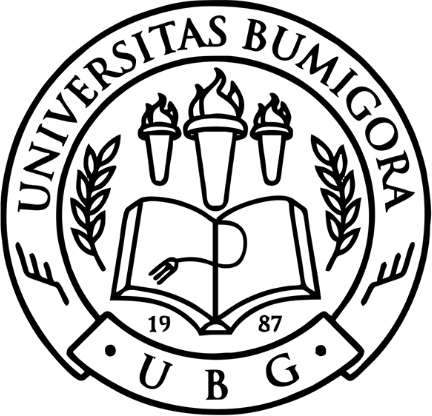
**“Aplikasi Layanan Cerdas Kendaraan Berbasis Mobile dan Tampilan Dashboard”**



**Disusun oleh:**

**Hazrul Hafid Januarta ( 2201010080 )**

**PROGRAM STUDI ILMU KOMPUTER**

**FAKULTAS TEKNIK UNIVERSITAS BUMIGORA**

**2025**

# DAFTAR ISI

[DAFTAR ISI 2](#_Toc203326692)

[DAFTAR GAMBAR 3](#_Toc203326693)

[BAB I PENDAHULUAN 4](#_Toc203326694)

[1.1. Latar belakang 4](#_Toc203326695)

[1.2. Rumusan masalah 4](#_Toc203326696)

[1.3. Tujuan 5](#_Toc203326697)

[1.4. Manfaat 5](#_Toc203326698)

[BAB II LANDASAN TEORI 6](#_Toc203326699)

[2.1. Teori Dasar 6](#_Toc203326701)

[2.1.1. Mobile Computing 6](#_Toc203326702)

[2.1.2. Penjadwalan Dinamis 6](#_Toc203326703)

[2.1.3. Sistem Rekomendasi 6](#_Toc203326704)

[2.1.4. Arsitektur Client-Server 6](#_Toc203326705)

[2.1.5. Sistem Monitoring Kendaraan 6](#_Toc203326706)

[2.1.6. Layanan Darurat Kendaraan 7](#_Toc203326707)

[2.1.7. Integrasi Tampilan Perangkat 7](#_Toc203326708)

[2.1.8. Geolokasi dan Google Maps API 7](#_Toc203326709)

[2.1.9. Firebase Cloud Messaging (FCM) 7](#_Toc203326710)

[2.2. Teknologi Yang Digunakan 7](#_Toc203326711)

[2.2.1. Platform Mobile Development 7](#_Toc203326712)

[2.2.2. Backend dan Server 7](#_Toc203326713)

[2.2.3. Sistem Notifikasi dan Komunikasi 8](#_Toc203326714)

[2.2.4. GPS dan Layanan Peta 8](#_Toc203326715)

[2.2.5. Integrasi Tampilan Kendaraan 8](#_Toc203326716)

[2.2.6. Teknologi Client-Server 8](#_Toc203326717)

[2.3. Penelitian Terkait 8](#_Toc203326718)

[BAB III RANCANGAN SISTEM 9](#_Toc203326719)

[3.1. Konsep dan Mekanisme Kerja Sistem 9](#_Toc203326721)

[3.2. Mekanisme Kerja 9](#_Toc203326722)

[3.3. Alur Interaksi Sistem 10](#_Toc203326723)

[3.4. Arsitektur Sistem 10](#_Toc203326724)

[3.5. Komponen Utama 11](#_Toc203326725)

[3.6. Penjelasan Alur Kerja Infrastruktur 11](#_Toc203326726)

[3.7. Alur Kerja 11](#_Toc203326727)

[3.8. Desain dan Arsitektur Sistem yang Terintegrasi 11](#_Toc203326728)

[3.9. Arsitektur Sistem 12](#_Toc203326729)

[BAB IV RANCANGAN SISTEM 13](#_Toc203326730)

[4.1. Visualisasi ERD (Entity Relationsip Diagram) 13](#_Toc203326732)

[4.1.1. Entities dan Atributnya: 13](#_Toc203326733)

[4.1.2. Relasi: 14](#_Toc203326734)

[4.1.3. Visualisasi ERD 14](#_Toc203326735)

[4.2. Struktur Aplikasi 15](#_Toc203326736)

[4.2.1. Struktur Aplikasi VehicleGuard 15](#_Toc203326737)

[4.3. Visualisasi gambar apk 17](#_Toc203326738)

[DAFTAR PUSTAKA 21](#_Toc203326739)

# DAFTAR GAMBAR

[Gambar 1. Mekanisme Kerja Sistem 10](#_Toc203326672)

[Gambar 2. Visualiasi Arsitektur 12](#_Toc203326673)

[Gambar 3. Visualisasi ERD 15](#_Toc203326674)

[Gambar 4. Login dan registrasi 18](#_Toc203326675)

[Gambar 5. Menyambungkan Kendaraan 18](#_Toc203326676)

[Gambar 6. Dashboard 19](#_Toc203326677)

[Gambar 7. Monitoring 19](#_Toc203326678)

[Gambar 8. Jadual Service 20](#_Toc203326679)

[Gambar 9. Menghubungi Teknisi 20](#_Toc203326680)

# BAB I PENDAHULUAN

## Latar belakang

Kemajuan teknologi di bidang mobile computing dan sistem informasi kendaraan telah membuka peluang besar dalam menciptakan layanan yang lebih cerdas dan adaptif bagi pengguna kendaraan. Salah satu kebutuhan penting di jalan raya adalah layanan darurat yang cepat serta kemampuan untuk memantau kondisi kendaraan secara real-time. Saat ini, banyak pengendara kesulitan dalam mendeteksi kondisi kendaraannya secara dini, tidak mengetahui jadwal servis berkala, atau tidak dapat menghubungi teknisi ketika terjadi masalah secara cepat dan efisien (Rahman et al., 2022).

Dengan menggabungkan teknologi smartphone dan sistem tampilan di kendaraan seperti infotainment system, dapat dibangun sebuah aplikasi cerdas yang memungkinkan pengguna melihat statistik kendaraan, melakukan penjadwalan pemeriksaan berkala, dan menghubungi teknisi darurat saat terjadi kerusakan. Integrasi smartphone dan layar kendaraan ini memanfaatkan pendekatan mobile-edge computing dan komunikasi real-time yang dapat meningkatkan pengalaman berkendara sekaligus mempercepat penanganan insiden di jalan (Kumar & Ghosh, 2020).

