tersebut	√	
	pengguna dapat memilih pengguna yang disukai, kemudian akan diarahkan ke halaman detail	√	
Halaman Pencarian	Halaman ditampilkan dengan benar	√	
	Pengguna dapat mencari pengguna lain berdasarkan nama	√	

Jenis Pengujian	Hasil yang diharapkan	Keterangan	
		Sesuai	Tidak
	Jika pengguna telah menemukan akun yang dicari, maka akan diarahkan ke halaman detail	√	
Halaman Suka	Halaman ditampilkan dengan benar	√	
	Saat pengguna menyukai pengguna lain maka akan tampil di halaman suka	√	
	Saat pengguna batal menyukai pengguna lain, maka list pengguna tersebut akan hilang di halaman suka	√	
Halaman Pesan	Halaman ditampilkan dengan benar	√	
	Pengguna dapat melihat pesan masuk berdasarkan nama	√	
	Pengguna dapat menerima dan mengirim pesan teks	√	
	Saat menekan <i>refresh</i> , maka halaman akan diperbarui	√	
Halaman Profil	Halaman ditampilkan dengan benar	√	
	Pengguna dapat mengubah profil berikut fotonya	√	
	Pengguna dapat melakukan keluar akun	√	
	Pengguna dapat mengakses halaman “ <i>about</i> ” yang	√	

Jenis Pengujian	Hasil yang diharapkan	Keterangan	
		Sesuai	Tidak
	menjelaskan versi dan <i>developer</i> aplikasi		
Halaman Map	Peta ditampilkan dengan benar	√	
	Pengguna dapat melihat lokasi dia dan pengguna lain	√	
Halaman Detail	Halaman ditampilkan dengan benar	√	
	Pengguna dapat melihat profil lengkap pengguna lain	√	
	Pengguna dapat melakukan aksi suka pada pengguna lain	√	
	Pengguna dapat melakukan aksi batal suka pada pengguna lain	√	
	Pengguna dapat mengirim pesan ke pengguna lain	√	
	Pengguna dapat mengakses <i>map</i>	√	
	Pengguna dapat memperbesar foto pengguna lain	√	

Hasil uji *blackbox* yang dilakukan oleh responden Yusuf Nomi Syaripudin menunjukkan bahwa aplikasi berfungsi dengan baik pada perangkat yang diuji, yaitu Samsung Galaxy M50 dengan sistem operasi Android 12. Dalam pengujian login, pengguna memasukkan email dan kata sandi yang benar dan berhasil masuk ke aplikasi tanpa masalah, dengan notifikasi berhasil ditampilkan. Pada skenario registrasi, pendaftaran pengguna baru dengan email yang valid berjalan lancar, dan pengguna menerima email verifikasi sesuai yang diharapkan.

Pengujian pengiriman pesan menunjukkan bahwa pesan dapat terkirim dengan cepat dan ditampilkan di layar penerima tanpa penundaan. Fitur pencarian

pengguna juga berfungsi dengan baik, di mana hasil pencarian ditampilkan dengan akurat dan cepat. Saat menguji fitur logout, pengguna berhasil keluar dari aplikasi dan kembali ke layar login tanpa kendala.

Responsivitas aplikasi diuji dengan menavigasi melalui berbagai halaman, yang menunjukkan bahwa aplikasi berjalan lancar tanpa lag atau crash, dan semua halaman dimuat dengan cepat. Pengujian mode tampilan mengungkap bahwa pergantian dari mode terang ke mode gelap dan sebaliknya berjalan lancar tanpa masalah visual atau keterlambatan.

Secara keseluruhan, hasil pengujian blackbox menunjukkan bahwa aplikasi berfungsi dengan baik dan memenuhi ekspektasi fungsionalitas sesuai dengan skenario yang telah ditetapkan. Semua fitur utama bekerja dengan lancar, dan tidak ditemukan masalah signifikan yang dapat mengganggu pengalaman pengguna.

4.2.2 Pengujian Akurasi Jarak

Tahap pengujian akurasi algoritma Haversine dilakukan untuk memastikan bahwa algoritma yang digunakan dalam aplikasi dapat menghitung jarak antara dua titik geografis dengan tingkat akurasi yang tinggi. Pengujian ini terbagi menjadi dua tahap. Pada tahap pertama, dilakukan pengujian terhadap lokasi yang terkait dengan pengguna perempuan, dan pada tahap kedua, dilakukan pengujian terhadap lokasi pengguna laki-laki. Setiap tahap mencakup perbandingan hasil perhitungan jarak yang dihasilkan oleh aplikasi dengan perhitungan manual menggunakan rumus Haversine, untuk memastikan bahwa hasilnya konsisten dan akurat. Berikut adalah hasil pengujian yang dilakukan:

Tabel 4. 3 Pengujian Akurasi Perempuan

	Lokasi Uji	Lokasi Fitri	Lokasi Leni	Lokasi Annisah
Latitude	-6.65612380	-6.66440495	-6.65764965	-6.66440224
Longitude	108.29867110	108.29040988	108.28920167	108.29040433
Hasil Manual		1,30 km	1.06 km	1.30 km

