

BAB 3

ANALISIS DAN PERANCANGAN

Analisis dan perancangan membutuhkan tahapan yang sistematis untuk mendapatkan sistem yang baik dan tepat sasaran. Tahapan dimulai dari menganalisis sistem yang meliputi analisis masalah, analisis sistem yang sedang berjalan, prosedur yang sedang berjalan, sistem yang akan dibangun, metode yang digunakan, alat bantu yang sudah ada, analisis kebutuhan pengguna, kebutuhan non-fungsional, kebutuhan fungsional, kebutuhan perangkat keras dan kebutuhan perangkat lunak. Sedangkan untuk tahapan perancangan sistem membahas tentang arsitektur sistem dan perancangan antar muka yang akan dibangun.

3.1 Analisis Sistem

Analisis sistem merupakan cara penguraian dari suatu sistem yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk melakukan identifikasi dan evaluasi permasalahan, kesempatan, hambatan yang terjadi dan kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan untuk tercapainya sistem yang lebih baik [27].

3.1.1 Analisis Masalah

SKK Migas menyediakan fasilitas dan layanan berupa kendaraan dinas operasional untuk kegiatan kedinasan diluar kantor. Kendaraan yang disediakan saat ini berjumlah 18 (delapan belas). Biaya operasional penggunaan kendaraan seperti bahan bakar minyak, tarif tol dan tarif parkir ditanggung sepenuhnya oleh SKK Migas. Fasilitas kendaraan dinas perlu dimanfaatkan sebaik mungkin terutama untuk tercapainya layanan yang maksimal dan biaya operasional yang efisien.

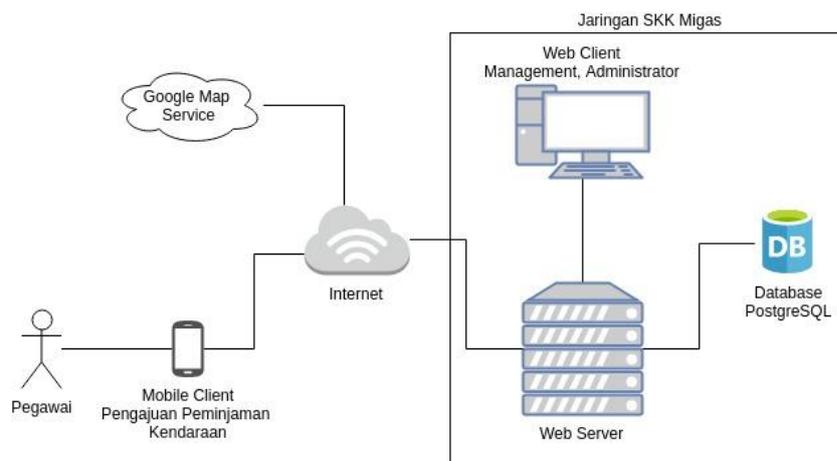
Berikut masalah dan kebutuhan dalam pengelolaan fasilitas kendaraan dinas operasional:

1. Data penggunaan kendaraan dinas yang dapat dikumpulkan saat ini hanya berupa riwayat peminjaman kendaraan yaitu profil penumpang, waktu keberangkatan, lokasi keberangkatan, tujuan dan profil sopir.
2. Adanya kebutuhan untuk menganalisa data kendaraan selama beroperasi atau selama sedang dalam perjalanan yaitu riwayat perjalanan dan kesalahan mengemudi.
3. Informasi yang dapat dihasilkan dari data penggunaan kendaraan dinas saat ini masih berupa laporan *excel* yang dirasa kurang lengkap, kurang informatif dan memerlukan pengolahan lebih lanjut.

Maka dari itu, diperlukan suatu sistem yang dapat mengumpulkan, mengolah dan menyajikan data secara visual yang mampu menghasilkan informasi penggunaan kendaraan dinas yang lebih lengkap yang dapat digunakan sebagai bahan evaluasi dan pengambilan keputusan.

3.1.2 Analisis Sistem Yang Sedang Berjalan

Sistem yang sedang berjalan saat ini dinamakan *Vehicle Management System* atau disingkat VMS. Sistem ini memungkinkan setiap pengajuan peminjaman kendaraan tercatat dan terkelola. Setiap pengajuan peminjaman kendaraan oleh pegawai, persetujuan oleh atasan dan pemberitahuan penugasan kepada sopir dilakukan menggunakan aplikasi pada *smartphone* Android.



Gambar 3.1 Arsitektur Sistem Yang Sedang Berjalan

3.1.2.1 Web Client

Web Client yaitu berupa aplikasi berbasis web yang digunakan oleh petugas Admin untuk pengelolaan peminjaman kendaraan. Pada *web client* ini petugas Admin memantau setiap peminjaman kendaraan yang dilakukan oleh pegawai.

Daftar Pengajuan										Semua	Disetujui	Teralokasi	Selesai
No.	Tgl. Pengajuan	Nama Pengaju	Jenis Perjalanan	Inap	Tgl. Pinjam	Jam	Status	Alokasikan		Detail			
1	31/05/2018	Akhmad Haikal	pulang-pergi	tidak	31/05/2018	02:00	teralokasi	Alokasi	Menunggu	Detail	Batalan		
2	31/05/2018	Shinta Swasti Santasayacitta	pulang-pergi	tidak	31/05/2018	11:30	sudah-selesai	Alokasi	Menunggu	Detail	Batalan		
3	31/05/2018	M. Nurdin Zulkarnain	pulang-pergi	tidak	31/05/2018	12:30	teralokasi	Alokasi	Menunggu	Detail	Batalan		
4	31/05/2018	Yanin Kholison	pulang-pergi	tidak	31/05/2018	11:30	teralokasi	Alokasi	Menunggu	Detail	Batalan		
5	31/05/2018	Yanin Kholison	sekali-jalan	tidak	31/05/2018	10:15	dibatalkan	Alokasi	Menunggu	Detail	Batalan		
6	31/05/2018	Jeffry Nugraha	pulang-pergi	tidak	31/05/2018	13:30	sudah-selesai	Alokasi	Menunggu	Detail	Batalan		
7	31/05/2018	Quartiana Zwesty	pulang-pergi	tidak	31/05/2018	08:20	sudah-selesai	Alokasi	Menunggu	Detail	Batalan		
8	31/05/2018	Bintoro	sekali-jalan	tidak	31/05/2018	10:30	sudah-selesai	Alokasi	Menunggu	Detail	Batalan		
9	31/05/2018	Handi Septiawan	pulang-pergi	tidak	31/05/2018	09:00	sudah-selesai	Alokasi	Menunggu	Detail	Batalan		
10	30/05/2018	Bintoro	sekali-jalan	tidak	31/05/2018	11:00	dibatalkan	Alokasi	Menunggu	Detail	Batalan		

Showing 1 to 10 of 478 entries

Previous 1 2 3 4 5 ... 48 Next

*Tombol Alokasi dan Menunggu akan aktif (dapat diklik) apabila status pengajuan telah disetujui

Gambar 3.2 Halaman Daftar Pengajuan

Pada halaman daftar pengajuan petugas Admin mengalokasikan peminjaman yang telah disetujui. Alokasi dilakukan dengan menentukan kendaraan yang dipakai beserta sopirnya. Selain itu petugas Admin juga dapat membatalkan pengajuan.

Dengan aplikasi web ini petugas Admin membuat laporan riwayat penggunaan kendaraan. Laporan dapat diunduh berdasarkan *filter* yang dipilih. Laporan berupa berkas Excel.

Cari Laporan Berdasarkan

Dari Sampai

[Cari Laporan](#)

Show entries Search:

Tgl. Pengajuan	Tgl. Peminjaman	Tgl. Selesai	Nama Peminjam	Nama Pemakai	Pengemudi	Kendaraan	Jarak (KM)	Detail
03/04/2018 09:21	03/04/2018 10:00	03/04/2018 13:00	mnzulkamain	deden	ADI SULISTYO	TOYOTA INNOVA - B 2067SOV	9.311	Lihat Detail
03/04/2018 08:34	03/04/2018 10:00	03/04/2018 14:00	diasminisari	zuchrizan zain	ANDY RAMA	TOYOTA INNOVA - B2624SOT	11.029	Lihat Detail
03/04/2018 09:53	03/04/2018 10:00	03/04/2018 13:30	ahwinarso	ahwinarso	ITANG WAHYUDI	TOYOTA INNOVA - B 2785.SOU	7.693	Lihat Detail
03/04/2018 09:53	03/04/2018 10:00	03/04/2018 13:30	ahwinarso	ahwinarso	ITANG WAHYUDI	TOYOTA INNOVA - B 2785.SOU	7.693	Lihat Detail
03/04/2018 09:21	03/04/2018 10:00	03/04/2018 13:00	mnzulkamain	deden	ADI SULISTYO	TOYOTA INNOVA - B 2067SOV	9.311	Lihat Detail
03/04/2018 09:21	03/04/2018 10:00	03/04/2018 13:00	mnzulkamain	deden	ADI SULISTYO	TOYOTA INNOVA - B 2067SOV	9.311	Lihat Detail
03/04/2018 08:32	03/04/2018 08:45	03/04/2018 11:30	diasminisari	yusuf asmara	MULYONO	TOYOTA INNOVA 2 - B 2358.SOU	4.445	Lihat Detail
02/04/2018 15:21	03/04/2018 08:00	03/04/2018 10:00	annovian	Erian	Noke	TOYOTA - B 2359.SOT	10.893	Lihat Detail
02/04/2018 15:18	03/04/2018 08:00	03/04/2018 04:00	mulyani	Heidy Josephine	ADI SULISTYO	TOYOTA INNOVA - B 2067SOV	11.029	Lihat Detail
02/04/2018 15:18	03/04/2018 08:00	03/04/2018 04:00	mulyani	Heidy Josephine	ADI SULISTYO	TOYOTA INNOVA - B 2067SOV	11.029	Lihat Detail

Showing 31 to 40 of 502 entries

Previous 1 2 3 4 5 ... 51 Next

Export Laporan Ke: [EXCEL](#)

Gambar 3.3 Halaman Laporan

Detail Perjalanan

Status Pengajuan	Selesai
Tanggal Pengajuan	2018-04-03
Tanggal Pinjam	2018-04-03 10:00
Nama Pemesan	mnzulkarnain
Email Pemesan	mnzulkarnain@skkmigas.go.id
No HP. Pemesan	628119440910
Pemesan dan Pemakai orang yang sama	<input checked="" type="checkbox"/>
Nama Pemakai	deden
Email Pemakai	
No. HP Pemakai	081386126752
Jenis Perjalanan	pulang-pergi
Keperluan Pinjam	pengiriman invoice ke pertamina pusat dan dja kemenkeu
Tujuan Perjalanan	Jalan Medan Merdeka Timur No.1A, RT.2/RW.1, Gambir, Kota Jakarta Pusat, Daerah Khusus Ibukota Jakarta 10110, Indonesia
Menginap	Tidak, 0 hari
Nama Pengemudi	ADI SULISTYO
Nama Kendaraan	TOYOTA INNOVA - B 2067SOV

Penilaian Submitter ★★★★★

[Kembali](#)

Gambar 3.4 Halaman Detail Pengajuan

3.1.2.2 Mobile Client

Mobile client berupa aplikasi berbasis Android yang digunakan untuk pengajuan peminjaman kendaraan untuk pegawai, penyetujuan peminjaman yang dilakukan oleh atasan dan pemberitahuan penugasan kepada sopir. Petugas yang ingin meminjam kendaraan untuk keperluan rapat atau kedinasan harus melalui pengajuan menggunakan aplikasi ini. Informasi yang perlu disampaikan untuk dapat mengajukan peminjaman kendaraan yaitu keperluan pinjam, menginap atau tidak, tanggal dan jam berangkat sampai pulang, pulang pergi atau sekali jalan.

2:43 PM 1.68K/s 100%

Form Pengajuan

KEPERLUAN PEMINJAMAN :

Menyerahkan Invoice

Mengingat

Tidak Ya

WAKTU PEMINJAMAN:

Tanggal 02-06-2018

Dari Jam 14:00 WIB

Sampai Jam 05:00 WIB

JENIS PERJALANAN

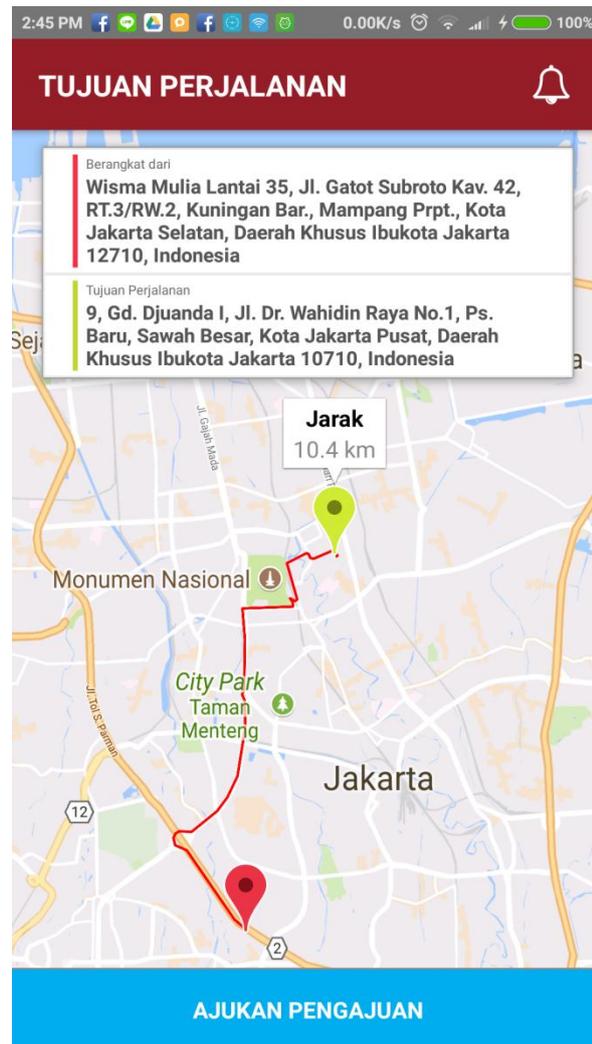
Sekali Jalan Pulang Pergi

Tujuan Perjalanan

Gambar 3.5 Formulir Pengajuan Peminjaman Kendaraan

Setelah menentukan detail pengajuan selanjutnya peminjam menentukan lokasi penjemputan dan tujuan perjalanan. Pada proses ini peminjam memilih titik pada peta lokasi penjemputan dan tujuannya. Setelah selesai memilih selanjutnya melalui layanan Google Map dibuatkan jalur atau rute dari lokasi awal ke lokasi tujuan serta menghitung jaraknya.