Aplikasi semacam ini juga sejalan dengan tren pengembangan smart vehicle systems yang menekankan pada interaktivitas, mobilitas, dan keamanan pengguna (Chen et al., 2021). Pengembangan sistem ini tidak hanya meningkatkan efisiensi pemeliharaan kendaraan, tetapi juga mendukung pengemudi dalam mengantisipasi kerusakan kendaraan, memberikan rasa aman dan nyaman saat berkendara.

## Rumusan masalah

1. Bagaimana sistem dapat melakukan panggilan darurat ke teknisi terdekat secara otomatis?
2. Bagaimana mengatur dan mengingatkan penjadwalan pemeriksaan kendaraan secara berkala?
3. Bagaimana merancang aplikasi mobile yang mampu menampilkan statistik kondisi kendaraan secara real-time?

## Tujuan

1. Menyediakan fitur panggilan teknisi otomatis ketika terjadi kerusakan atau masalah.
2. Menyediakan penjadwalan servis berkala berbasis waktu atau jarak tempuh kendaraan.
3. Merancang dan membangun aplikasi mobile yang dapat menampilkan kondisi kendaraan secara real-time.

## Manfaat

Penelitian ini bermanfaat bagi pengguna kendaraan dalam memantau kondisi kendaraan secara real-time, menjadwalkan servis berkala, serta memperoleh bantuan teknisi dengan cepat saat terjadi masalah, sekaligus memudahkan teknisi dalam menerima notifikasi darurat, mengakses lokasi pengguna, dan mengatur jadwal layanan secara efisien.

# BAB II LANDASAN TEORI



## Teori Dasar

### Mobile Computing

Komputasi bergerak adalah paradigma komputasi yang memungkinkan perangkat portabel seperti smartphone atau tablet untuk melakukan proses komputasi kapan saja dan di mana saja tanpa tergantung pada lokasi fisik (Sari & Widodo, 2021). Teknologi ini mendukung mobilitas pengguna dan fleksibilitas sistem, sehingga sangat sesuai digunakan dalam aplikasi pemantauan dan layanan darurat kendaraan.

### Penjadwalan Dinamis

Penjadwalan dinamis adalah metode penjadwalan yang fleksibel dan dapat berubah sesuai kondisi atau masukan yang terjadi saat sistem berjalan. Dalam konteks aplikasi ini, penjadwalan dinamis digunakan untuk mengatur jadwal pemeriksaan kendaraan berdasarkan waktu penggunaan, jarak tempuh, atau riwayat servis kendaraan sebelumnya.

### Sistem Rekomendasi

Sistem rekomendasi adalah sistem yang memberikan saran kepada pengguna berdasarkan data dan preferensi pengguna sebelumnya. Dalam aplikasi layanan kendaraan, sistem rekomendasi dapat digunakan untuk menyarankan waktu servis, teknisi terdekat, atau lokasi bengkel terdekat berdasarkan riwayat dan lokasi pengguna (Aggarwal, 2016).

### Arsitektur Client-Server

Arsitektur client-server adalah model komunikasi jaringan di mana klien (smartphone pengguna) meminta layanan dari server (pusat data/layanan teknisi). Dalam aplikasi ini, arsitektur client-server digunakan untuk mengelola pengiriman data statistik kendaraan, pemanggilan teknisi, dan sinkronisasi data ke tampilan kendaraan.

### Sistem Monitoring Kendaraan

Sistem monitoring kendaraan adalah sistem berbasis sensor atau input manual yang memantau kondisi kendaraan seperti suhu mesin, jarak tempuh, dan performa umum kendaraan. Informasi ini ditampilkan kepada pengguna secara real-time untuk mendorong perawatan preventif (Putra & Hartono, 2022).

### Layanan Darurat Kendaraan

Layanan darurat kendaraan merujuk pada bantuan teknis yang dapat dipanggil dengan cepat ketika terjadi insiden di jalan. Sistem ini melibatkan deteksi lokasi pengguna, komunikasi real-time dengan teknisi, dan pelacakan proses layanan hingga selesai.

### Integrasi Tampilan Perangkat

Integrasi tampilan antara smartphone dan layar kendaraan memungkinkan informasi penting ditampilkan di **infotainment system** kendaraan melalui teknologi seperti Android Auto atau protokol mirroring lainnya. Ini memberikan kemudahan akses data tanpa mengganggu aktivitas berkendara.

### Geolokasi dan Google Maps API

Geolokasi merupakan proses menentukan lokasi pengguna secara akurat dengan bantuan GPS. Google Maps API digunakan untuk menampilkan posisi pengguna, teknisi, dan bengkel terdekat, serta menyediakan rute tercepat bagi teknisi menuju lokasi.

### Firebase Cloud Messaging (FCM)

FCM adalah layanan dari Google yang memungkinkan pengiriman pesan atau notifikasi secara real-time ke perangkat pengguna. Dalam aplikasi ini, FCM digunakan untuk mengirim pemberitahuan seperti status teknisi, jadwal servis, dan konfirmasi permintaan darurat.

## Teknologi Yang Digunakan

### Platform Mobile Development

Aplikasi dikembangkan menggunakan **Flutter**, framework open-source dari Google yang memungkinkan pengembangan antarmuka pengguna yang responsif dan lintas platform (Android dan iOS). Flutter menggunakan bahasa pemrograman Dart, dan mendukung komponen UI modern serta integrasi dengan API eksternal.

### Backend dan Server

Untuk backend, digunakan **Firebase Realtime Database** sebagai database cloud NoSQL yang memungkinkan pembaruan data secara instan dan sinkronisasi lintas perangkat. Firebase juga mendukung autentikasi pengguna dan notifikasi melalui Firebase Cloud Messaging (FCM).