	Lokasi Uji	Lokasi Fitri	Lokasi Leni	Lokasi Annisah
Hasil Sistem		 Fitriyani 1.3 km Guru SDIT	 Leni Anggraeni 1.1 km Wiraswasta	 Anisah Jamilah 1.3 km Mahasiswa

Jarak dari lokasi uji ke lokasi fitri

- 1) Konversi Derajat ke Radian

$$lat1 = -6.6561238 \times \frac{\pi}{180} = -0.1161712757 \text{ rad}$$

$$lon1 = 108.2986711 \times \frac{\pi}{180} = 1.8901683862 \text{ rad}$$

$$lat2 = -6.66440495 \times \frac{\pi}{180} = -0.1163158091 \text{ rad}$$

$$lon2 = 108.29040988 \times \frac{\pi}{180} = 1.8900242007 \text{ rad}$$

- 2) Hitung Perbedaan Lintang dan Bujur

$$\Delta lat = lat2 - lat1$$

$$\Delta lat = -0.1163158091 - (-0.1161712757)$$

$$\Delta lat = -0.0001445333 \text{ rad}$$

$$\Delta lon = lon2 - lon1$$

$$\Delta lon = 1.8900242007 - 1.8901683862$$

$$\Delta lon = -0.0001441855 \text{ rad}$$

- 3) Hitung Nilai 'a'

$$\alpha = \sin^2 \left(\frac{-0.0001445333}{2} \right) + \cos(-0.1161712757) \times$$

$$\cos(-0.1163158091) \times \sin^2 \left(\frac{-0.0001441855}{2} \right)$$

$$\alpha = -0.0000722667^2 + 0.9932597029 \times 0.9932429397 \times -0.0000720927^2$$

$$\alpha = 0.0000000103$$

- 4) Hitung Nilai 'c'

$$c = 2 \times \operatorname{atan}^2(\sqrt{0.0000000103}, \sqrt{1 - 0.0000000103})$$

$$c = 2 \times \operatorname{atan}^2(0.0001017346, 0.9999999948)$$

$$c = 0.0002034691$$

- 5) Hitung Jarak

$$\text{distance} = 6371 \times 0.0002034691 = 1.30 \text{ km}$$

Jadi, jarak antara lokasi uji dan lokasi fitri adalah sekitar 1.30 kilometer.

Jarak dari lokasi uji ke lokasi Leni

- 1) Konversi Derajat ke Radian

$$\text{lat1} = -6.65764965 \times \frac{\pi}{180} = -0.1161979068 \text{ rad}$$

$$\text{lon1} = 108.28920167 \times \frac{\pi}{180} = 1.8900031135 \text{ rad}$$

$$\text{lat2} = -6.65612380 \times \frac{\pi}{180} = -0.1161712757 \text{ rad}$$

$$\text{lon2} = 108.29867110 \times \frac{\pi}{180} = 1.8901683862 \text{ rad}$$

- 2) Hitung Perbedaan Lintang dan Bujur

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

$$\Delta \text{lat} = -0.1161712757 - (-0.1161979068)$$

$$\Delta \text{lat} = 0.0000266311 \text{ rad}$$

$$\Delta \text{lon} = \text{lon2} - \text{lon1}$$

$$\Delta \text{lon} = 1.88932799 - 1.88948579$$

$$\Delta \text{lon} = 0.0001652727 \text{ rad}$$

- 3) Hitung Nilai 'a'

$$\alpha = \sin^2 \left(\frac{0.0000266311}{2} \right) + \cos(-0.1161979068) \times$$

$$\cos(-0.1161712757) \times \sin^2 \left(\frac{0.0001652727}{2} \right)$$

$$\alpha = \sin^2 (0.0000133156^2) + \cos(0.9932566158) \times$$

$$\cos(0.9932597029) \times \sin^2(-0.00007890)$$

$$\alpha = 0.0000000069$$

- 4) Hitung Nilai 'c'

$$c = 2 \times \operatorname{atan}^2(\sqrt{0.0000000069}, \sqrt{1 - 0.0000000069})$$

$$c = 2 \times \operatorname{atan}^2(0.0000831523, 0.9999999965)$$

$$c = 0.0001663046$$

- 5) Hitung Jarak

$$\text{distance} = 6371 \times 0.0001663046 = 1.06 \text{ km}$$

Jadi, jarak antara lokasi uji dan lokasi Leni adalah sekitar 1.06 kilometer.