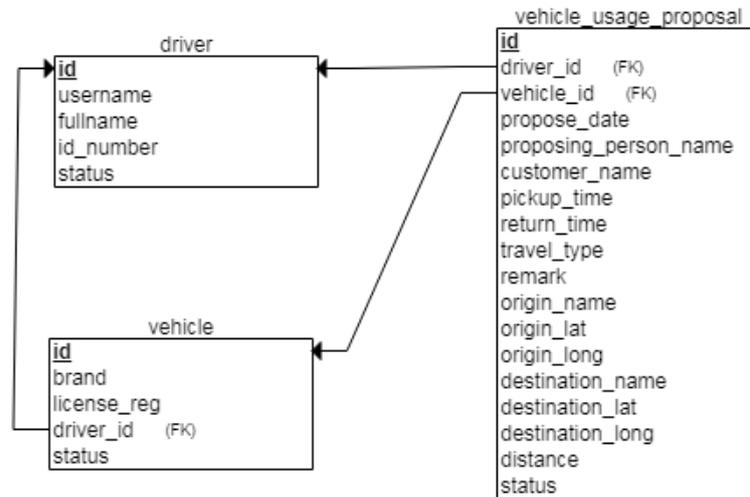
Setelah semua detail peminjaman diisi oleh peminjam maka informasi tersebut dikirimkan ke *server* dan pegawai menunggu pemberitahuan selanjutnya apakah pengajuannya disetujui atau tidak. Informasi detail peminjaman yang dikirim yaitu nama penumpang, nomer telepon penumpang, keperluan, mengingat atau tidak, tanggal dan jam berangkat, tanggal dan jam pulang, pulang pergi atau sekali jalan, lokasi penjemputan, lokasi tujuan, jarak antara lokasi penjemputan dan tujuan.



Gambar 3.6 Formulir Penentuan Tujuan Perjalanan

3.1.2.3 Analisis Database

Vehicle Management System menggunakan *database* dengan perangkat lunak PostgreSQL. Berikut adalah struktur database yang sedang berjalan di *Vehicle Management System*. Struktur database yang diteliti hanya sebagian yang berhubungan dengan sistem yang akan dibangun. Beberapa tabel pada *database* VMS yang berhubungan yaitu *vehicle_usage_proposal*, *driver* dan *vehicle*.



Gambar 3.7 Relasi Tabel Database Vehicle Management System

Tabel *vehicle_usage_proposal* adalah tabel untuk menyimpan *record* pengajuan yang dilakukan oleh pegawai. Tabel *vehicle* adalah tabel untuk menyimpan informasi kendaraan dinas dan tabel *driver* adalah tabel untuk menyimpan profil sopir. Pada tabel *vehicle* terdapat atribut yaitu:

1. *propose_date*, tanggal dan jam pengajuan dilakukan.
2. *driver_id*, setelah dialokasikan maka pengajuan telah ditentukan pengemudanya siapa.
3. *vehicle_id*, setelah dialokasikan maka pengajuan telah ditentukan memakai kendaraan yang mana.
4. *proposing_person_name*, nama pegawai yang mengajukan peminjaman kendaraan.
5. *customer_name*, nama penumpang kendaraan yang diajukan bisa saja pegawai yang mengajukan itu sendiri.
6. *pickup_time*, waktu penjemputan berupa tanggal dan jam berangkat.
7. *return_time*, waktu selesai peminjaman berupa tanggal dan jam pulang.
8. *travel_type*, apakah peminjaman sekali jalan atau pulang pergi.
9. *remak*, keterangan peminjaman berupa deskripsi keperluan apa. Misalkan: *meeting*, menyerahkan dokumen ke kementerian A.

10. *origin_name*, alamat penjemputan
11. *destination_name*, alamat tujuan
12. *origin_lat* *origin_long*, titik penjemputan yang dipilih oleh pegawai yang mengajukan dengan menandai titik pada peta di aplikasi *mobile client*.
13. *destination_lat/destination_long*, titik lokasi yang dipilih oleh pegawai yang mengajukan dengan menandai titik pada peta di aplikasi *mobile client*.
14. *distance*, perkiraan jarak antara lokasi awal dengan tujuan yang dihitung dengan API Google Map pada *mobile client*.
15. *status*, status pengajuan bisa baru, dialokasikan dan selesai.

3.1.3 Analisis Prosedur Yang Sedang Berjalan

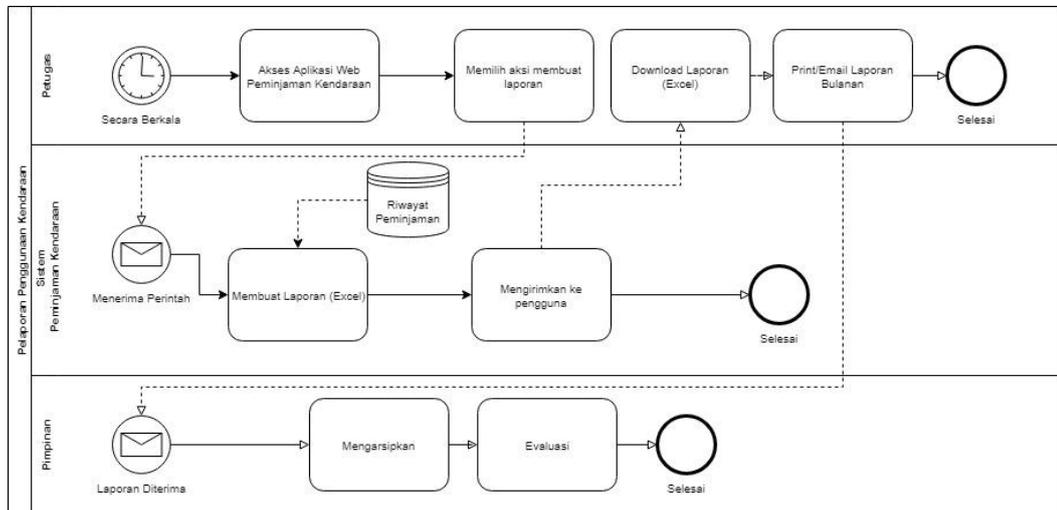
Analisis prosedur yang sedang berjalan menjelaskan proses analisis terhadap sistem yang sedang berjalan di SKK Migas beserta cara kerja dan masalah yang dihadapi untuk dapat dijadikan landasan dalam penelitian. Analisis prosedur yang sedang berjalan di SKK Migas diuraikan berdasarkan kejadian yang ada secara berurut dengan menggunakan diagram proses bisnis BPMN (*Business Process Model Notation*).

3.1.3.1 Prosedur Pelaporan Penggunaan Kendaraan Dinas

Laporan dibuat melalui aplikasi web oleh petugas.

1. Petugas masuk kedalam aplikasi web di jaringan SKK Migas.
2. Petugas masuk ke halaman pembuatan laporan pada aplikasi web.
3. Petugas memilih periode pelaporan berdasarkan hari atau bulan..
4. Aplikasi web meminta ke server untuk membuat laporan berdasarkan periode yang dipilih.
5. Server membuat laporan berupa berkas *excel* riwayat peminjaman kendaraan yang berisi nama peminjam, nama sopir, tanggal dan waktu peminjaman, tanggal dan waktu selesai, keperluan, lokasi keberangkatan dan lokasi tujuan.
6. Petugas mengunduh laporan sesuai periode yang dipilih berupa berkas *excel*.

7. Petugas menyerahkan laporan bulanan penggunaan kendaraan berbentuk *hardcopy excel* kepada pimpinan.



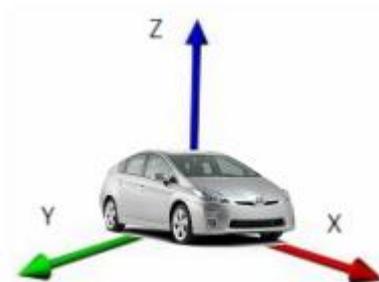
Gambar 3.8 BPMN Proses Pelaporan Penggunaan Kendaraan

3.1.4 Analisis Kebutuhan Data

Analisis kebutuhan data menguraikan data apa saja yang diolah dan akan menjadi keluaran apa saja. Data yang akan dianalisis yaitu data dari akselerometer, gps dan peminjaman kendaraan dari sistem yang sedang berjalan yaitu *vehicle management system*.

3.1.4.1 Analisis Data Akselerometer

Akselerometer menghasilkan data pergerakan dalam sumbu x dan y Nilai sumbu selanjutnya diubah menjadi nilai *g-force* dan dengan *threshold* yang telah ditentukan dapat dikenali gerakan kendaraan saat mengerem, melaju atau berbelok.



Gambar 3.9 Arah Akselerometer Pada Kendaraan

Nilai *threshold* G-Force yang digunakan untuk mendeteksi pada akselerometer kendaraan dilihat pada tabel 3.1.

Tabel 3.1 Nilai Akselerometer Pada Kendaraan

Sumbu	Threshold	Keterangan
X	0.3g	Melaju dengan cepat
-X	0.6g	Mengerem dengan keras
Y	0.5g	Berbelok Kanan dengan cepat
-Y	0.5g	Berbelok ke kiri dengan cepat

Pada tabel 3.2 adalah contoh data akselerometer pada kendaraan.

Tabel 3.2 Contoh Data Akselerometer Pada Kendaraan

Sopir	Kendaraan	X	Y	Z	Keterangan
Pratama	B 1234 CD	0.12g	0.15g	0.11g	Normal
Revo	B 1111 DC	0.51g	0.12g	0.13g	Nilai G Force X \geq 0.3g Melaju dengan cepat
Kurniawan	B 2222 EF	-0.62g	0.15g	0.10g	Nilai G Force X \geq 0.6g Mengerem dengan keras
Adi	B 3333 DD	0.22g	0.52g	0.12g	Nilai G Force Y \geq 0.5g Berbelok dengan cepat

3.1.4.2 Analisis Data Peminjaman Kendaraan

Data dari yang didapat dari *vehicle management system* yaitu riwayat peminjaman kendaraan. Peminjaman kendaraan mempunyai atribut yaitu kendaraan, sopir, penumpang, tanggal dan waktu, lokasi penjemputan, lokasi tujuan, status dan estimasi jarak. Dari data ini selanjutnya dilakukan pengolahan sehingga menjadi informasi baru.

Tabel 3.3 Contoh Data peminjaman Kendaraan

Penumpang	Tujuan (Koordinat)	Sopir	Estimasi Jarak (KM)	Status
Yuliana	-6.12345, 106.123	Ryan	10	Selesai
Rudi	-6.11111, 106.222	Sulaiman	15	Ditolak
Erlan	-6.33333, 106.333	Noke	12	Dibatalkan
Seno	-6.44444, 106.333	Sugianto	16	Disetujui

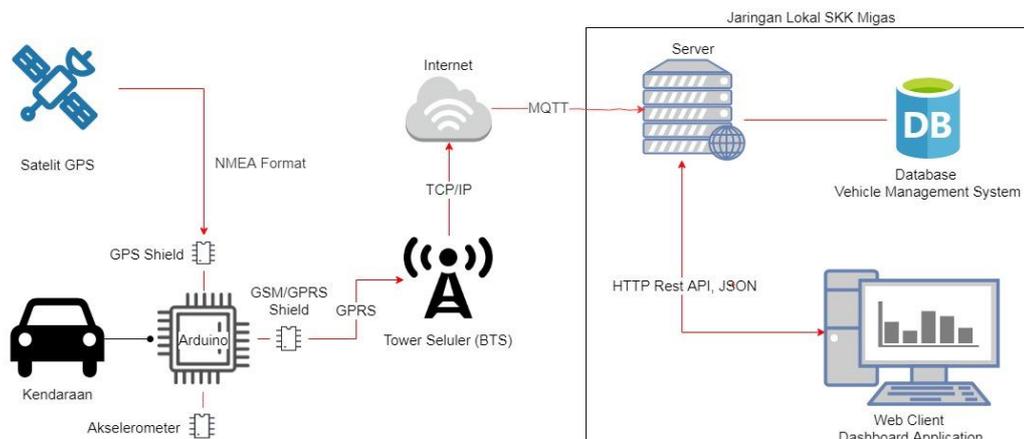
Dari data peminjaman yang didapat diolah dilakukan untuk selanjutnya dapat divisualisasikan. Berikut beberapa pengolahan yang dapat dilakukan dapat dilihat pada tabel 3.4.

Tabel 3.4 Pengolahan Data Peminjaman

Field	Pengolahan	Bentuk Visualisasi
Penumpang	Penumpang yang sering menggunakan fasilitas kendaraan	<i>Bar-chart</i> Top 10 Penumpang yang paling banyak menggunakan fasilitas kendaraan
Tujuan	Tujuan yang sering dikunjungi	<i>Heatmap, Cluster Marker</i> tujuan yang sering dikunjungi
Estimasi jarak	Rata-rata jarak peminjaman	Angka
Sopir	Sopir yang sering mendapatkan tugas	<i>Bar-chart</i> top 10 sopir yang paling banyak mendapatkan tugas
Status	Jumlah peminjaman yang disetujui, ditolak, dibatalkan dan sudah selesai.	<i>Pie-chart</i> persentase peminjaman yang disetujui, ditolak, dibatalkan dan sudah selesai.