### Sistem Notifikasi dan Komunikasi

Firebase Cloud Messaging (FCM) digunakan untuk mengirimkan notifikasi secara real-time kepada pengguna maupun teknisi, seperti notifikasi permintaan darurat, pembaruan status teknisi, atau pengingat servis.

### GPS dan Layanan Peta

Untuk pelacakan lokasi dan navigasi teknisi, digunakan **Google Maps API** yang memungkinkan pemetaan lokasi pengguna dan teknisi serta menampilkan rute tercepat menuju lokasi. Layanan ini juga mendukung fitur pencarian lokasi bengkel terdekat.

### Integrasi Tampilan Kendaraan

Integrasi antara aplikasi dan tampilan layar kendaraan dapat dilakukan melalui Android Auto SDK, yang memungkinkan tampilan informasi penting dari aplikasi ditampilkan di layar infotainment kendaraan tanpa mengganggu konsentrasi pengemudi.

### Teknologi Client-Server

Model arsitektur client-server digunakan untuk komunikasi antara aplikasi mobile (client) dengan server Firebase atau API backend lainnya. Client mengirimkan permintaan (request), dan server memberikan respon berupa data atau notifikasi.

## Penelitian Terkait

Penelitian ini merupakan pengembangan dari karya Rahman et al. (2022) yang hanya menyediakan fitur panggilan darurat dan Putra & Hartono (2022) yang fokus pada monitoring kendaraan, dengan menggabungkan fitur pemantauan kondisi kendaraan, penjadwalan servis dinamis, layanan darurat real-time, integrasi tampilan ke layar kendaraan, serta sistem notifikasi teknisi berbasis lokasi dan Firebase dalam satu aplikasi mobile yang saling terhubung.

# BAB III RANCANGAN SISTEM

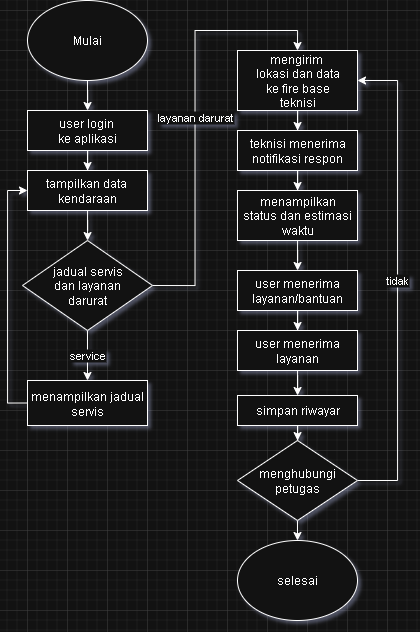


## Konsep dan Mekanisme Kerja Sistem

Sistem yang dirancang merupakan sebuah aplikasi mobile yang memungkinkan pengguna untuk memantau kondisi kendaraan secara real-time, menjadwalkan servis berkala, dan memanggil teknisi secara darurat apabila terjadi masalah di jalan. Aplikasi ini dikembangkan dengan pendekatan mobile computing dan terhubung ke cloud backend menggunakan layanan Firebase. Selain diakses melalui smartphone, informasi penting dari aplikasi juga ditampilkan di layar kendaraan melalui integrasi dengan sistem Android Auto. Mekanisme kerja sistem memanfaatkan GPS, database real-time, serta layanan cloud messaging agar seluruh proses berlangsung cepat, akurat, dan responsif terhadap situasi darurat.

## Mekanisme Kerja

Aplikasi bekerja dengan membagi fungsionalitasnya menjadi tiga bagian utama. Pertama, fitur monitoring kendaraan yang menampilkan data penting seperti suhu mesin dan jarak tempuh kendaraan secara langsung melalui aplikasi. Kedua, sistem penjadwalan servis yang akan memberikan notifikasi atau rekomendasi waktu servis berikutnya berdasarkan data pemakaian kendaraan. Ketiga, layanan darurat yang memungkinkan pengguna mengirim permintaan bantuan ke teknisi terdekat lengkap dengan lokasi saat ini secara otomatis. Semua proses ini berlangsung secara terintegrasi dengan backend Firebase yang menyimpan dan memproses data pengguna dan teknisi.



Gambar 1. Mekanisme Kerja Sistem

## Alur Interaksi Sistem

Pengguna akan terlebih dahulu melakukan proses login dan mendaftarkan informasi kendaraan. Setelah berhasil masuk, sistem akan langsung menampilkan data statistik kendaraan yang diperbarui secara berkala. Ketika kendaraan mendekati batas servis, sistem akan memberikan pengingat yang dapat langsung dijadwalkan oleh pengguna. Dalam keadaan darurat, pengguna hanya perlu menekan tombol "Panggil Teknisi", maka sistem akan secara otomatis mengirimkan informasi lokasi dan identitas pengguna ke teknisi yang paling dekat. Teknisi yang menerima notifikasi dapat merespon, dan status bantuan akan diperbarui secara real-time di aplikasi pengguna.

## Arsitektur Sistem

Sistem ini dirancang dengan menggunakan arsitektur client-server, di mana aplikasi mobile bertindak sebagai client yang berkomunikasi dengan server Firebase sebagai backend. Aplikasi client dijalankan di perangkat pengguna berbasis Android dan terhubung dengan sistem Android Auto di kendaraan. Server Firebase bertugas menangani otentikasi pengguna, menyimpan data kendaraan, mengelola jadwal servis, serta memfasilitasi pengiriman notifikasi kepada teknisi. Komunikasi antara client dan server menggunakan protokol HTTPS dengan dukungan Firebase Cloud Messaging dan Realtime Database.

## Komponen Utama

Komponen utama dari sistem ini terdiri dari aplikasi mobile yang menjadi antarmuka pengguna untuk melakukan berbagai fungsi seperti pemantauan kendaraan, jadwal servis, dan permintaan bantuan. Komponen server menggunakan Firebase yang berfungsi untuk menyimpan dan mengelola seluruh data pengguna, kendaraan, dan teknisi. Selain itu, digunakan juga layanan Firebase Cloud Messaging untuk mengirimkan notifikasi secara real-time. Sistem juga memanfaatkan teknologi GPS dan Google Maps API untuk melacak lokasi pengguna dan teknisi, serta Android Auto SDK untuk menampilkan data pada layar kendaraan.