Jarak dari lokasi uji ke lokasi Anisah

- 1) Konversi Derajat ke Radian

$$\text{lat1} = -6.66440224 \times \frac{\pi}{180} = -0.1163157618 \text{ rad}$$

$$\text{lon1} = 108.29040433 \times \frac{\pi}{180} = 1.8900241039 \text{ rad}$$

$$\text{lat2} = -6.6561238 \times \frac{\pi}{180} = -0.1161712757 \text{ rad}$$

$$\text{lon2} = 108.2986711 \times \frac{\pi}{180} = 1.8901683862 \text{ rad}$$

- 2) Hitung Perbedaan Lintang dan Bujur

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

$$\Delta \text{lat} = -0.1161712757 - (-0.1163157618)$$

$$\Delta \text{lat} = 0.0001444860 \text{ rad}$$

$$\Delta \text{lon} = \text{lon2} - \text{lon1}$$

$$\Delta \text{lon} = 1.8901683862 - 1.8900241039$$

$$\Delta \text{lon} = 0.0001442824 \text{ rad}$$

- 3) Hitung Nilai 'a'

$$\alpha = \sin^2\left(\frac{0.0001444860}{2}\right) + \cos(-0.1163157618) \times$$

$$\cos(-0.1161712757) \times \sin^2\left(\frac{0.0001442824}{2}\right)$$

$$\alpha = 0.0000722430^2 + 0.9932429452 \times 0.9932597029 \times 0.0000721412^2$$

$$\alpha = 0.0000000104$$

- 4) Hitung Nilai 'c'

$$c = 2 \times \operatorname{atan}^2(\sqrt{0.0000000104}, \sqrt{1 - 0.0000000104})$$

$$c = 2 \times \operatorname{atan}^2(0.0001017516, 0.9999999948)$$

$$c = 0.0002035033$$

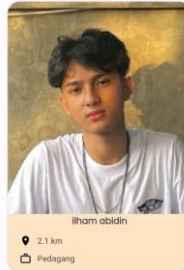
5) Hitung Jarak

$$\text{distance} = 6371 \times 0.0002034691 = 1.30 \text{ km}$$

Jadi, jarak antara lokasi uji dan lokasi Annisah adalah sekitar 1.30 kilometer.

Setelah melakukan pengujian terhadap lokasi pengguna perempuan, langkah selanjutnya adalah menguji lokasi untuk pengguna laki-laki. Pengujian ini mengikuti prosedur yang sama dengan pengujian sebelumnya, di mana hasil perhitungan jarak oleh aplikasi dibandingkan dengan hasil perhitungan manual menggunakan rumus Haversine. Berikut adalah hasil pengujian untuk lokasi pengguna laki-laki:

Tabel 4. 4 Pengujian Akurasi Laki-laki

	Lokasi Uji	Lokasi Saepudin	Lokasi Hamid	Lokasi Ilham
Latitude	-6.66440495	-6.65265770	-6.68023480	-6.65291340
Longitude	108.29040988	108.28246340	108.30236430	108.30570870
Hasil Manual		1,57 km	2,2 km	2.12 km
Hasil Sistem				

Jarak dari lokasi uji ke lokasi Saepudin

1) Konversi Derajat ke Radian

$$\text{lat1} = -6.6526577 \times \frac{\pi}{180} = -0.1161107809 \text{ rad}$$

$$\text{lon1} = 108.2824634 \times \frac{\pi}{180} = 1.8898855085 \text{ rad}$$

$$\text{lat2} = -6.66440495 \times \frac{\pi}{180} = -0.1163158091 \text{ rad}$$

$$lon2 = 108.29040988 \times \frac{\pi}{180} = 1.8900242007 \text{ rad}$$

- 2) Hitung Perbedaan Lintang dan Bujur

$$\Delta lat = lat2 - lat1$$

$$\Delta lat = -0.1163158091 - (-0.1161107809)$$

$$\Delta lat = -0.0002050282 \text{ rad}$$

$$\Delta lon = lon2 - lon1$$

$$\Delta lon = 1.8900242007 - 1.8898855085$$

$$\Delta lon = 0.0001386922 \text{ rad}$$

- 3) Hitung Nilai 'a'

$$\alpha = \sin^2 \left(\frac{-0.0002050282}{2} \right) + \cos(-0.1161107809) \times$$

$$\cos(-0.1163158091) \times \sin^2 \left(\frac{0.0001386922}{2} \right)$$

$$\alpha = -0.0001025141^2 + 0.9932667131 \times 0.9932429397 \times 0.0000693461^2$$

$$\alpha = 0.0000000153$$

- 4) Hitung Nilai 'c'

$$c = 2 \times \operatorname{atan}^2(\sqrt{0.0000000153}, \sqrt{1 - 0.0000000153})$$

$$c = 2 \times \operatorname{atan}^2(0.0001235045, 0.9999999924)$$

$$c = 0.0002470091$$

- 5) Hitung Jarak

$$distance = 6371 \times 0.0002470091 = 1.57 \text{ km}$$

Jadi, jarak antara lokasi uji dan lokasi Saepudin adalah sekitar 1.57 kilometer.