3.1.5 Analisis Sistem Yang Akan Dibangun

Sistem yang akan dibangun adalah sebuah sistem yang dapat mengumpulkan data kendaraan dikolaborasikan dengan data peminjaman kendaraan dari *Vehicle Management System* lalu divisualisasikan menjadi aplikasi web *dashboard*. Sistem yang akan dibangun berbasis teknologi *Internet of Things* (IoT) dengan perangkat berupa mikrokontroler dan sensor-sensor yang dapat terhubung dengan Internet dan dapat mengumpulkan informasi terkait kendaraan selama operasional.



Gambar 3.10 Arsitektur Sistem Yang Akan Dibangun

Pada gambaran arsitektur sistem komponen-komponen yang terlibat yaitu kendaraan, perangkat mikrokontroler beserta sensor-sensor, *Web Server*, *Web Client* dan *Database Vehicle Management System*.

Perangkat mikrokontroler yang digunakan adalah mikrokontroler berbasis Arduino dengan modul GPS untuk merekam lokasi, GSM/GPRS untuk komunikasi data dan akselerometer untuk merekam pergerakan kendaraan. Perangkat mikrokontroler menggunakan daya dari baterai dan menggunakan *casing* yang portabel sehingga memudahkan pemasangan pada kendaraan. Selain menggunakan daya baterai, perangkat ini mampu untuk diberi daya menggunakan USB yang ada pada kendaraan sehingga perangkat dapat bekerja dalam kondisi kendaraan menyala ataupun tidak.

Modul GPS merekam lokasi kendaraan selama perjalanan dengan berkomunikasi dengan satelit GPS. Modul GPS dapat berkomunikasi dengan satelit dengan bergantung pada antena GPS. Informasi yang didapat dari sensor GPS yaitu lokasi berupa *latitude* dan *longitude*, kecepatan, arah (dalam 360 derajat), tanggal dan waktu pemosisian dan jumlah satelit yang menjadi partisipan. Informasi lokasi dari GPS ini nantinya jadi bahan analisa spasial bagaimana kendaraan beroperasi setiap harinya.

Modul GSM/GPRS berfungsi untuk terhubung dengan Internet dan bertukar informasi dengan *server*. Modul ini membutuhkan kartu SIM Card

(*Subscriber Identification Module*) dan terhubung ke jaringan provider. Modul ini juga membutuhkan sinyal ke tower selular atau *Base Transceiver Station* (BTS) untuk terhubung ke jaringan provider. Kuatnya sinyal ditentukan oleh kualitas antena yang terpasang pada modul, lokasi serta jarak dengan BTS. Setelah terhubung ke jaringan provider selanjutnya menghubungkan ke Internet dengan menggunakan layanan *General Packet Radio Service* (GPRS). Untuk terhubung dengan GPRS diperlukan *Access Point Name* (APN) yang telah disediakan oleh provider.

Setelah terhubung dengan Internet selanjutnya dihubungkan dengan server menggunakan protokol *Messaging Queueing Telemetry Transport* (MQTT). Protokol MQTT ini adalah protokol khusus untuk *machine-to-machine* (M2M) atau *Internet of Things*. Protokol ini digunakan dikarenakan desainnya yang sangat ringan untuk transportasi pesan dengan metode *publish* dan *subscribe*. Dengan menggunakan protokol ini komunikasi data dengan server menjadi lebih terjamin dari segi ukuran, konsumsi daya dan paket data atau *bandwidth*.

Database yang digunakan adalah database yang sudah ada (*existing*) yaitu database *Vehicle Management System* dengan teknologi PostgreSQL. Dilakukan restruktur database untuk menambahkan ruang penyimpanan baru untuk sistem yang akan dibangun sehingga dapat terhubung dengan data yang sudah ada. Untuk menyimpan data lokasi yang berupa *latitude* dan *longitude* diperlukan ekstension PostGIS yang mendukung penyimpanan dan pengolahan data spasial. Extension PostGIS diinstal pada sistem operasi dan diaktifkan pada database *Vehicle Management System*.

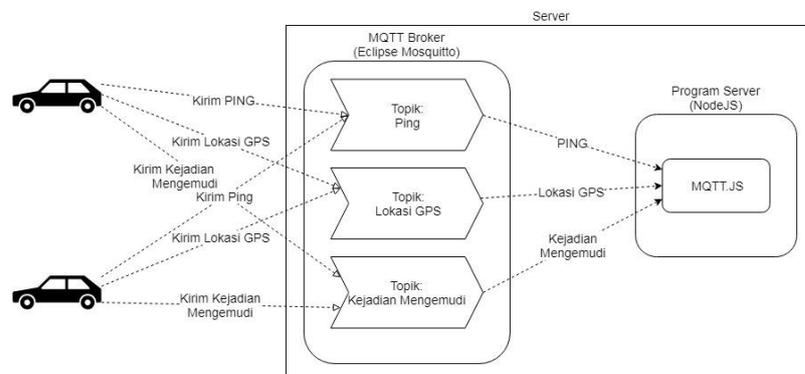
Server berperan sebagai perangkat yang melayani permintaan, menerima masukan dan memproses data. Server juga berperan sebagai gerbang komunikasi untuk perangkat mikrokontroler. Server mampu menerima data dari setiap perangkat mikrokontroler dengan menggunakan protokol *Messaging Queueing Telemetry Transport* (MQTT). Maka dari itu server ini juga dijadikan sebagai broker MQTT.

Server menjalankan program berbasis NodeJS dan terhubung dengan database *Vehicle Management System*. Program ini terhubung dengan broker MQTT yang berjalan di *local*. Program akan melakukan *subscribe* pada MQTT tersebut untuk mengumpulkan data dari sensor. Data yang didapat selanjutnya diproses dan disimpan ke database. Program ini juga menyediakan layanan untuk menerima permintaan data dengan metode REST Service untuk selanjutnya dikonsumsi oleh aplikasi web.

Aplikasi web adalah program yang menjadi antarmuka pengguna dengan sistem. Dari informasi yang dapat dari *server* melalui REST API selanjutnya dilakukan visualisasi dengan bentuk dashboard. Visualisasi data berbentuk peta, grafik dan tabel yang dapat disajikan berdasarkan waktu seperti mingguan, bulanan sampai tahunan serta memungkinkan untuk mengekspor kedalam format file digital PDF sebagai laporan.

3.1.6 Analisis Arsitektur Komunikasi Data

Komunikasi data merupakan suatu hal yang penting dalam pembuatan sistem dengan teknologi *Internet of Things* (IoT). Komunikasi data yang digunakan pada sistem yang akan dibangun menggunakan protokol MQTT (*Messaging Queueing Telemetry Transport*). Protokol MQTT ini menggunakan model *publish* dan *subscribe*. Pada sistem ini akan menggunakan MQTT dari Mosquitto karena *open-source* dan kompatibel dengan perangkat Arduino dan sensor yang digunakan.



Gambar 3.11 Arsitektur Komunikasi Data

Dapat dilihat pada gambar 3.11 bahwa pada komputer server dilakukan instalasi perangkat lunak Eclipse Mosquitto MQTT. Server akan menjalankan *service* MQTT Broker sebagai jembatan komunikasi data.

Perangkat mikrokontroler melalui jaringan Internet dihubungkan dengan broker MQTT. *Library* yang digunakan pada mikrokontroler untuk terhubung ke broker MQTT menggunakan PubSubClient untuk Arduino. Setelah terhubung, perangkat mikrokontroler melakukan *publish* ketika lokasi dari GPS didapat atau ketika terdeteksi rem mendadak (*hard braking*), melaju dengan sontak (*hard acceleration*) dan berbelok dengan cepat (*hard cornering*). Untuk melihat apakah perangkat mikrokontroler *online* ketika kendaraan beroperasi maka pesan PING setiap 10 detik dikirimkan ke topik Ping dengan pengenal berupa ID (*identifier*).

Program server berbasis NodeJS dihubungkan dengan broker MQTT yang berjalan di *localhost*. *Library* yang digunakan untuk terhubung dengan broker MQTT yaitu MQTT.js. Setelah terhubung dengan broker MQTT, program server melakukan *subscribe* ke setiap topik yang tersedia dan membaca serta memproses data yang didapat dari semua sensor yang terhubung.

Pengiriman data berupa teks dengan pemisah tanda koma “,” untuk setiap nilainya. Format dapat dilihat sebagai berikut ini:

1. Format Data GPS

```
[ID_KENDARAAN,RAW_LATITUDE,RAW_LONGITUDE,TAHUN,BULAN,TANGGAL,JAM,MENIT,DETIK]
```

Contoh: 1,-616632630,106820755,2018,08,02,09,05,20

2. Format Data Kejadian (Tanpa Lokasi)

```
[ID_KENDARAAN,TIPE_KEJADIAN,NILAI_AKSELEROMETER]
```

Contoh: 1,B,1232

Tipe Kejadian:

A = Akselerasi keras / *hard accelerating*

B = Rem mendadak / *hard braking*

C = Berbelok keras / *hard cornering*

3. Format Data Kejadian (Dengan Lokasi)

[ID_KENDARAAN, TIPE_KEJADIAN, NILAI_AKSELEROMETER, RAW_LATITUDE, RAW_LONGITUDE

, TAHUN, BULAN, TANGGAL, JAM, MENIT, DETIK]

Contoh: 1,B,1520,-616632630,106820755,2019,08,02,09,05,20

4. Format Data Ping

[ID_KENDARAAN]

Contoh: 1

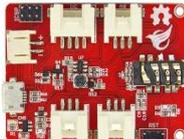
Nilai akselerometer yang dikirim merupakan nilai *raw* yang mana perlu diolah lagi. Untuk mendapatkan nilai G-Force pada nilai akselerometer yaitu dengan membaginya dengan nilai LSB/g (*least significant bit per g*) pada akselerometer yaitu 8192 pada akselerometer MPU6050 dengan pengaturan sensitifitas 4g. Jadi dari contoh nilai akselerometer 1520 dibagi dengan 8192 didapat nilai G-Force 0,18. Nilai ini yang selanjutnya disimpan di *database*.

Nilai latitude dan longitude yang terkirim yaitu nilai *raw* dalam bentuk sepersepjuta derajat. Jadi untuk mendapatkan nilai latitude dan longitude sebenarnya adalah dengan membaginya dengan 1000000 (satu juta). Contohnya nilai *raw* latitude -616632630 dan longitude 106820755 dibagi dengan 1000000 menghasilkan nilai latitude -6,16632630 dan longitude 106,6820755. Nilai ini yang selanjutnya disimpan di *database*.

3.1.7 Analisis Perangkat Keras

Berikut adalah perangkat beserta komponen-komponen yang digunakan pada sistem yang akan dibangun dilihat pada tabel 3.5.

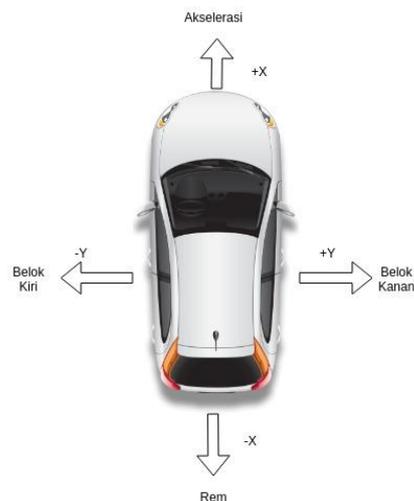
Tabel 3.5 Perangkat yang digunakan

Nama	Gambar	Deskripsi	Sumber Gambar
Papan Elecrow 32u4 with A7 GPS/GPRS/GSM		a) Satu paket GPS, GSM dan GPRS. b) Daya 3.7V sampai 5V	https://www.elecrow.com/wiki/index.php?title=File:32U4_with_GSM_GPS.jpg

Crowtail MPU-6050		<ul style="list-style-type: none"> a) Microelectromechanical systems (MEMS) mendukung akselerometer dan gyroscope b) Skala akselerasi kurang lebih 16g 	https://www.electrow.com/wiki/images/7/79/Crowtail-MPU6050-Accelerometer%26Gyroscope-1.jpg
Baterai standar 3.7V 820mAh		<ul style="list-style-type: none"> a) Rechargeable Li-Po atau Li-Ion. b) Direkomendasikan 1500 mAh. 	https://cdn-shop.adafruit.com/1200x900/258-00.jpg
Case		<ul style="list-style-type: none"> a) Aklirik 	https://aws.robu.in/wp-content/uploads/2017/04/robu-1-16.jpg
SIM Card		<ul style="list-style-type: none"> a) GSM b) Ukuran Micro c) Aktif Paket Data Internet 	https://support.apple.com/library/content/dam/edam/applecare/images/en_US/iphone/micro-sim-card.png

3.1.8 Analisis Deteksi Kejadian Mengemudi

Terdapat 3 *axis* pada akselerometer yaitu *axis* x, y dan z. Ketiga *axis* itu digunakan untuk merasakan gerakan dan mengenali bagaimana kendaraan berakselerasi, mengerem ataupun berbelok. Berikut adalah konfigurasi arah akselerasi pada kendaraan.



Gambar 3.12 Konfigurasi Arah Akselerasi Pada Kendaraan

Pada kendaraan, perangkat dan sensor yang dipasang diletakkan sesuai dengan arah kendaraan melaju. Arah akselerometer disejajarkan dengan kendaraan. *Axis x* pada akselerometer sejajar dengan arah melaju lurus pada kendaraan. Berikut adalah penjelasan arah akselerometer pada kendaraan.