## Penjelasan Alur Kerja Infrastruktur

Alur kerja infrastruktur sistem dimulai dari proses pengumpulan data kendaraan melalui aplikasi mobile. Data yang dikirim akan disimpan di Firebase dan disinkronkan ke berbagai perangkat yang terhubung. Informasi ini juga dapat ditampilkan secara langsung di layar kendaraan. Ketika pengguna membutuhkan bantuan, aplikasi akan mengirimkan permintaan darurat yang berisi lokasi dan deskripsi singkat ke teknisi melalui notifikasi yang dikirimkan oleh Firebase. Teknisi kemudian dapat menerima permintaan tersebut dan memulai perjalanan menuju lokasi pengguna. Seluruh aktivitas dicatat dan ditampilkan kembali dalam aplikasi untuk transparansi layanan.

## Alur Kerja

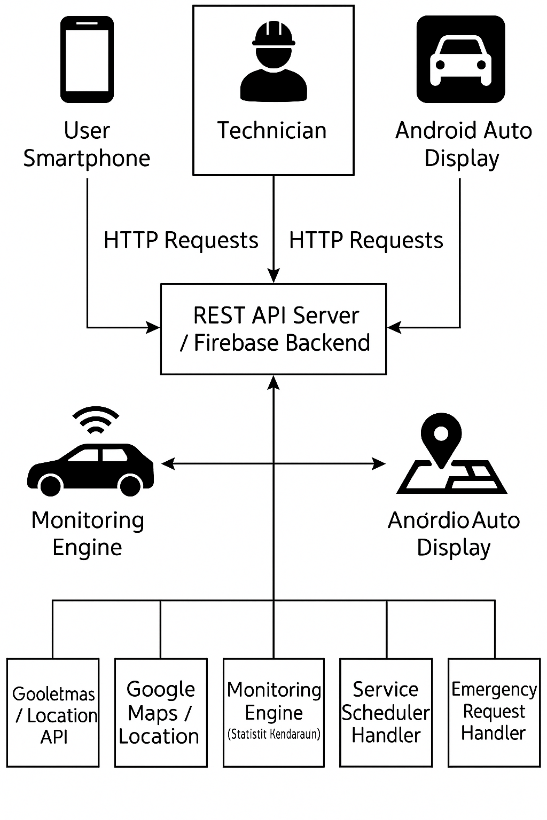
Alur kerja utama dalam sistem ini meliputi dua skenario. Pertama, untuk kasus panggilan darurat, pengguna menekan tombol bantuan darurat di aplikasi, sistem kemudian mengirimkan informasi lokasi dan identitas pengguna ke server, dan server akan mengirimkan permintaan ke teknisi terdekat. Setelah teknisi merespons, status bantuan akan diperbarui secara otomatis dan ditampilkan kepada pengguna. Kedua, dalam alur monitoring dan penjadwalan, sistem mencatat data penggunaan kendaraan dan secara otomatis memberikan notifikasi ketika kendaraan mendekati batas servis. Pengguna dapat mengonfirmasi atau mengubah jadwal servis sesuai kebutuhan, dan sistem akan menyimpan riwayatnya.

## Desain dan Arsitektur Sistem yang Terintegrasi

Desain sistem dirancang agar terintegrasi penuh antara smartphone dan layar kendaraan. Aplikasi mobile menjadi pusat interaksi pengguna, sedangkan layar kendaraan menampilkan informasi penting tanpa perlu mengalihkan perhatian pengguna saat berkendara. Desain antarmuka aplikasi dibuat sederhana dan intuitif untuk memudahkan navigasi, serta menggunakan pendekatan mobile-first agar tampil responsif di berbagai ukuran layar. Semua fitur dan proses sistem terhubung ke server Firebase sehingga seluruh data dan interaksi dapat berjalan secara sinkron dan real-time.

## Arsitektur Sistem

Arsitektur sistem secara keseluruhan menunjukkan bahwa aplikasi pengguna dan aplikasi teknisi bekerja saling terhubung melalui server Firebase sebagai pusat data dan pengelolaan notifikasi. Aplikasi pengguna berfungsi mengelola data kendaraan, permintaan bantuan, dan riwayat servis, sementara aplikasi teknisi berfungsi menerima dan merespons permintaan layanan. Firebase menangani semua komunikasi data secara real-time, dan Android Auto digunakan untuk menampilkan data aplikasi di layar kendaraan, menciptakan sistem yang terintegrasi antara perangkat mobile, cloud, dan kendaraan.



Gambar 2. Visualiasi Arsitektur

# BAB IV RANCANGAN SISTEM

## Gambaran Umum Aplikasi

Aplikasi **VehicleGuard** merupakan aplikasi mobile berbasis Android yang dirancang untuk membantu pengguna kendaraan dalam memantau kondisi kendaraan secara real-time, melakukan penjadwalan servis berkala, serta meminta bantuan darurat dari teknisi terdekat saat mengalami kendala di jalan. Aplikasi ini mendukung integrasi dengan layar kendaraan (seperti Android Auto) serta memanfaatkan konektivitas internet, GPS, dan Firebase sebagai infrastruktur backend-nya.

## Fitur Utama Aplikasi

### Fitur Utama Aplikasi

Fitur autentikasi pada aplikasi *VehicleGuard* merupakan tahap awal interaksi pengguna, yang memungkinkan pengguna atau teknisi melakukan proses login ataupun pendaftaran akun baru. Sistem ini menggunakan Firebase Authentication yang mendukung dua metode login: melalui email dan password, serta melalui integrasi akun Google untuk akses cepat.

Saat pengguna baru mendaftar, data seperti nama, alamat email, nomor telepon, dan password dienkripsi dan disimpan dengan aman. Validasi dilakukan secara real-time untuk memastikan email tidak duplikat dan password memenuhi ketentuan. Jika pengguna lupa kata sandi, tersedia juga fitur reset password yang akan mengirimkan tautan pemulihan melalui email terdaftar.