Jarak dari lokasi uji ke lokasi Hamid

- 1) Konversi Derajat ke Radian

$$lat1 = -6.6802348 \times \frac{\pi}{180} = -0.1165920921 \text{ rad}$$

$$lon1 = 108.3023643 \times \frac{\pi}{180} = 1.8902328447 \text{ rad}$$

$$lat2 = -6.66440495 \times \frac{\pi}{180} = -0.1163158091 \text{ rad}$$

$$lon2 = 108.29040988 \times \frac{\pi}{180} = 1.8900242007 \text{ rad}$$

- 2) Hitung Perbedaan Lintang dan Bujur

$$\Delta lat = lat2 - lat1$$

$$\Delta lat = -0.1163158091 - (-0.1165920921)$$

$$\Delta lat = 0.0002762830 \text{ rad}$$

$$\Delta lon = lon2 - lon1$$

$$\Delta lon = 1.8900242007 - 1.8902328447$$

$$\Delta lon = -0.0002086440 \text{ rad}$$

- 3) Hitung Nilai 'a'

$$\alpha = \sin^2 \left(\frac{0.0002762830}{2} \right) + \cos(-0.1165920921) \times$$

$$\cos(-0.1163158091) \times \sin^2 \left(\frac{-0.0002086440}{2} \right)$$

$$\alpha = 0.0001381415^2 + 0.9932108381 \times 0.9932429397 \times -0.0001043220^2$$

$$\alpha = 0.0000000298$$

- 4) Hitung Nilai 'c'

$$c = 2 \times \operatorname{atan}^2(\sqrt{0.0000000298}, \sqrt{1 - 0.0000000298})$$

$$c = 2 \times \operatorname{atan}^2(0.0001726824, 0.9999999851)$$

$$c = 0.0003453649$$

- 5) Hitung Jarak

$$distance = 6371 \times 0.0003453649 = 2.20 \text{ km}$$

Jadi, jarak antara lokasi uji dan lokasi Abdullah adalah sekitar 2.20 kilometer.

Jarak dari lokasi uji ke lokasi Ilham

- 1) Konversi Derajat ke Radian

$$lat1 = -6.6529134 \times \frac{\pi}{180} = -0.1161152437 \text{ rad}$$

$$lon1 = 108.3057087 \times \frac{\pi}{180} = 1.8902912155 \text{ rad}$$

$$lat2 = -6.66440495 \times \frac{\pi}{180} = -0.1163158091 \text{ rad}$$

$$lon2 = 108.29040988 \times \frac{\pi}{180} = 1.8900242007 \text{ rad}$$

- 2) Hitung Perbedaan Lintang dan Bujur

$$\Delta lat = lat2 - lat1$$

$$\Delta lat = -0.1163158091 - (-0.1161152437)$$

$$\Delta lat = -0.0002005654 \text{ rad}$$

$$\Delta lon = lon2 - lon1$$

$$\Delta lon = 1.8900242007 - 1.8902912155$$

$$\Delta lon = -0.0002670148 \text{ rad}$$

- 3) Hitung Nilai 'a'

$$\alpha = \sin^2 \left(\frac{-0.0002005654}{2} \right) + \cos(-0.1161152437) \times$$

$$\cos(-0.1163158091) \times \sin^2 \left(\frac{-0.0002670148}{2} \right)$$

$$\alpha = -0.0001002827^2 + 0.9932661960 \times 0.9932429397 \times -0.0001335074^2$$

$$\alpha = 0.0000000276$$

- 4) Hitung Nilai 'c'

$$c = 2 \times \text{atan}^2(\sqrt{0.0000000276}, \sqrt{1 - 0.0000000276})$$

$$c = 2 \times \text{atan}^2(0.0001662564, 0.9999999862)$$

$$c = 0.0003325128$$

- 5) Hitung Jarak

$$distance = 6371 \times 0.0003325128 = 2.12 \text{ km}$$

Jadi, jarak antara lokasi uji dan lokasi Ilham adalah sekitar 2.12 kilometer.

Setelah menganalisis hasil pengujian untuk dua kelompok lokasi (pengguna perempuan dan laki-laki) dapat disimpulkan bahwa hasil perhitungan jarak yang dihasilkan oleh aplikasi serta perhitungan manual menggunakan rumus Haversine menunjukkan kesamaan yang signifikan.

Hasil pengujian menunjukkan bahwa aplikasi yang dikembangkan menggunakan algoritma Haversine dapat menghitung jarak antara lokasi dengan

akurasi yang tinggi. Tidak ditemukan perbedaan yang mencolok antara hasil perhitungan aplikasi dan hasil perhitungan manual. Hal ini menunjukkan bahwa algoritma Haversine yang diterapkan dalam aplikasi bekerja dengan baik dalam mengukur jarak secara konsisten dan akurat.

Selama uji coba, data dari lokasi pengguna perempuan dan laki-laki menunjukkan bahwa jarak yang dihitung oleh aplikasi sesuai dengan perhitungan manual. Ini membuktikan bahwa algoritma Haversine yang digunakan dalam aplikasi efektif dalam melakukan perhitungan jarak dengan ketepatan yang memadai.

Dengan demikian, uji coba ini dapat dinyatakan berhasil. Keberhasilan ini tidak hanya mencerminkan efisiensi algoritma yang digunakan, tetapi juga menegaskan keandalan aplikasi dalam memberikan hasil yang akurat dan konsisten kepada pengguna. Hal ini sangat penting untuk memastikan bahwa fitur filter jarak dalam aplikasi dapat digunakan dengan percaya diri dan memberikan manfaat yang maksimal kepada penggunanya.