Tabel 3.6 Tabel Akselerometer Untuk Mengenali Tipikal Mengendarai

<i>Axis</i>	Arah	Tipikal Mengendarai
X dan -X	Depan / Belakang	Melaju (<i>accelerating</i>) Mengerem (<i>braking/decelerating</i>)
Y dan -Y	Kiri / Kanan	Berputar (<i>turning</i>), Berbelok (<i>swerving</i>)

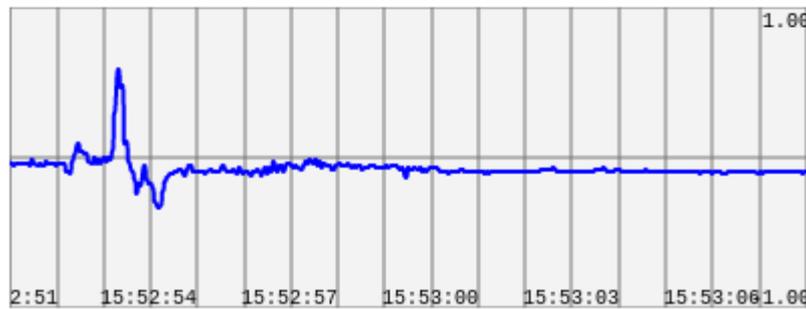
Deteksi dilakukan dengan membandingkan nilai pergerakan kendaraan dengan nilai *threshold* yang ditentukan pada tabel dibawah ini. Nilai deteksi yang digunakan adalah nilai dengan kategori *hard*.

Tabel 3.7 Target G-Force Untuk Manuver Mengemudi [28]

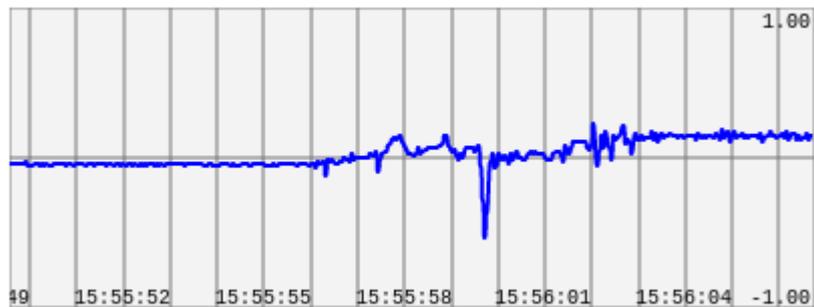
Manuver Mengemudi	Target G-Force
Berbelok (<i>Cornering</i>)	
Mild Left	0.2 – 0.3 G
Moderate Left	0.3 – 0.4 G
Hard Left	≥ 0.5 G
Mild Right	0.2 – 0.3 G
Moderate Right	0.3 – 0.4 G
Hard Right	≥ 0.5 G
Mengerem (<i>Braking</i>)	
Mild	0.4 – 0.5 G
Moderate	0.5 – 0.6 G
Hard	≥ 0.6 G
Akselerasi (<i>Accelerating</i>)	
Mild	0.2 G
Hard	≥ 0.3 G

3.1.8.1 Akselerasi dan Mengerem

Akselerasi dan mengerem dapat dikenali dengan mengamati pergerakan nilai *axis x* pada akselerometer. Saat kendaraan mulai melaju maka nilai *x* pada akselerometer akan menuju positif. Contohnya saat mengerem atau berakselerasi, keadaan semula nilai *axis x* berada di posisi disekitar nol selanjutnya menurun drastis saat mengerem dan bertambah drastis saat berakselerasi. Berikut adalah grafik saat kendaraan mengerem dan berakselerasi dilihat pada gambar 3.13 dan gambar 3.14.

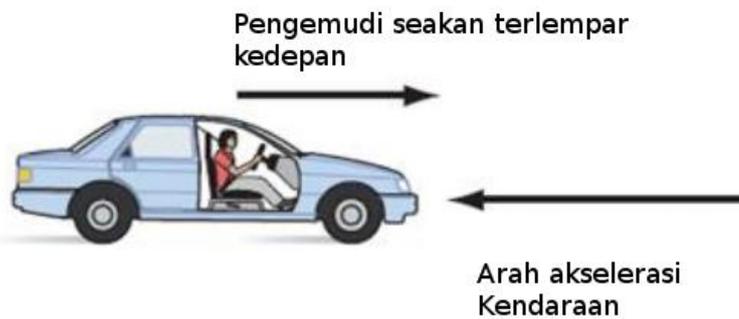


Gambar 3.13 Grafik *axis x* saat kendaraan berakselerasi dengan cepat.



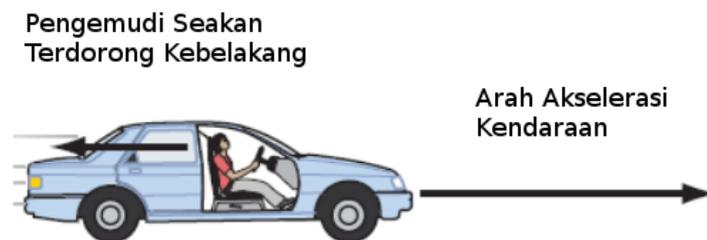
Gambar 3.14 Grafik *axis x* saat kendaraan rem mendadak

Saat mengerem dengan keras maka pengereman dengan keras dapat dikenali dengan nilai *x* akselerometer lebih dari 0.6G (G-Force) [28]. Pengemudi dan penumpang akan merasakan seperti mereka sedang terlempar kedepan menuju kaca depan kendaraan.



Gambar 3.15 Kondisi Saat Kendaraan Mengerem

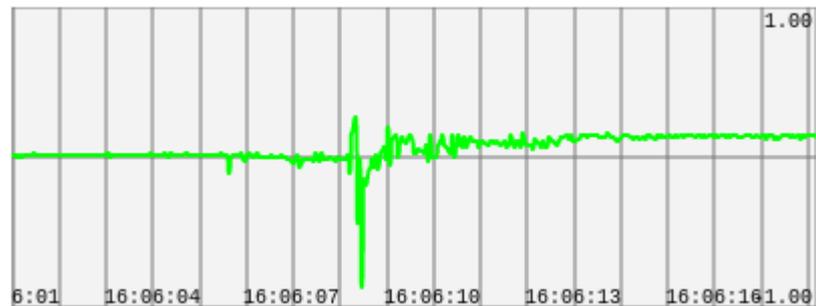
Sedangkan saat berakselerasi dengan keras maka nilai akselerometer akan menunjukkan lebih dari atau sama dengan 0.3 G (G-Force) pada kendaraan [28]. Pada saat kendaraan berakselerasi dengan 0.3 G penumpang atau pengemudi akan merasakan seperti mereka sedang didorong ke belakang ke kursi kendaraan.



Gambar 3.16 Kondisi Saat Kendaraan Berakselerasi

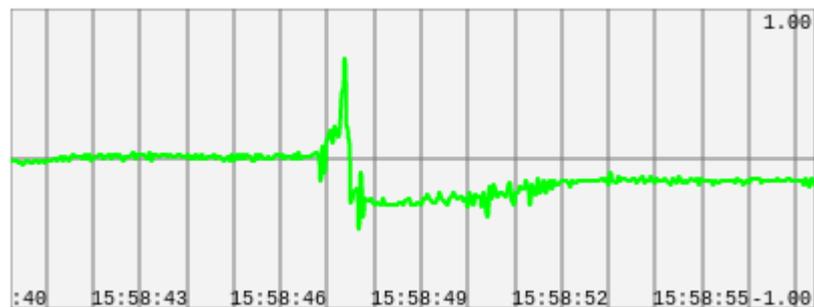
3.1.8.2 Berbelok

Berbelok ke kiri maupun ke kanan dapat dikenali dengan melihat nilai *Y-axis* pada akselerometer. Pada saat kendaraan belum berbelok maka nilai *Y-axis* berada di posisi sekitar nol. Saat mulai berbelok maka posisi *Y-axis* akan naik atau turun sesuai arah berbeloknya. Belok ke kiri akan menyebabkan nilai *Y-axis* ke arah positif dan belok ke kanan ke arah negatif.



Gambar 3.17 Grafik saat kendaraan berbelok kekanan

Pada saat berbelok ke kiri maka nilai *Y-axis* akan turun drastis ke arah positif berbanding dengan ketajaman dan kecepatan berbelok.



Gambar 3.18 Grafik saat kendaraan berbelok ke kiri

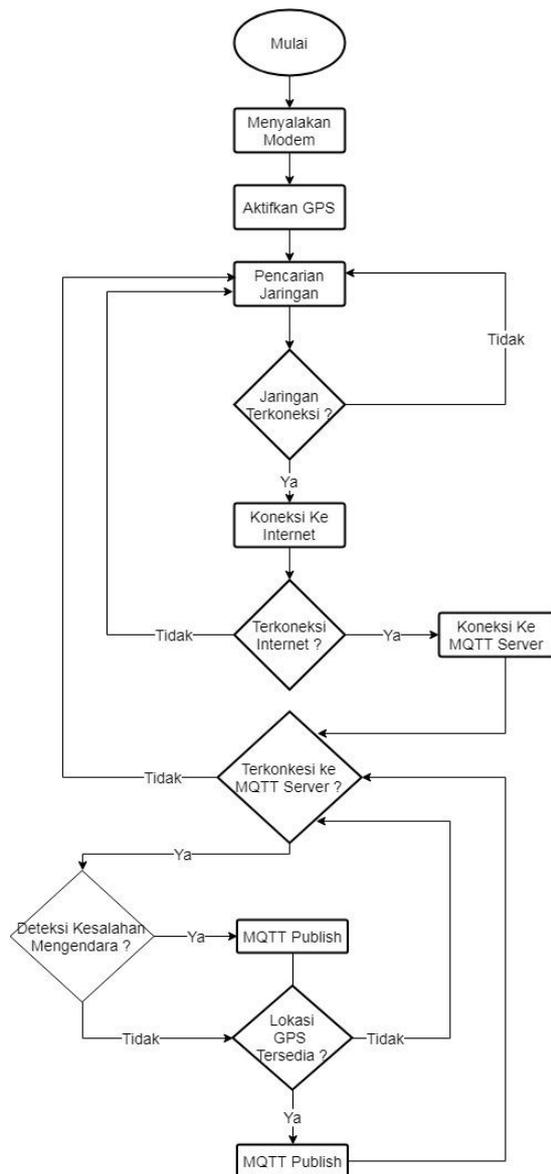
Kendaraan yang berbelok dengan keras dapat dikenali dengan nilai akselerometer pada *axis Y* yang melebihi atau sama dengan 0.5 G (*g-force*) [28]. Pada saat berbelok dengan nilai *g-force* 0.5g pengemudi dan penumpang akan merasakan seperti sedang melaju lurus dan tertarik berlawanan arah berbelok.



Gambar 3.19 Kondisi Saat Kendaraan Berbelok

3.1.9 Analisis Alur Program Mikrokontroler

Program pada mikrokontroler dimulai dengan menyalakan modem GPS/GSM/GPRS. Terdapat dua bagian dalam memprogram Arduino yaitu *Setup* dan *Loop*. Pada bagian *Setup* modem dinyalakan beserta menghubungkannya ke jaringan dan Internet. setelah terhubung selanjutnya mengaktifkan GPS. Setelah terkoneksi internet dan GPS aktif selanjutnya melakukan koneksi ke server MQTT. Informasi lokasi yang didapat GPS akan dikirimkan melalui protokol MQTT.



Gambar 3.20 Diagram Alur Program Mikrokontroler

3.1.9.1 Alur Program Deteksi Kejadian Mengendara

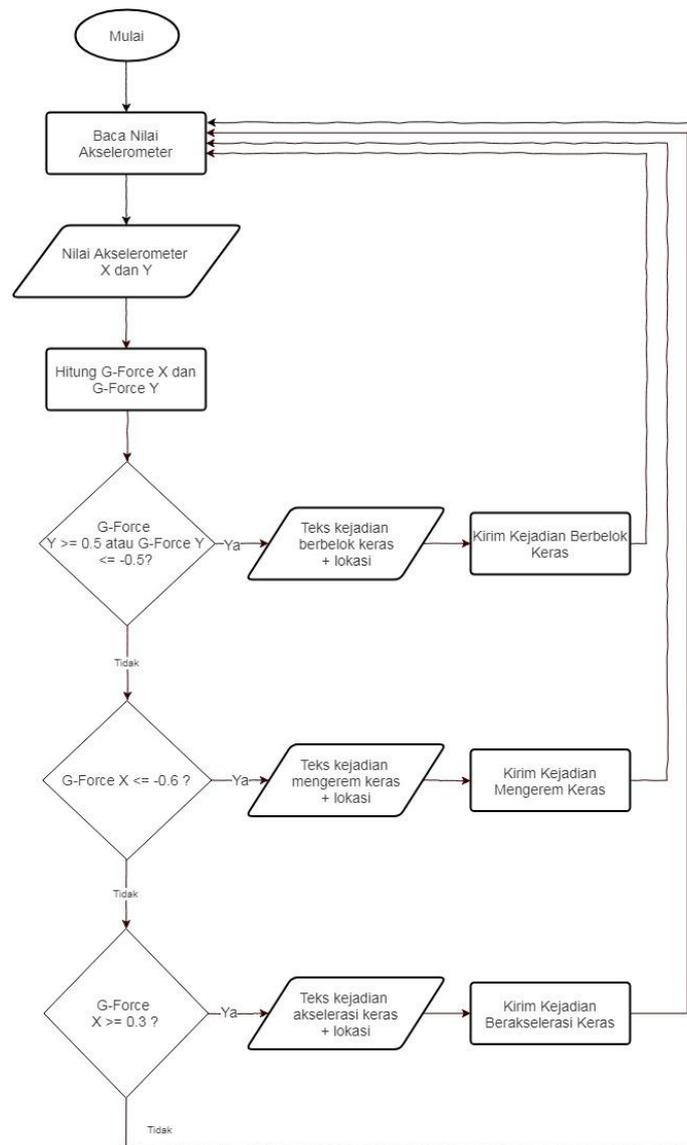
Pada program mikrokontroler kejadian mengemudi dideteksi dengan membandingkan nilai *g-force* yang didapat akibat pergerakan kendaraan. Mikrokontroler membaca nilai *x* dan *y* pada sensor akselerometer selanjutnya dikonversi menjadi nilai *g-force*. Dengan menggunakan pustaka DMP (*Digital Motion Processing*) pada akselerometer MPU6050 maka didapat nilai akselerometer linear. Nilai *g-force* didapatkan dengan menghitung nilai

akselerometer linear dengan membaginya dengan nilai sensitivitas akselerometer pada sensor MPU-6050:

Tabel 3.8 Nilai Skala Akselerasi MPU-6050

Jangkauan Skala Akselerasi (g)	Sensitifitas Akselerasi (LSB/g)
±2	16384
±4	8192
±8	4096
±16	2048

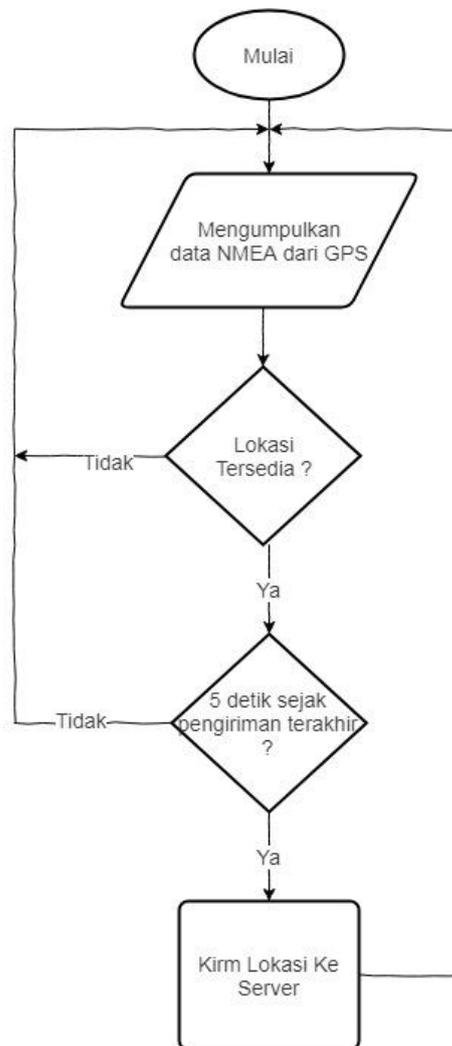
Jangkauan skala akselerasi yang dipakai yaitu ± 4 jadi nilai sensitifitas akselerasi yang dipakai adalah 8192. Hasil perhitungan nilai *g-force* yang didapatkan selanjutnya diproses dan dibandingkan dengan nilai *threshold* kejadian mengemudi. Setiap terdeteksi suatu kejadian mengemudi maka perangkat langsung mengirimkannya ke *server*.



Gambar 3.21 Diagram Alur Pendeteksian Kejadian Mengemudi

3.1.9.2 Alur Program Lokasi GPS

Data yang didapatkan dari modul GPS mempunyai format NMEA. Dengan menggunakan pustaka MicroNMEA untuk Arduino format NMEA ini selanjutnya diolah dan menghasilkan data lokasi yang dibutuhkan berupa *latitude*, *longitude*, tanggal serta waktu lokasi didapatkan. Lokasi ini selanjutnya dikirimkan ke server setiap 5 detik.



Gambar 3.22 Diagram Alur Pengiriman Lokasi GPS

3.1.10 Analisis Kebutuhan Non-Fungsional

Analisis kebutuhan non-fungsional menjelaskan kebutuhan sistem yang diperlukan yang mendukung berjalannya sistem yang dibangun. Kebutuhan non-fungsional memiliki tiga tahapan yaitu kebutuhan perangkat keras, kebutuhan perangkat lunak dan pengguna sistem yang harus dipenuhi.

Tabel 3.9 Spesifikasi Kebutuhan Perangkat Lunak Non-Fungsional

No	Kode Kebutuhan	Deskripsi Kebutuhan
1.	SKPL-NF01	Kebutuhan Perangkat Keras dan Perangkat Lunak Komputer <i>Server</i>
2.	SKPL-NF02	Kebutuhan Perangkat Keras dan Perangkat Lunak Komputer <i>Client</i>
3.	SKPL-NF03	Kebutuhan Perangkat Keras Mikrokontroler dan Sensor
4.	SKPL-NF04	Kebutuhan Pengguna

3.1.10.1 Analisis Perangkat Keras

Spesifikasi perangkat keras yang perlu dipenuhi untuk berjalannya sistem evaluasi kendaraan dinas yaitu:

a) Perangkat Keras Komputer Server

Perangkat keras untuk *server* yaitu sebuah perangkat komputer yang dapat melayani selama 24 jam setiap hari. Perangkat komputer untuk *server* harus dapat diakses melalui Internet baik itu secara langsung atau dengan cara *port forward* dari router. Berikut spesifikasi untuk *web server* yang dibutuhkan dilihat pada tabel 3.10.

Tabel 3.10 Spesifikasi Perangkat Keras Komputer Server

No.	Perangkat Keras	Spesifikasi
1	Prosesor	Prosesor dengan kecepatan 2.0 GHz Arsitektur x64
2	RAM	4 GB DDR3
3	Koneksi Internet	1 Mbps upload/download
4	Media Penyimpanan	Hard disk drive dengan Free space 20 GB

b) Perangkat Keras Komputer Client

Komputer *Client* digunakan sebagai antarmuka pengguna terhadap sistem. Adapun kebutuhan perangkat keras untuk Komputer *Client* yaitu sebuah komputer yang dapat terhubung dengan server melalui jaringan lokal dengan spesifikasi yang cukup untuk membuka sebuah aplikasi web. Adapun kebutuhan perangkat keras untuk komputer *client* yaitu pada tabel 3.11.

Tabel 3.11 Spesifikasi Perangkat Keras Komputer Client

No.	Perangkat Keras	Spesifikasi
1	Prosesor	Prosesor dengan kecepatan 2.0 GHz
2	RAM	2 GB DDR3
3	Koneksi Internet / Jaringan	1 Mbps upload / download Terhubung jaringan lokal dengan <i>web server</i>
4	Media Penyimpanan	Hard disk drive dengan Free space 5 GB

c) Perangkat Keras Mikrokontroler

Perangkat mikrokontroler yang digunakan merupakan perangkat yang mampu untuk berkomunikasi melalui Internet. Perangkat yang bergerak sama halnya seperti *smartphone* menggunakan jaringan GPRS untuk dapat terhubung dengan Internet. Maka dari itu perangkat IoT perlu mendukung GPRS untuk komunikasi data dan sensor-sensor yang sesuai dengan kebutuhan. Adapun spesifikasi perangkat mikrokontroler yaitu dilihat pada tabel 3.12.

Tabel 3.12 Spesifikasi Perangkat Keras Perangkat Mikrokontroler

No	Perangkat Keras	Spesifikasi
1	Papan Mikrokontroler	Kecepatan CPU \geq 16 MIPS/DMIPS Memory 32 KB Tipe memori Flash
2	Modul GPS	Mendukung GPS dan AGPS
3	Modul GSM/GPRS	GPRS kelas 10 Quad-band sampai 1900MHZ Mendukung TCP/IP dan HTTP Support SMS
4	Modul MEMS <i>Accelerometer</i>	Daya 3~5V Onboard dengan onboard regulator Mode komunikasi dengan IIC Rentang akselerasi: +/- 2g, +/-4g, +/- 8g, +/-16g

3.1.10.2 Analisis Kebutuhan Perangkat Lunak

Dalam analisis kebutuhan perangkat lunak menjelaskan spesifikasi perangkat lunak yang dibutuhkan dan mendukung untuk digunakan dalam implementasi. Adapun perangkat lunak yang dibutuhkan untuk berjalannya sistem evaluasi kendaraan dinas SKK Migas adalah sebagai berikut.

a) Perangkat Lunak Server

Untuk mendukung berjalannya sistem dibutuhkan perangkat lunak yang dapat dilihat pada tabel 3.13.

Tabel 3.13 Spesifikasi Perangkat Lunak Untuk Server

No.	Kategori	Spesifikasi
1	Sistem Operasi	Berbasis Linux Console-based / No-GUI
2.	Perangkat Lunak Server	NodeJS Supervisor Mosquitto MQTT Broker

b) Perangkat Lunak Komputer *Client*

Perangkat pada web client yaitu perangkat lunak yang dibutuhkan untuk dapat mendukung akses terhadap sistem dan menjadi antarmuka antara pengguna dengan sistem. Adapun kebutuhan perangkat lunak untuk *Web Client* yaitu pada tabel 3.14.

Tabel 3.14 Spesifikasi Perangkat Lunak Komputer Client

No.	Kategori	Spesifikasi
1.	Sistem Operasi	Windows, Linux, Mac OS GUI-based
2.	Browser	Google Chrome Mozilla Firefox

c) Perangkat Lunak Pengembangan

Dalam proses pengembangan sistem dibutuhkan perangkat lunak dari mulai sistem operasi yang digunakan, *code editor* sampai perangkat pengembangan yang dijelaskan pada tabel 3.15.

Tabel 3.15 Spesifikasi Kebutuhan Perangkat Lunak Untuk Pengembangan

No.	Kategori	Spesifikasi
1.	Sistem Operasi	Windows atau Linux
2.	Perangkat lunak Pengembangan	Visual Studio Code Arduino IDE
3.	Perangkat lunak <i>command-line</i>	NodeJS NPM Git create-react-app
4.	Perancangan Antarmuka / <i>Wireframing Tools</i>	Pencils Project Draw.io

3.1.10.3 Analisis Pengguna

Analisis pengguna merupakan penguraian kebutuhan apa saja yang harus dimiliki oleh pengguna agar dapat menggunakan sistem. Adapun kebutuhan pengguna yang terlibat dalam penggunaan sistem evaluasi kendaraan dinas SKK Migas dapat dilihat pada tabel 3.16.

Tabel 3.16 Tabel Analisis Kebutuhan Pengguna

Pengguna	Hak akses	Pengalaman Pengguna
Web Administrator/Staff	1. Mendaftarkan dan mengelola	1. Dapat mengoperasikan

IT	kendaraan	komputer 2. Dapat mengoperasikan browser 3. Dapat menggunakan aplikasi web
Pimpinan/Direktur	1. Melihat dashbor visualisasi 2. Melihat riwayat kendaraan 3. Mengunduh laporan	1. Dapat mengoperasikan komputer 2. Dapat mengoperasikan browser 3. Dapat menggunakan aplikasi web

3.1.11 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional menggambarkan kebutuhan sistem yang akan dibangun. Adapun kebutuhan fungsional pada sistem yang akan dibangun ini yaitu dengan pemodelan berorientasi objek. Sistem yang akan dibangun dimodelkan menggunakan UML (*Unified Modeling Language*).

Tabel 3.17 Spesifikasi Kebutuhan Perangkat Lunak Fungsional

No	Kode Kebutuhan	Deskripsi Kebutuhan
1.	SKPL-F01	Sistem mewajibkan untuk melakukan login dan hanya dapat diakses oleh aktor yang mempunyai hak akses.
2.	SKPL-F02	Sistem dapat menampilkan dashboard berupa data visual yang menampilkan ringkasan penggunaan kendaraan.
3.	SKPL-F03	Sistem dapat menampilkan riwayat perjalanan kendaraan dan riwayat kejadian mengendara disajikan dalam peta.
4.	SKPL-F04	Sistem dapat menghasilkan laporan ringkasan penggunaan kendaraan dengan pilihan periode dengan format PDF.
5.	SKPL-F05	Sistem dapat merekam perjalanan kendaraan dan kejadian mengendara.

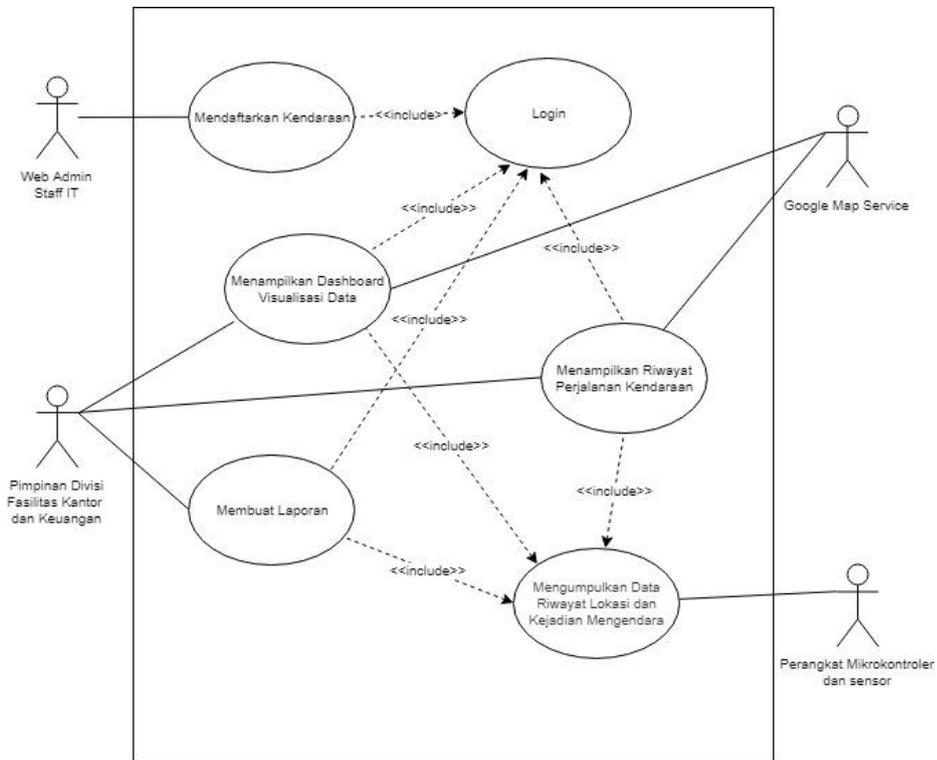
3.1.11.1 Use Case Diagram

Merupakan diagram yang menggambarkan fungsionalitas suatu sistem dan interaksi yang terjadi dengan hal diluar sistem. *Use Case* mendeskripsikan interaksi antara pengguna sistem dengan sistem. Ada 3 bagian *Use Case* diagram yaitu:

1. *Use Case*
2. Deskripsi Aktor
3. Deskripsi *Use Case*

Berikut adalah *Use Case* diagram yang dirancang untuk sistem yang akan dibangun dapat dilihat pada gambar 3.23.