Fitur ini penting untuk menjaga privasi data kendaraan dan mempersonalisasi layanan berdasarkan identitas masing-masing pengguna.

### Pairing atau Sambungan Kendaraan

Setelah berhasil login, pengguna diminta untuk menghubungkan aplikasi dengan kendaraannya. Proses ini dapat dilakukan melalui beberapa metode, yaitu:

* **Pemindaian QR Code** yang tertera pada perangkat onboard kendaraan.
* **Input kode kendaraan** secara manual (jika tidak memiliki QR).
* **Pairing otomatis via Bluetooth**, apabila perangkat onboard mendukung konektivitas nirkabel.

Setelah proses pairing berhasil, kendaraan tersebut akan terdaftar dalam akun pengguna dan menjadi sumber data untuk semua fitur monitoring dan layanan lainnya. Jika pengguna memiliki lebih dari satu kendaraan, sistem memungkinkan pengelolaan multi-kendaraan dengan memilih kendaraan aktif.

### Monitoring Kendaraan Secara Real-Time

Fitur ini menjadi salah satu fitur inti dalam aplikasi, yang memungkinkan pengguna memantau kondisi kendaraan secara langsung. Data yang ditampilkan antara lain:

* **Suhu mesin** (dalam derajat Celsius),
* **Jarak tempuh harian dan total**,
* **Tekanan ban** (jika perangkat mendukung),
* **Konsumsi bahan bakar**,
* **Status sistem kendaraan** secara umum.

Data-data ini diperbarui secara berkala melalui sinkronisasi dengan sensor kendaraan (melalui Bluetooth atau API lokal), kemudian ditampilkan dalam bentuk visual seperti grafik garis (line chart), ikon status (warna hijau, kuning, atau merah), serta label informasi.

Pengguna akan mendapatkan notifikasi otomatis jika nilai suhu mesin atau tekanan ban melebihi batas aman, sehingga dapat segera mengambil tindakan pencegahan.

### Penjadwalan Servis Berkala

Fitur ini berfungsi untuk membantu pengguna dalam mengatur dan mengingat jadwal servis kendaraan secara rutin. Aplikasi menyediakan:

* **Rekomendasi waktu servis berikutnya**, baik berdasarkan jarak tempuh maupun tanggal.
* **Kalender servis**, untuk memilih dan menetapkan waktu layanan.
* **Riwayat servis sebelumnya**, sebagai referensi.

Pengguna dapat mengatur pengingat servis seperti ganti oli, pemeriksaan rem, atau tune-up, sesuai dengan jenis kendaraan dan kebutuhannya. Jika batas waktu atau kilometer servis telah tercapai, sistem akan memberikan notifikasi yang bersifat mendesak. Hal ini penting untuk menjaga performa kendaraan serta menghindari kerusakan yang lebih serius.

### Layanan Darurat (Emergency SOS)

Dalam kondisi darurat seperti kendaraan mogok, ban bocor, atau mesin overheating di tengah jalan, pengguna dapat menggunakan fitur **Layanan Darurat** yang tersedia di halaman utama. Fitur ini menyediakan:

* **Tombol “Minta Bantuan Sekarang”** yang mengaktifkan sistem pencarian teknisi terdekat.
* **Deteksi lokasi otomatis** berdasarkan GPS pengguna.
* **Estimasi waktu teknisi tiba** dan identitas teknisi.

Setelah permintaan dikirim, sistem backend akan mencarikan teknisi aktif dengan lokasi terdekat yang tersedia. Jika ditemukan, teknisi akan menerima notifikasi dan mulai menuju ke lokasi pengguna. Pengguna dapat memantau progres teknisi secara langsung di aplikasi.

Fitur ini memberikan rasa aman bagi pengguna saat berkendara, khususnya di daerah yang jauh dari bengkel atau pada malam hari.

### Fitur Chat Konsultasi dengan Teknisi

Salah satu fitur yang membedakan *VehicleGuard* dari aplikasi layanan kendaraan lain adalah **adanya ruang chat langsung** antara pengguna dan teknisi. Setelah permintaan darurat dikirim dan teknisi ditugaskan, sistem akan membuka ruang obrolan di mana:

* Pengguna dapat menjelaskan masalah secara lebih rinci.
* Teknisi dapat memberikan saran awal, seperti pengecekan manual.
* Dapat mengirim teks atau gambar (jika fitur diaktifkan).

Chat ini bersifat **sementara**, hanya aktif selama status bantuan berlangsung, dan akan ditutup otomatis ketika layanan selesai. Riwayat percakapan disimpan untuk dokumentasi dan peningkatan layanan di masa depan.

### Notifikasi dan Peringatan Otomatis

Untuk mendukung kenyamanan dan keamanan pengguna, aplikasi menyediakan sistem notifikasi otomatis berbasis Firebase Cloud Messaging (FCM). Notifikasi yang diberikan meliputi:

* **Peringatan servis mendekati batas waktu**.
* **Peringatan kondisi kendaraan berbahaya** (misal: suhu tinggi).
* **Status bantuan teknisi**: menuju lokasi, tiba, selesai.

Notifikasi ini muncul baik dalam bentuk *push notification* di luar aplikasi, maupun dalam daftar notifikasi yang dapat diakses di dalam aplikasi.

### Riwayat Kendaraan dan Layanan

Semua aktivitas pengguna, seperti:

* Riwayat servis kendaraan,
* Riwayat monitoring statistik,
* Riwayat layanan darurat,
* Riwayat interaksi dengan teknisi,

akan disimpan dan ditampilkan dalam menu **Riwayat & Laporan**. Data ini bisa difilter berdasarkan tanggal, jenis layanan, atau status.

Fitur ini memudahkan pengguna melakukan audit kendaraan pribadi dan juga membantu teknisi mengetahui jejak kondisi kendaraan sebelum melakukan tindakan.