4.2.3 Pengujian Filter Jarak

Pengujian filter jarak terdekat adalah langkah penting untuk memastikan fitur ini bekerja dengan baik dalam menampilkan lokasi yang paling dekat dengan titik referensi yang telah ditentukan. Tujuan utama dari pengujian ini adalah untuk memeriksa apakah sistem dapat mengurutkan dan menampilkan lokasi berdasarkan jarak dengan akurat. Keberhasilan fitur ini sangat penting untuk memberikan pengalaman pengguna yang memuaskan dan memenuhi kebutuhan mereka dalam mencari lokasi terdekat.

Pengujian dilakukan dalam dua tahap utama, dengan fokus pada kelompok pengguna yang berbeda, yaitu perempuan dan laki-laki. Pembagian ini bertujuan untuk mengevaluasi apakah ada perbedaan dalam hasil pengurutan dan penampilan lokasi terdekat berdasarkan jenis kelamin pengguna. Dengan memisahkan kelompok ini, diharapkan dapat diperoleh wawasan yang lebih mendalam tentang bagaimana fitur filter jarak bekerja untuk berbagai kelompok pengguna.

Berikut adalah hasil uji filter jarak terdekat pada kelompok pengguna perempuan :

Tabel 4. 5 Pengujian Filter Jarak Perempuan

Data Awal			Data Sesudah Difilter		
No	Nama Titik	Jarak	No	Nama Titik	Jarak
1	Fitri	1,3 km	1	Leni	1,1 km
2	Leni	1,1 km	2	Fitri	1,3 km
3	Annisah	1,3 km	3	Annisah	1,3 km

Tabel di atas menunjukkan perbandingan data sebelum dan sesudah dilakukan proses penyaringan berdasarkan jarak terdekat. Pada data awal, urutan nama yang ditampilkan adalah Fitri dengan jarak 1,3 km, Leni dengan jarak 1,1 km, dan Annisa dengan jarak 1,3 km. Namun, setelah data difilter, urutan berubah berdasarkan jarak, di mana Leni, yang memiliki jarak terdekat 1,1 km, berada di urutan pertama, disusul oleh Fitri dan Annisa yang masing-masing memiliki jarak

1,3 km. Proses penyaringan ini memprioritaskan pengurutan berdasarkan jarak, sehingga individu yang lebih dekat muncul lebih dulu.

Hasil pengujian menunjukkan bahwa filter jarak terdekat berfungsi dengan baik, menampilkan lokasi terdekat secara akurat dan konsisten. Selanjutnya, pengujian dilakukan untuk kelompok pengguna laki-laki dengan prosedur yang sama. Berikut adalah hasil uji:

Tabel 4. 6 Pengujian Filter Jarak Laki-laki

Data Awal			Data Sesudah Difilter		
No	Nama Titik	Jarak	No	Nama Titik	Jarak
1	Saepudin	1,6 km	1	Saepudin	1,6 km
2	Hamid	2,2 km	2	Ilham	2,1 km
3	Ilham	2,1 km	3	Hamid	2,2 km





Tabel di atas menampilkan perbandingan data sebelum dan sesudah difilter berdasarkan jarak. Pada data awal, urutan nama dimulai dengan Saepudin yang berjarak 1,6 km, diikuti oleh Hamid dengan jarak 2,2 km, dan Ilham dengan jarak 2,1 km. Setelah data difilter, urutan diubah berdasarkan jarak terdekat. Saepudin

tetap berada di urutan pertama karena jaraknya yang paling dekat (1,6 km), sementara Ilham berpindah ke urutan kedua dengan jarak 2,1 km, menggantikan Hamid yang turun ke urutan ketiga dengan jarak 2,2 km. Kesimpulannya, filter ini memprioritaskan jarak terdekat sehingga urutan berubah mengikuti jarak dari yang paling dekat hingga yang terjauh.

Tidak ditemukan perbedaan yang signifikan antara hasil perhitungan aplikasi dan perhitungan manual.

Dari kedua pengujian tersebut, dapat disimpulkan bahwa fitur filter jarak terdekat berfungsi dengan baik untuk kedua kelompok pengguna, baik perempuan maupun laki-laki, sehingga fitur ini dapat dinyatakan berhasil.

BAB V

PENUTUP

5.1 Kesimpulan

Aplikasi cari jodoh berbasis Android yang dikembangkan dalam penelitian ini berhasil memenuhi tujuan yang telah ditetapkan. Berdasarkan hasil pengujian, baik melalui pengujian *blackbox* maupun pengujian akurasi, aplikasi menunjukkan performa yang sangat memuaskan. Pengujian *blackbox* memastikan bahwa semua fitur utama aplikasi, termasuk halaman splash, login, pendaftaran, dashboard, pencarian, pesan, profil, map, dan detail pengguna, berfungsi dengan sempurna di perangkat yang diuji. Validasi data dan pengalihan antar halaman berjalan lancar, serta fitur-fitur penting seperti pengiriman pesan dan pencarian pengguna berfungsi tanpa masalah signifikan.