1. Use Case



Gambar 3.23 Use Case Diagram

2. Deskripsi Aktor

Deskripsi aktor dimaksudkan adalah pengguna yang berperan dalam sistem yang dibangun. Aktor disini adalah pihak yang berinteraksi dengan *Use Case*. Daftar aktor yang terlibat beserta deskripsinya dapat dilihat pada tabel 3.18.

Tabel 3.18 Deskripsi Aktor

No.	Aktor	Deskripsi
1	Web Admin Staff IT	Web admin adalah orang yang mempunyai hak akses ke dalam sistem dan mengelola sistem.
2	Pimpinan Divisi Fasilitas Kantor dan Keuangan	Aktor yang dapat melihat hasil dari pengolahan data berbentuk visualisasi dan laporan.
3	Perangkat Mikrokontroler dan Sensor	Aktor yang berperan sebagai sumber dari data yang didapat dan mengirimkannya ke sistem yang selanjutnya disimpan.
4	Google Map Service	Aktor yang berperan sebagai penyedia layanan untuk peta.

3. Deskripsi Use Case

Deskripsi *Use Case* menjelaskan urutan langkah-langkah dalam proses yang dilakukan oleh sebuah *Use Case*. Adapun deskripsi *Use Case* dari sistem evaluasi kendaraan dinas adalah sebagai berikut:

a) Deskripsi Use Case Login

Skenario use case login menguraikan proses saat pengguna melukan login untuk mengakses sistem.

Tabel 3.19 Deskripsi Use Case Login

Nama Use Case	Login	
Persyaratan Terkait	SKPL-F01	
Tujuan	Pengguna dapat masuk kedalam sistem dan menggunakan sistem.	
Prasyarat	Pengguna telah mempunyai akun dan mempunyai hak akses untuk masuk kedalam sistem.	
Kondisi Apabila Sukses	Login Berhasil pengguna Berpindah Halaman Ke Halaman Admin.	
Kondisi Apabila Gagal	Muncul peringatan bahwa login gagal	
Aktor Utama	Pimpinan Divisi Fasilitas Kantor dan Keuangan.	
Aktor Sekunder	Web Admin Staff IT.	
Pemicu	Pengguna mengakses sistem tetapi belum melakukan login.	
Alur Utama	Langkah	Aksi
	1	Pengguna membuka aplikasi
	2	Sistem menampilkan form login
	3	Pengguna memasukkan username dan password.
	4	Sistem verifikasi username dan password
	5	Ssitem menampilkan halaman Dashboard
Ekstensi	Langkah	Aksi
	5.1	Sistem menampilkan pesan login gagal
	5.2	Kembali ke langkah 2

b) Deskripsi Use Case Menampilkan *Dashboard* Visualisasi Data

Tabel 3.20 Deskripsi Use Case Menampilkan *Dashboard* Visualisasi Data

Nama Use Case	Menampilkan Dashboard Visualisasi Data
Persyaratan Terkait	SKPL-F02
Tujuan	Sistem menampilkan seluruh ringkasan informasi penggunaan kendaraan dalam satu halaman dengan visualisasi data interaktif.
Prasyarat	Pengguna telah melakukan login
Kondisi Apabila Berhasil	Sistem menampilkan semua informasi berupa visualisasi data (chart) penggunaan kendaraan

Kondisi Apabila Gagal	Halaman tidak termuat dikarenakan hubungan dengan server terputus	
Aktor Utama	Pimpinan Divisi Fasilitas Kantor dan Keuangan	
Aktor Sekunder	Web Admin Staff IT	
Pemicu	Setelah pengguna login atau pengguna memilih menu <i>dashboard</i> .	
Alur Utama	Langkah	Aksi
	1	Pengguna membuka aplikasi
	2	Pengguna memilih menu dashboard
	3	Sistem menampilkan halaman dashboard
	4	Sistem memuat data dari server
	5	Sistem menampilkan data dalam bentuk grafik

c) Deskripsi *Use Case* Menampilkan Riwayat Perjalanan Kendaraan

Tabel 3.21 Deskripsi *Use Case* Menampilkan Riwayat Perjalanan Kendaraan

Nama Use Case	Menampilkan Riwayat Perjalanan Kendaraan	
Persyaratan Terkait	SKPL-F03	
Tujuan	Sistem menampilkan informasi riwayat perjalanan kendaraan beserta riwayat kejadian mengemudi.	
Prasyarat	Pengguna telah melakukan login	
Kondisi Apabila Berhasil	Sistem menampilkan informasi riwayat perjalanan kendaraan beserta riwayat kejadian mengemudi.	
Kondisi Apabila Gagal	Halaman tidak termuat dikarenakan hubungan dengan server terputus	
Aktor Utama	Pimpinan Divisi Divisi Fasilitas Kantor dan Keuangan	
Aktor Sekunder	Web Admin Staff IT	
Pemicu	Setelah pengguna login atau pengguna memilih menu Kendaraan.	
Alur Utama	Langkah	Aksi
	1	Pengguna membuka aplikasi
	2	Pengguna memilih menu Kendaraan
	3	Sistem menampilkan halaman antamuka riwayat kendaraan
	4	Pengguna memilih kendaraan pada list
	5	Pengguna memilih periode
	6	Sistem memuat data riwayat kendaraan
	7	Sistem menampilkan riwayat kendaraan

d) Deskripsi *Use Case* Membuat Laporan

Tabel 3.22 Deskripsi *Use Case* Membuat Laporan

Nama Use Case	Membuat Laporan
Persyaratan Terkait	SKPL-F04
Tujuan	Sistem menghasilkan laporan berupa berkas PDF
Prasyarat	Pengguna telah melakukan login
Kondisi Apabila	Sistem memproses permintaan pencetakan laporan dan pengguna dapat

Berhasil	mendownload laporan berupa berkas PDF	
Kondisi Apabila Gagal	Proses pencetakan laporan di server gagal dan sistem menampilkan pesan kesalahan	
Aktor Utama	Pimpinan Divisi Fasilitas Kantor dan Keuangan	
Aktor Sekunder	Web Admin Staff IT	
Pemicu	Pengguna memilih untuk mengunduh laporan	
Alur Utama	Langkah	Aksi
	1	Pengguna Membuka Aplikasi
	2	Pengguna memilih menu Laporan
		Sistem menampilkan halaman Laporan
	3	Pengguna Memilih Periode Laporan
	4	Pengguna Klik pada tombol <i>preview</i>
	5	Sistem memuat data laporan
	6	Sistem menampilkan <i>preview</i> laporan
	7	Pengguna klik pada tombol <i>download</i>
	8	Sistem memproses download laporan
9	Pengguna mendownload laporan	

e) Deskripsi *Use Case* Mengumpulkan data Riwayat Lokasi dan Kejadian Mengendara

Tabel 3.23 Deskripsi *Use Case* Mengumpulkan data Riwayat Lokasi dan Kejadian Mengendara

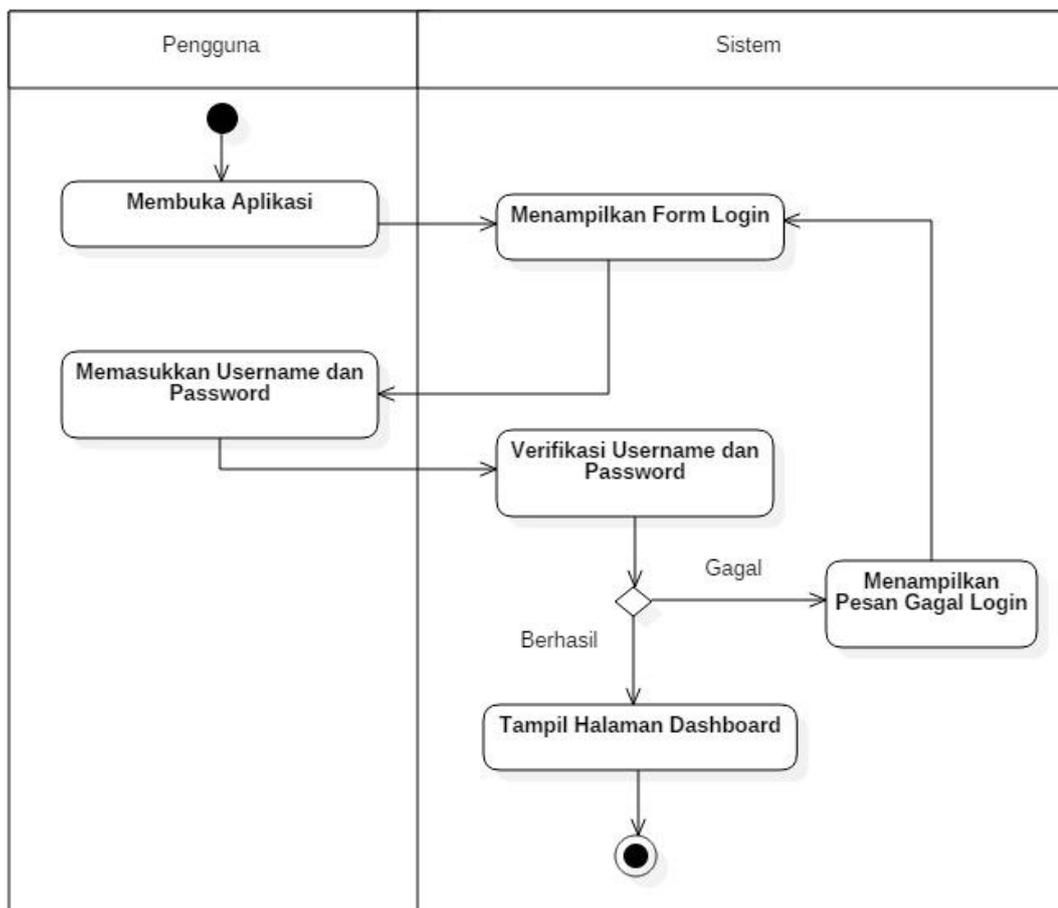
Nama Use Case	Mengumpulkan data Riwayat Lokasi dan Kejadian Mengendara	
Persyaratan Terkait	SKPL-F05	
Tujuan	Sistem mengumpulkan riwayat lokasi dan kejadian mengendara	
Prasyarat	-	
Kondisi Apabila Berhasil	Data riwayat lokasi dan kejadian mengendara terekam di basis data	
Kondisi Apabila Gagal	Data riwayat lokasi dan kejadian mengendara tidak terekam di basis data	
Aktor Utama	Perangkat Mikrokontroler dan Sensor	
Aktor Sekunder	-	
Pemicu	-	
Alur Utama	Langkah	Aksi
	1	Mikrokontroler menghubungkan ke server
	2	Mikrokontroler mendeteksi kejadian mengendara
	3	Mikrokontroler merekam lokasi
	4	Mikrokontroler mengirimkan lokasi dan kejadian mengendara ke server
	5	Sistem menerima data lokasi dan kejadian mengendara
	6	Sistem menyimpan data lokasi dan kejadian mengendara ke basis data

3.1.11.2 Activity Diagram

Activity Diagram merupakan digram yang menggambarkan aliran kerja dari urutan aktifitas pada suatu proses yang mengacu pada *use case diagram*. Berikut *activity diagram* pada sistem ini:

1. Login

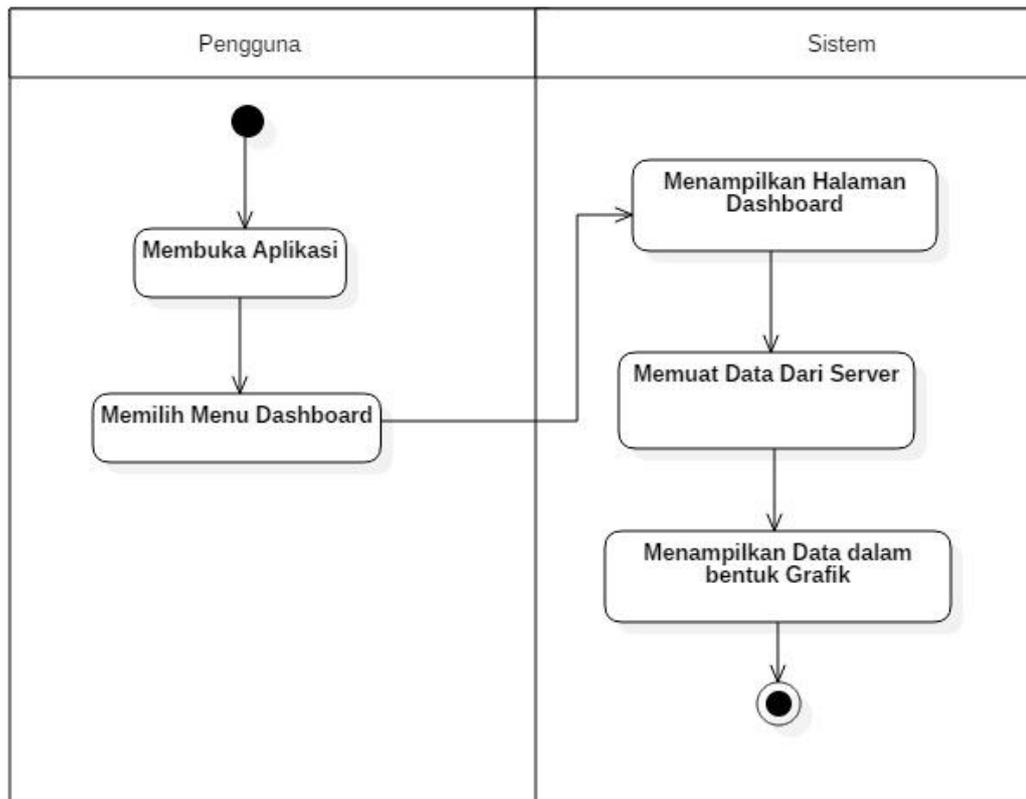
Activity Diagram login menjelaskan aliran kerja aktor pengguna pada saat melakukan login pada sistem. Berikut *activity diagram* login dapat dilihat pada gambar 3.24.