## Visualisasi ERD (Entity Relationsip Diagram)

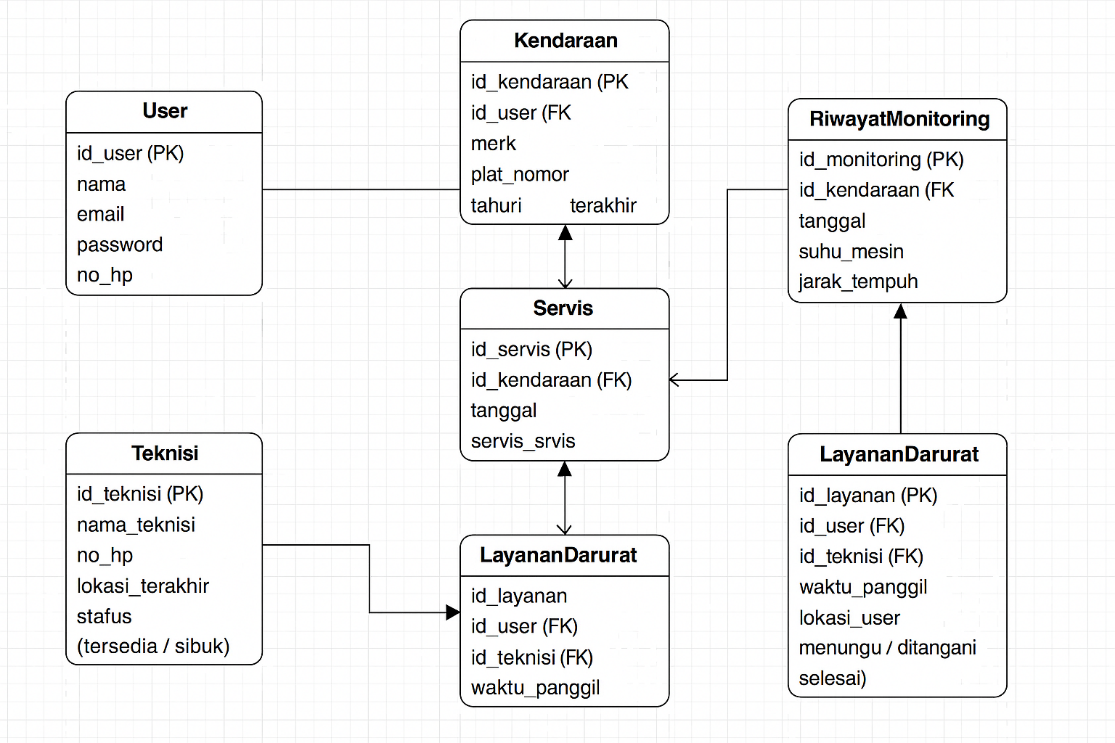
### Entities dan Atributnya:

1. User
   * id\_user (PK)
   * nama
   * email
   * password
   * no\_hp
2. Kendaraan
   * id\_kendaraan (PK)
   * id\_user (FK)
   * merk
   * model
   * plat\_nomor
   * tahun
   * kilometer\_terakhir
3. Servis
   * id\_servis (PK)
   * id\_kendaraan (FK)
   * tanggal\_servis
   * jenis\_servis
   * status\_servis (jadwal / selesai)
4. RiwayatMonitoring
   * id\_monitoring (PK)
   * id\_kendaraan (FK)
   * tanggal
   * suhu\_mesin
   * jarak\_tempuh
   * status\_kendaraan
5. Teknisi
   * id\_teknisi (PK)
   * nama\_teknisi
   * no\_hp
   * lokasi\_terakhir
   * status (tersedia / sibuk)
6. LayananDarurat
   * id\_layanan (PK)
   * id\_user (FK)
   * id\_teknisi (FK)
   * waktu\_panggil
   * lokasi\_user
   * status (menunggu / ditangani / selesai)

### Relasi:

* User memiliki banyak Kendaraan
* Kendaraan memiliki banyak Servis
* Kendaraan memiliki banyak RiwayatMonitoring
* User bisa melakukan banyak LayananDarurat
* Teknisi menangani banyak LayananDarurat

### Visualisasi ERD



Gambar 3. Visualisasi ERD

## Struktur Aplikasi

### Struktur Aplikasi VehicleGuard

1. **Halaman Sign In / Sign Up**

**Fungsi:**

* Untuk registrasi akun pengguna dan teknisi.
* Untuk login menggunakan akun yang sudah terdaftar.

**Fitur & Komponen:**

* Tombol: Sign In, Sign Up, Continue with Google.
* Form input: Email, Password.
* Validasi input otomatis.
* Navigasi ke Reset Password.

1. **Halaman Beranda / Dashboard**

**Fungsi:**

* Menampilkan ringkasan statistik kendaraan secara real-time.
* Menyediakan akses cepat ke fitur penting: monitoring, jadwal servis, dan layanan darurat.

**Komponen:**

* Ringkasan suhu mesin, jarak tempuh, dan status kendaraan.
* Tombol akses cepat: "Monitoring", "Jadwal Servis", "Layanan Darurat".
* Informasi kendaraan terdaftar.
* Navigasi ke pengaturan dan riwayat.

1. **Halaman Monitoring Kendaraan**

**Fungsi:**

* Menampilkan data statistik kendaraan yang diambil secara berkala.
* Memberikan peringatan jika ada nilai sensor di luar batas normal.

**Komponen:**

* Grafik suhu mesin & jarak tempuh harian/mingguan.
* Indikator kondisi kendaraan (normal/warning).
* Notifikasi servis otomatis jika parameter kritis terdeteksi.

1. **Halaman Jadwal Servis**

**Fungsi:**

* Mengatur, melihat, dan mengubah jadwal servis berkala kendaraan.

**Komponen:**

* Kalender servis.
* Form input kilometer/jadwal servis berikutnya.
* Tombol konfirmasi & edit jadwal.
* Riwayat servis sebelumnya.

1. **Halaman Layanan Darurat**

**Fungsi:**

* Mengirim permintaan bantuan teknisi darurat saat kendaraan mengalami masalah.

**Komponen:**

* Tombol SOS.
* Deteksi otomatis lokasi pengguna.
* Estimasi waktu kedatangan teknisi.
* Status penanganan (menunggu – ditangani – selesai).
* Navigasi ke riwayat bantuan.