Selain itu, pengujian akurasi terhadap algoritma Haversine menunjukkan hasil yang sangat baik. Algoritma ini mampu mengukur jarak antara berbagai lokasi geografis dengan tingkat akurasi yang tinggi. Hal ini memberikan kepercayaan bagi pengguna dalam menggunakan fitur pencarian berbasis lokasi yang ditawarkan oleh aplikasi.

Secara keseluruhan, aplikasi ini memenuhi ekspektasi fungsionalitas dan akurasi, menunjukkan bahwa aplikasi ini stabil, responsif, dan dapat diandalkan untuk digunakan oleh pengguna.

5.2 Saran

Untuk pengembangan lebih lanjut, terdapat beberapa saran yang dapat dipertimbangkan untuk meningkatkan aplikasi ini:

- 1) **Optimisasi Performa:** Meskipun performa aplikasi sudah memuaskan, optimisasi lebih lanjut dapat dilakukan untuk memastikan aplikasi tetap responsif di berbagai perangkat dengan spesifikasi yang berbeda.
- 2) **Pengujian di Berbagai Perangkat:** Melakukan pengujian pada lebih banyak jenis perangkat dengan berbagai versi Android akan memastikan kompatibilitas dan stabilitas aplikasi di berbagai situasi.

- 3) **Keamanan Data Pengguna:** Memperkuat aspek keamanan data pengguna, seperti enkripsi data dan proteksi privasi, untuk memberikan rasa aman kepada pengguna dalam menggunakan aplikasi.
- 4) **Ketersediaan di Platform Lain:** Saat ini, aplikasi baru dapat dijalankan di sistem operasi Android. Untuk pengembangan ke depan, disarankan agar aplikasi juga dapat diadaptasi untuk sistem operasi lain seperti iOS guna menjangkau lebih banyak pengguna.
- 5) **Distribusi di Play Store:** Saat ini, aplikasi hanya bisa diunduh melalui situs web resmi. Sebaiknya, aplikasi segera didaftarkan dan tersedia di Google Play Store agar lebih mudah diakses oleh pengguna secara luas dan meningkatkan kredibilitas aplikasi di mata publik.

Dengan mengimplementasikan saran-saran tersebut, diharapkan aplikasi cari jodoh ini dapat terus berkembang dan memberikan manfaat yang lebih besar bagi para pengguna dalam mencari pasangan hidup yang sesuai dengan harapan mereka.

DAFTAR PUSTAKA

- Amelia, R., & Ayu, R. F. (2020). BIRO JODOH ONLINE: KEGUNAAN DAN DAMPAK. *Jurnal Ilmiah Syariah*, 163-175. Dipetik Agustus 25, 2024
- Anendya, A. (2022, Agustus 24). *Apa Itu Figma? Penjelasan, Fitur, Keunggulan dan Manfaatnya*. Dipetik Agustus 25, 2024, dari Dewaweb: <https://www.dewaweb.com/blog/apa-itu-figma/>
- Anendya, A. (2024, Januari 14). *Kenalan dengan Codeigniter: Framework Ringan, Cepat, Powerful*. Diambil kembali dari Dewaweb: <https://www.dewaweb.com/blog/apa-itu-codeigniter/>
- Badan Pusat Statistik. (2024). *Statistik Indonesia 2024*. Jakarta: Badan Pusat Statistik. Dipetik Juni 20, 2024
- Coding Studio Team. (2022, Januari 21). *Mengenal Sistem Operasi Android*. Dipetik Juli 14, 2024, dari Coding Studio: <https://codingstudio.id/blog/sistem-operasi-android/>
- Faulina, A. R. (2023, Maret 23). *Apa itu UML? Ini Pengertian, Fungsi, dan Contohnya*. Dipetik Juli 10, 2024, dari Sekawan Media: <https://www.sekawanmedia.co.id/blog/apa-itu-uml/>
- Faulina, A. r. (2024, Februari 9). *Rapid Application Development (RAD): Arti, Tahapan, dan Penerapannya*. Dipetik Agustus 18, 2024, dari Sekawanstudio: <https://sekawanstudio.com/blog/apa-itu-rad/>
- Hermawan, H. (2022, Desember 7). *Comparing the Haversine and Vincenty Algorithms for Calculating Great-Circle Distance*. Dipetik Juli 10, 2024, dari Medium: <https://medium.com/@herihermawan/comparing-the-haversine-and-vincenty-algorithms-for-calculating-great-circle-distance-5a2165857666>