Gambar 3.24 *Activity Diagram* Login

2. Menampilkan Dashboard Visualisasi Data

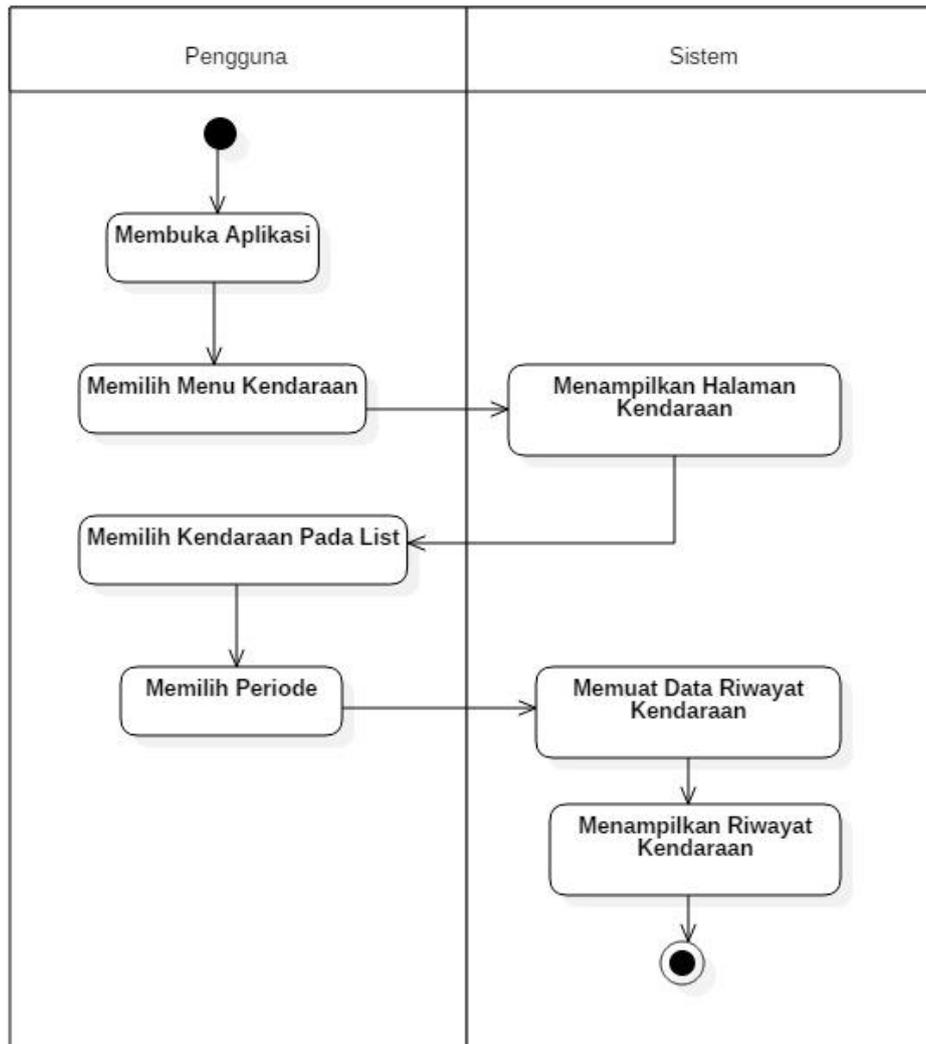
Activity diagram menampilkan dashboard visualisasi data menggambarkan kegiatan pengguna saat berada pada sistem dan menampilkan informasi berupa visualisasi data yang berbentuk grafik.



Gambar 3.25 Activity Diagram Menampilkan Dashboard Visualisasi Data

3. Menampilkan Riwayat Perjalanan Kendaraan

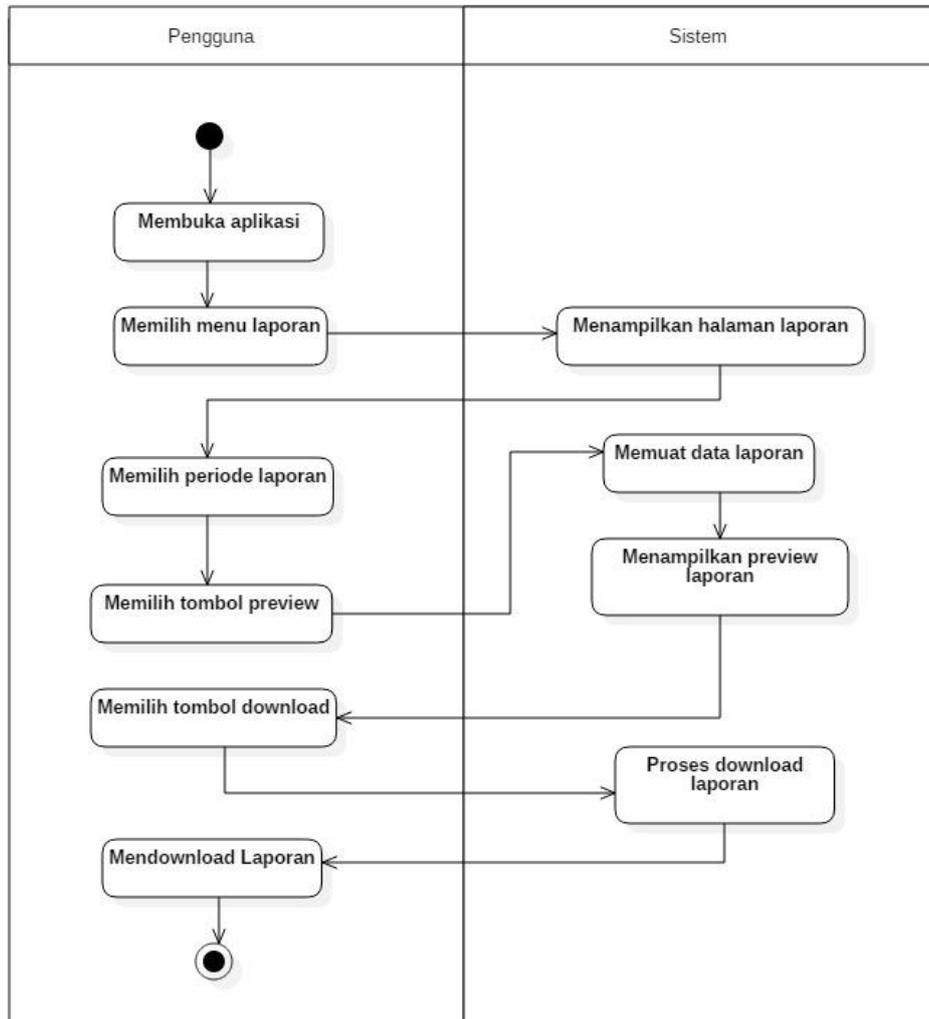
Activity diagram menampilkan riwayat perjalanan kendaraan menggambarkan aliran kerja pengguna pada saat menggunakan sistem dan menampilkan informasi riwayat kendaraan.



Gambar 3.26 Activity Diagram Menampilkan Riwayat Perjalanan Kendaraan

4. Membuat Laporan

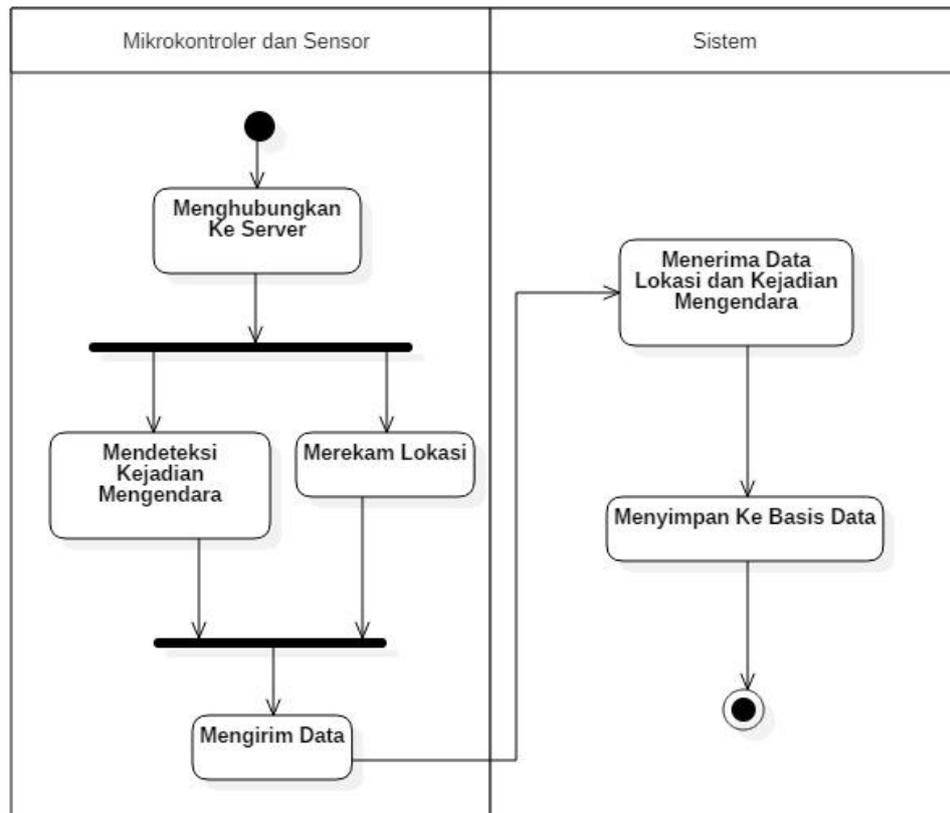
Activity diagram membuat laporan menggambarkan dan menjelaskan aktifitas aktor saat menggunakan sistem dan berinteraksi untuk meminta sistem membuatkan laporan sampai pengguna mendownload laporan tersebut.



Gambar 3.27 Activity Diagram Membuat Laporan

5. Mengumpulkan Data Riwayat Lokasi dan Kejadian Mengendara

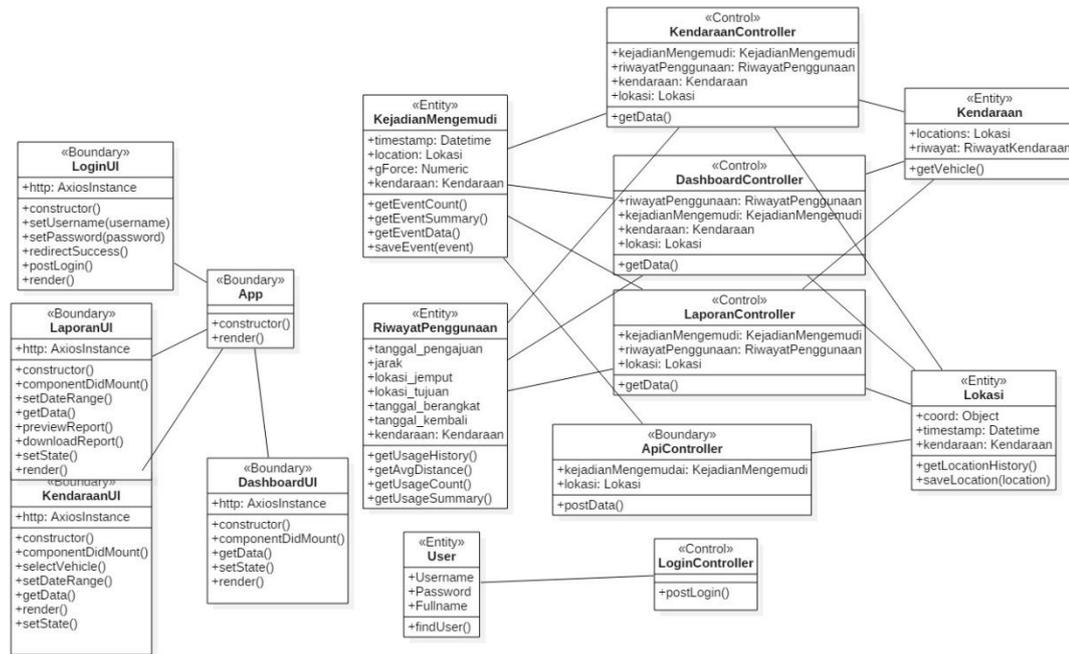
Activity Diagram mengumpulkan data riwayat lokasi dan kejadian mengendara dilakukan oleh mikrokontroler dan sensor berkomunikasi dengan sistem.



Gambar 3.28 Activity Diagram Mengumpulkan Data Riwayat Lokasi dan Kejadian Mengendara

3.1.11.3 Class Diagram

Class Diagram menjelaskan pemodelan hubungan antar *class* yang berisikan nama *class*, *attribute/variable*, *method* dan relasi antar *class*. Pada setiap *class* dijelaskan pula *type* dan sifat *method* itu sendiri. Class Diagram pada sistem yang akan dibangun dapat dilihat pada gambar 3.29.



Gambar 3.29 Class Diagram

Kelas (*class*) pada sistem yang dibangun terdiri dari tiga jenis kelas yaitu *class boundary*, *class control*, dan *class entity*. Yang menjadi *class boundary* yaitu kelas yang berhubungan dengan luar sistem yang menjadi antarmuka pengguna dengan sistem. Sedangkan *class control* yaitu kelas yang mengatur setiap aksi-aksi yang diperintahkan ke sistem maupun dari pengguna (*user*) atau dari perangkat sensor. Adapun *class entity* adalah kelas dimana terjadinya pengolahan data. Pada tabel 3.24 adalah daftar kelas beserta jenisnya.