1. **Halaman Notifikasi**

**Fungsi:**

* Memberikan update real-time terkait status teknisi, jadwal servis, dan peringatan kendaraan.

**Komponen:**

* Tabel histori dengan filter (tanggal, jenis layanan).
* Detail teknisi yang menangani.
* Catatan Daftar notifikasi dengan ikon prioritas.
* Notifikasi push dari Firebase (penanganan darurat, pengingat servis).

1. **Halaman Riwayat & Laporan**

**Fungsi:**

* Menampilkan riwayat servis kendaraan dan bantuan darurat sebelumnya.

**Komponen:**

* hasil pemeriksaan.

1. **Halaman Profil & Pengaturan**

**Fungsi:**

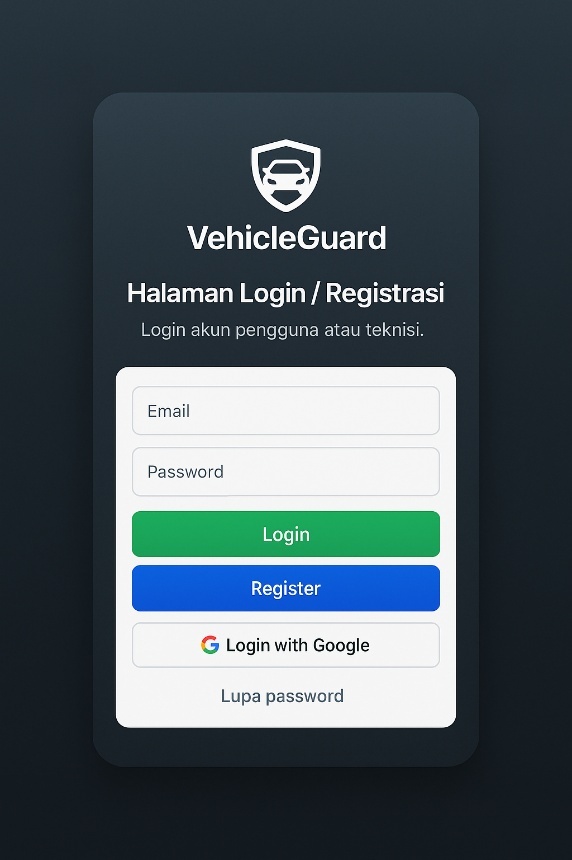
* Menampilkan dan mengatur profil pengguna dan kendaraan.

**Komponen:**

* Data diri pengguna.
* Data kendaraan (merk, model, plat).
* Tombol edit profil, logout, dan pengaturan integrasi Android Auto.