- Huda, N. (2023, Agustus 3). *Perbedaan Front End dan Back End, Calon Developer Wajib Tahu*. Dipetik Agustus 27, 2024, dari Dewaweb: <https://www.dewaweb.com/blog/perbedaan-front-end-dan-back-end/>
- Kusuma, A. D. (2020, September 16). *Apa itu Database? Contoh Produk dan Fungsinya*. Dipetik Juli 10, 2024, dari Dicoding: <https://www.dicoding.com/blog/apa-itu-database/>
- Okpatrioka. (2023). Research And Development (R&D) Penelitian Yang Inovatif Dalam. *Jurnal Pendidikan, Bahasa dan Budaya*, 87. Dipetik juli 14, 2024
- Prakarsha, G. H., Kristianty Lumbantobing, H. D., Ramadhan, M. R., & Prihandi, I. (2021). Haversine Algorithm Design using the Google. *International Journal of Computer Trends and Technology*, 53-60. Dipetik Juli 10, 2024
- Regita, N. (2021, November 19). *Mengapa Anda Harus Belajar Android Studio? Ini Dia 6 Alasannya!* Dipetik 10 Juli, 2024, dari Niagahoster: <https://www.niagahoster.co.id/blog/android-studio-adalah/>
- Setiawan, R. (2021, November 17). *Black Box Testing Untuk Menguji Perangkat Lunak*. Dipetik Juli 14, 2024, dari Dicoding: <https://www.dicoding.com/blog/black-box-testing/>
- Setiawan, R. (2021, Juli 28). *Metode SDLC Dalam Pengembangan Software*. Dipetik Juli 10, 2024, dari Dicoding: <https://www.dicoding.com/blog/metode-sdlc/>
- Vincenty, T. (1975). *Geodetic inverse solution between antipodal points*. Dipetik juli 12, 2024

Lampiran 1

Transkrip Wawancara

Informan : Sekretaris Desa Sukawera

1. Menurut Bapak, apa saja faktor utama yang menyebabkan penurunan angka pernikahan di desa ini?

“Penurunan angka pernikahan di Desa Sukawera disebabkan oleh beberapa faktor. Pertama, banyak warga yang lebih memilih fokus pada pendidikan dan karier sebelum menikah. Selain itu, meningkatnya biaya hidup dan kebutuhan ekonomi membuat banyak orang merasa belum siap secara finansial untuk menikah”.

2. Apakah ada pengaruh dari kondisi ekonomi atau lapangan pekerjaan terhadap penurunan angka pernikahan di desa ini?

“Tentu, kondisi ekonomi dan ketersediaan lapangan pekerjaan sangat mempengaruhi angka pernikahan di sini. Ketidakpastian ekonomi dan rendahnya kesempatan kerja sering kali membuat orang merasa belum stabil secara finansial, jadi mereka cenderung menunda pernikahan sampai merasa lebih mapan”.

3. Apakah ada perubahan dalam norma atau pandangan masyarakat tentang pernikahan yang mungkin mempengaruhi penurunan angka pernikahan?

“Ya, pasti ada perubahan. Dulu, menikah dianggap sebagai tujuan utama dalam hidup, tapi sekarang banyak yang melihatnya sebagai salah satu pilihan hidup yang bisa ditunda. Pandangan seperti ini tentu saja berdampak pada angka pernikahan di desa”.

4. Apakah ada dampak psikologis atau kesejahteraan mental yang dirasakan oleh penduduk yang belum menikah?

“Untuk beberapa orang, belum menikah bisa menimbulkan dampak psikologis seperti rasa kesepian atau tekanan sosial. Ini bisa jadi masalah yang cukup serius bagi mereka”.

5. Menurut Bapak, solusi apa dalam bidang IT yang dapat membantu meningkatkan angka pernikahan di Desa Sukawera?

“Saya pikir, salah satu solusi yang bisa membantu adalah rancang bangun aplikasi cari jodoh yang berbasis lokasi terdekat. Aplikasi seperti ini akan memudahkan warga desa untuk menemukan calon pasangan yang mungkin tinggal di sekitar mereka tapi belum mereka kenal. Dengan fitur yang memungkinkan pengguna melihat calon pasangan berdasarkan jarak, mereka bisa lebih mudah memulai percakapan dan membangun hubungan. Ini tentu bisa mempercepat pengenalan dan mungkin saja meningkatkan angka pernikahan di desa kita.”

Lampiran 2

Transkrip Kuesioner

N o	Nama	JK	usia	1	2	3	4	5	6	7	8	Total	hasil
1	Adi Saputra	L	21	T	8	5	4	5	1	1	2	18	3,0
2	Budiman	L	27	Y	8	5	4	3	1	3	4	20	3,3
3	Dani	L	24	T	8	3	3	3	3	4	3	19	3,2
4	Dedi Juita	L	32	T	11	4	2	2	3	5	4	20	3,3
5	Deni Setiawan	L	26	Y	8	3	3	5	1	5	4	21	3,5
6	Dewi Lia	P	22	T	8	4	2	3	3	5	5	22	3,7
7	Dimas Seto	L	20	T	8	2	4	3	4	3	3	19	3,2
8	Diva	L	22	T	8	1	2	3	1	4	5	16	2,7
9	Ferdiansyah	L	22	T	8	5	4	3	3	3	3	21	3,5
10	Hendrianto	L	22	T	8	5	3	3	3	3	3	20	3,3
11	Jaja Sulaeman	L	21	T	8	3	1	5	1	3	4	17	2,8
12	Ladid halwani	L	24	T	8	2	2	4	1	5	4	18	3,0
13	Luqman	L	24	T	8	2	4	2	4	5	5	22	3,7
14	Marinah	P	28	Y	6	5	2	2	1	5	5	20	3,3
15	Nanda Risye	P	26	T	6	4	2	4	1	5	4	20	3,3
16	Nendri	L	25	T	8	5	4	3	3	3	3	21	3,5
17	Rio Ramadhan	L	27	T	8	2	2	2	1	5	5	17	2,8
18	Rizki Rifaldo	L	21	T	8	4	2	5	1	1	2	15	2,5
19	Robi Andrian	L	27	T	8	5	2	2	4	3	3	19	3,2
20	Saroji	L	25	T	8	5	4	2	4	1	2	18	3,0
21	Wiranto	L	26	T	8	5	4	3	4	3	3	22	3,7
22	Yogi Sugiharto	L	21	T	8	3	3	3	3	3	4	19	3,2
23	Yusril	L	27	Y	8	2	1	5	4	4	3	19	3,2
24	Yusuf Nomi	L	20	T	8	4	3	4	1	3	3	18	3,0