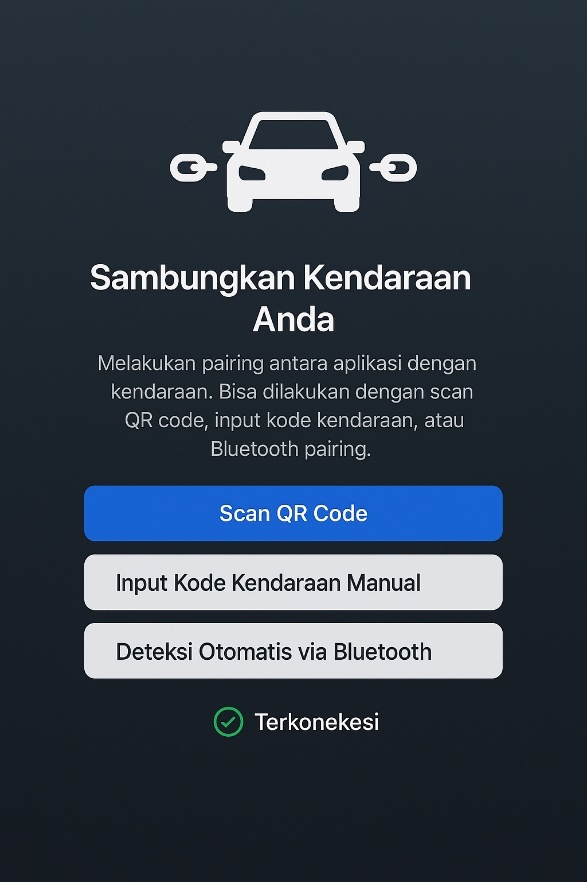
## Visualisasi gambar apk

1. Login dan registrasi



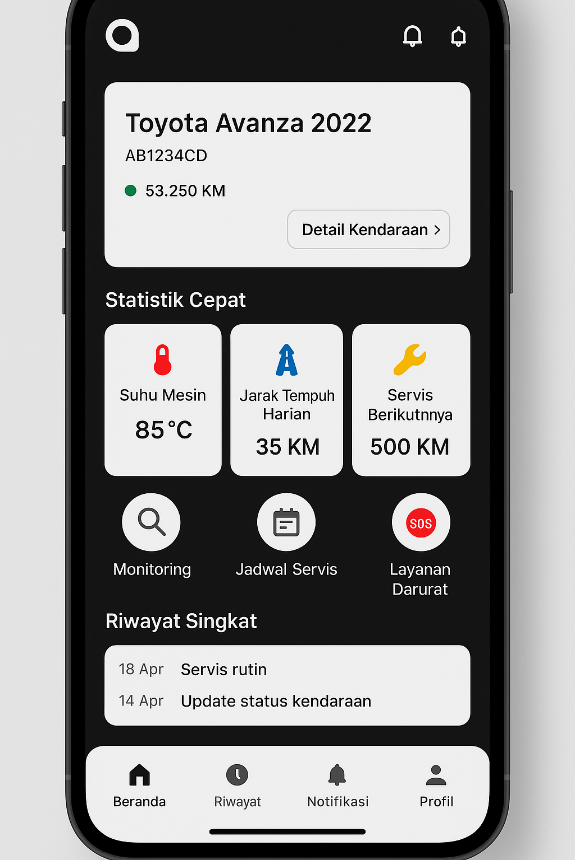
Gambar 4. Login dan registrasi

1. Configurasi Dengan Kendaraan



Gambar 5. Menyambungkan Kendaraan

1. Dashboard



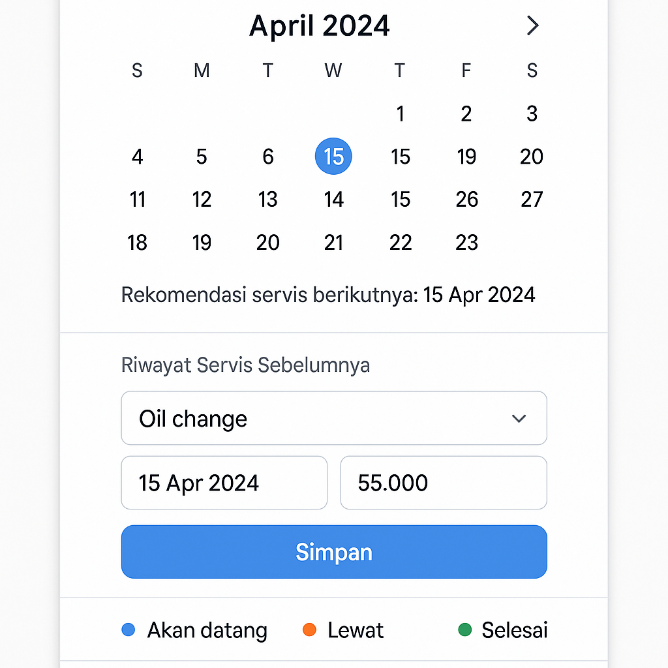
Gambar 6. Dashboard

1. Monitoring Kendaraan



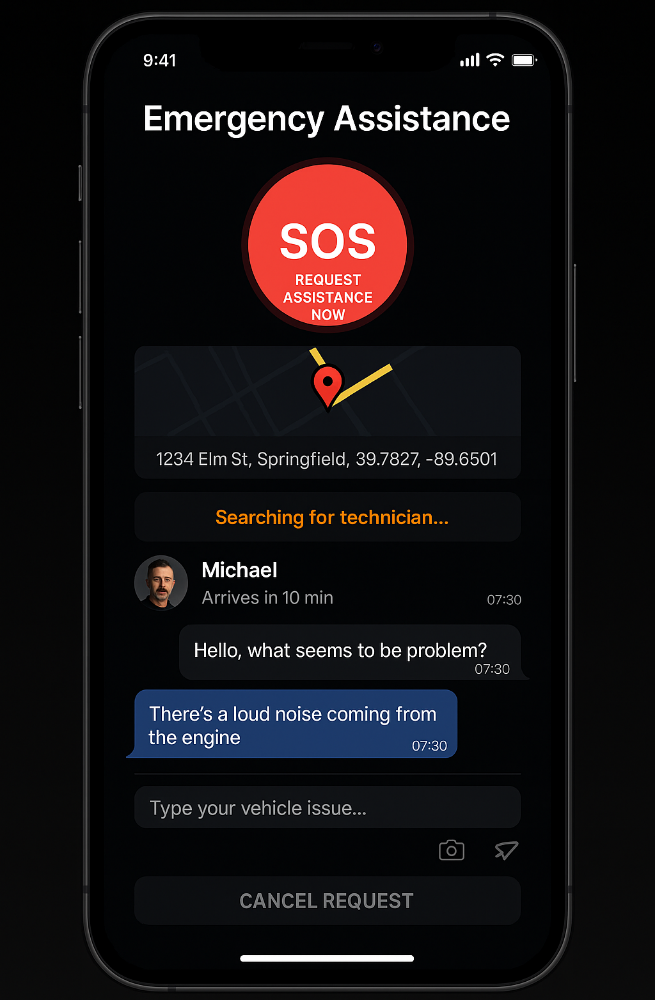
Gambar 7. Monitoring

1. Jdual Service



Gambar 8. Jadual Service

1. Layanan Dan SOS



Gambar 9. Menghubungi Teknisi

# DAFTAR PUSTAKA

Chen, Y., Zhang, Q., & Lin, X. (2021). Smart vehicle systems: Design, integration, and future trends. ACM Computing Surveys, 54(3), 1–32. https://doi.org/10.1145/3439848

Kumar, R., & Ghosh, D. (2020). Mobile edge computing for vehicle-to-everything (V2X) communication: Design and challenges. IEEE Communications Surveys & Tutorials, 22(1), 88–105. https://doi.org/10.1109/COMST.2019.2949987

Nugroho, M. D., & Santoso, H. (2020). Pengembangan aplikasi pemantauan kendaraan berbasis Android dengan notifikasi berkala. Jurnal Informatika dan Rekayasa Perangkat Lunak, 8(2), 45–52.

Rahman, A., Nugroho, Y., & Prasetyo, D. (2022). Perancangan aplikasi panggilan darurat berbasis mobile untuk kendaraan bermotor. Jurnal Teknologi Informasi dan Komunikasi, 14(1), 23–30.

Aggarwal, C. C. (2016). *Recommender systems: The textbook*. Springer. https://doi.org/10.1007/978-3-319-29659-3

Chen, Y., Zhang, Q., & Lin, X. (2021). Smart vehicle systems: Design, integration, and future trends. *ACM Computing Surveys (CSUR)*, *54*(3), 1–32. https://doi.org/10.1145/3439848

Kumar, R., & Ghosh, D. (2020). Mobile edge computing for vehicle-to-everything (V2X) communication: Design and challenges. *IEEE Communications Surveys & Tutorials*, *22*(1), 88–105. https://doi.org/10.1109/COMST.2019.2949987

Putra, R. D., & Hartono, A. (2022). Sistem monitoring kendaraan berbasis Android untuk deteksi dini kerusakan. *Jurnal Teknik Informatika dan Sistem Informasi*, *9*(1), 44–50.

Rahman, A., Nugroho, Y., & Prasetyo, D. (2022). Perancangan aplikasi panggilan darurat berbasis mobile untuk kendaraan bermotor. *Jurnal Teknologi Informasi dan Komunikasi*, *14*(1), 23–30.

Sari, N. P., & Widodo, A. (2021). Implementasi mobile computing dalam sistem informasi berbasis lokasi. *Jurnal Informatika*, *15*(2), 115–121.