No	Pernyataan
1	Status pernikahan
2	Waktu untuk bekerja
3	Tingkat kesibukan
4	Dampak kesibukan pada sosialisasi
5	Kesibukan sebagai alasan penurunan pernikahan
6	Kesulitan menemukan waktu berkenalan
7	Minat terhadap aplikasi cari jodoh
8	Efektivitas aplikasi cari jodoh

Warna	Nilai
	Sangat
	Setuju
	Cukup
	Netral
	Tidak

Lampiran 3

Dokumentasi Wawancara



Lampiran 4

Surat Izin Penelitian

 **UNIVERSITAS NAHDLATUL ULAMA CIREBON**
Ijin Dikti : 200/E/O/2011, Tanggal 16 September 2011
FAKULTAS ILMU KOMPUTER
Alamat Jl. Sisingamangaraja No. 33 Cirebon Telp. 0231-235900, Fax 0231-235900

Nomor : 011/200.09/PNT/II/2024
Lamp. : -
Perihal : Surat Izin Penelitian

Yth.
Wawan, S.Kom.I.
Kepala Desa Sukawera
di -
Tempat

Assalamualaikum warahmatullahi wabarakatuh

Dekan Fakultas Ilmu Komputer Universitas Nahdlatul Ulama Cirebon, dengan ini mengajukan permohonan kepada Bapak/Ibu untuk dapat menerima mahasiswa kami berikut:

Nama : Ade Trio Purwanto
NIM : 0402201021
Tema Penelitian : Pengembangan Aplikasi Cari Jodoh Menggunakan Algoritma Haversine Berbasis Android

Untuk melaksanakan penelitian di **Desa Sukawera, Kecamatan Ligung, Kabupaten Majalengka** yang Bapak/Ibu pimpin mulai tanggal **02.02. 2024** s/d **17.03. 2024**. Pengajuan pelaksanaan penelitian mahasiswa tersebut di perlukan untuk menyusun laporan Skripsi.

Demikian surat ini kami sampaikan, atas perhatian dan kerja samanya, terima kasih.

Wallahul muwafiq ila Aqamith Thariq
Wassalamualaikum warahmatullahi wabarakatuh.

Cirebon, 21 Februari 2024

Dekan Fakultas Ilmu Komputer
UNU Cirebon

Residul S.Kom., M. Kom.
NIP. 0415028303



Lampiran 5

Surat Keterangan Telah Melakukan Penelitian



PEMERINTAH KABUPATEN MAJALENGKA
KECAMATAN LIGUNG
KANTOR DESA SUKAWERA
Alamat : Jl. Raya Protokol Sukawera No. 279 Ligung – Majalengka - Kode Pos 45456

SURAT KETERANGAN
Nomor : 110 / 044. /Desa/ 2024

Yang bertanda tangan di bawah ini Kepala Desa Sukawera Kecamatan Ligung Kabupaten Majalengka, menerangkan bahwa :

Nama	: ADE TRIO PURWANTO
Tempat Tgl Lahir	: Majalengka 02-08-2003
Jenis Kelamin	: Laki-laki
Pekerjaan	: Pelajar/Mahasiswa
Alamat	: Dusun Dukuh Asih RT 01 RW 005 Desa Sukawera Kecamatan Ligung Kabupaten Majalengka

Dengan ini menerangkan bahwa yang bersangkutan telah melakukan Penelitian dengan Judul PENGEMBANGAN APLIKASI CARİ JODOH MENGGUNAKAN ALGORITMA HAVERSINE FORMULA DI KABUPATEN MAJALENGKA BERBASIS ANDROID terhitung mulai tanggal 02 -02-2024 s.d 17-03-2024 di Dusun Dukuh Asih Desa Sukawera Kecamatan Ligung.

Demikian surat keterangan ini kami buat dengan sebenarnya kepada yang berkepentingan agar menjadi maklum dan dapat dipergunakan sebagaimana mestinya.

Sukawera ,18 Maret 2024
Kepala Desa Sukawera


WAWAN, S.Kom. I