Tabel 3.24 Daftar Kelas

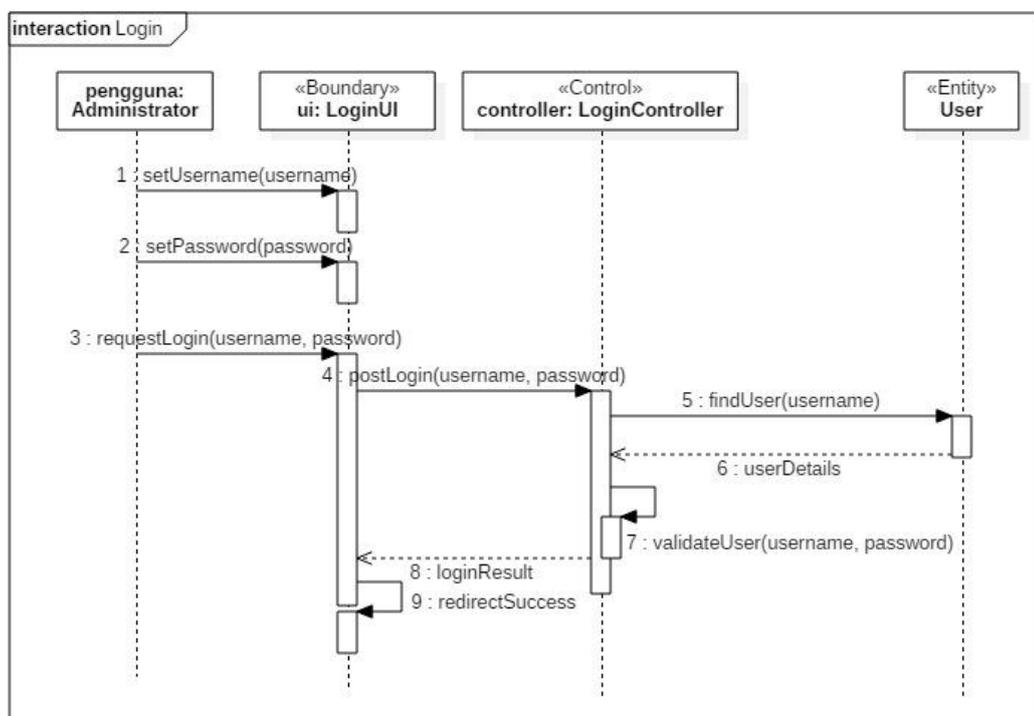
No	Nama Kelas	Jenis Kelas
1.	App	Boundary
2.	DashboardUI	Boundary
3.	KendaraanUI	Boundary
4.	LaporanUI	Boundary
5.	LoginUI	Boundary
6.	LaporanController	Control
7.	DashboardController	Control
8.	KendaraanController	Control
9.	LoginController	Control
10.	ApiController	Control
11.	Kendaraan	Entity
12.	KejadianMengemudi	Entity
13.	RiwayatPenggunaan	Entity

14.	Location	Entity
-----	----------	--------

3.1.11.4 Sequence Diagram

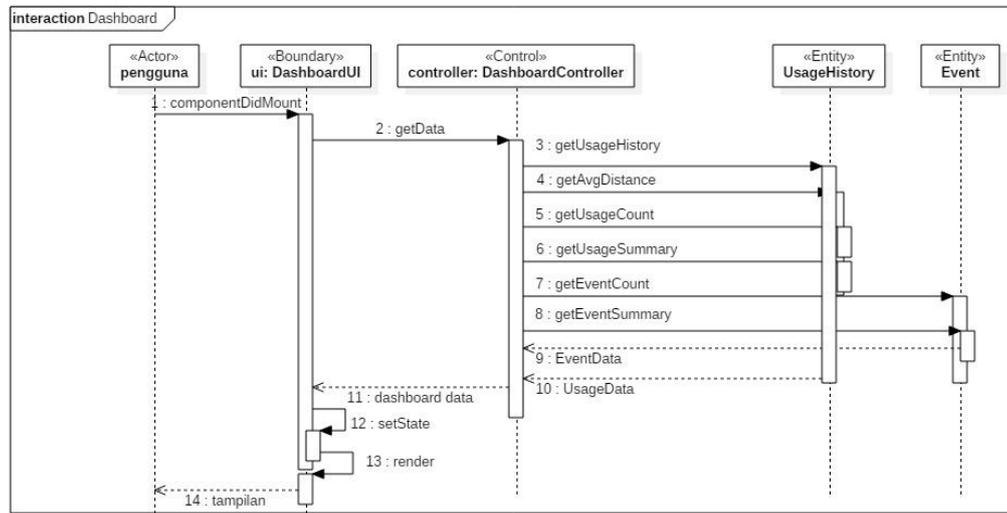
Sequence diagram menggambarkan interaksi antara masing-masing objek pada setiap *use case* dalam urutan waktu. Interaksi ini berupa pengiriman serangkaian data antar objek-objek yang saling berinteraksi.

1. Login



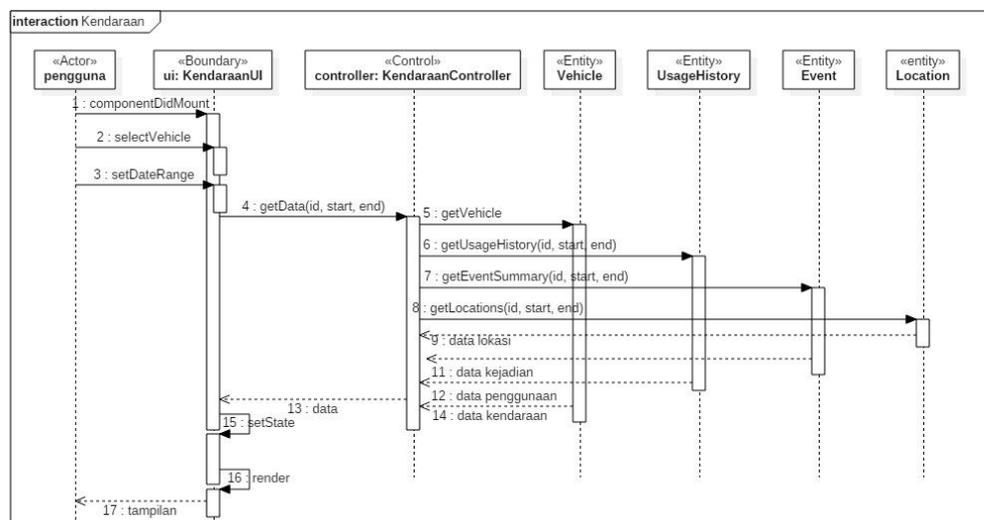
Gambar 3.30 *Sequence Diagram Login*

2. Menampilkan Dashboard Visualisasi Data



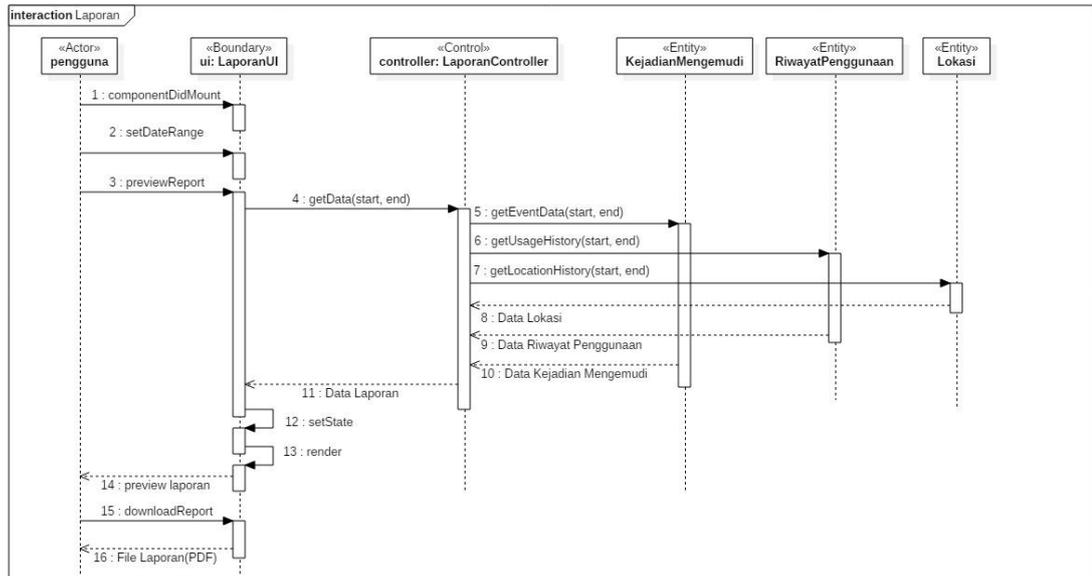
Gambar 3.31 Sequence Diagram Menampilkan Dashboard Visualisasi Data

3. Menampilkan Riwayat Perjalanan Kendaraan



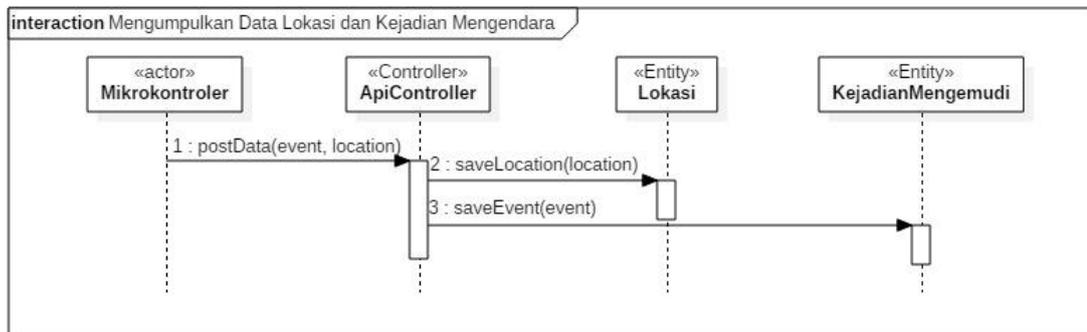
Gambar 3.32 Sequence Diagram Menampilkan Riwayat Perjalanan Kendaraan

4. Menampilkan dan Membuat Laporan



Gambar 3.33 Sequence Diagram Membuat Laporan

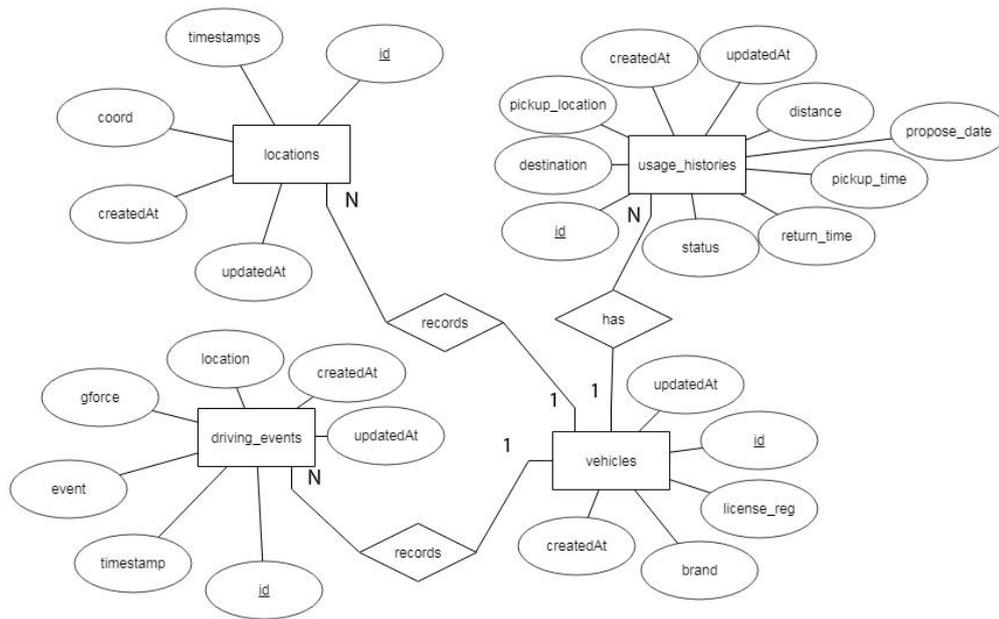
5. Mengumpulkan Riwayat Lokasi dan Kejadian Mengendara



Gambar 3.34 Sequence Diagram Mengumpulkan Riwayat Lokasi dan Kejadian Mengendara

3.1.12 Rancangan ERD (Entity Relationship Diagram)

Berikut adalah rancangan ERD atau *Entity Relationship Diagram* dimana entitas yang terlibat yaitu kendaraan, lokasi, kejadian dan riwayat penggunaan kendaraan. Rancangan ERD dapat dilihat pada gambar 3.35.



Gambar 3.35 Rancangan ERD Pada Sistem

3.1.13 Struktur Tabel

Berikut adalah struktur tabel yang dibuat berdasarkan rancangan ERD.

Tabel 3.25 Struktur Tabel driving_events

Nama Field	Type Data	Ukuran	Keterangan
id	Integer	11	Primary Key, Auto Increment
timestamp	Timestampz		
event	Enum	-	(“HARD_ACCELERATION”, “HARD_BRAKING”, “HARD_CORNERING”)
gforce	Numeric	255	
location	Geometry(POINT, 4326)		
vehicleId	Integer	11	Foreign Key
createdAt	Timestampz	-	
updatedAt	Timestampz	-	

Tabel 3.26 Struktur Tabel locations

Nama Field	Type Data	Ukuran	Keterangan
id	Integer	11	Primary Key, Auto Increment
timestamp	Timestampz	-	
coord	Geometry(POINT, 4326)	-	
vehicleId	Integer	11	Foreign Key
createdAt	Timestampz	-	

updatedAt	Timestampz	-	
-----------	------------	---	--

Tabel 3.27 Struktur Tabel vehicles

Nama Field	Tipe Data	Ukuran	Keterangan
id	Integer	11	Primary Key, Auto Increment
license_reg	Varchar	255	
brand	Varchar	255	
createdAt	Timestampz	-	
updatedAt	Timestampz	-	

Tabel 3.28 Struktur Tabel usage_histories

Nama Field	Tipe Data	Ukuran	Keterangan
id	Integer	11	Primary Key, Auto Increment
propose_date	Timestampz	-	
pickup_time	Timestampz	-	
return_time	Timestampz		
distance	Numeric	255	
pickup_location	Geometry(Point, 4326)	-	
destination	Geometry(Point, 4326)	-	
status	Enum		("Selesai", "Ditolak", "Dibatalkan")
vehicleId	Integer	11	Foreign Key
createdAt	Timestampz	-	
updatedAt	Timestampz	-	

3.2 Perancangan Sistem

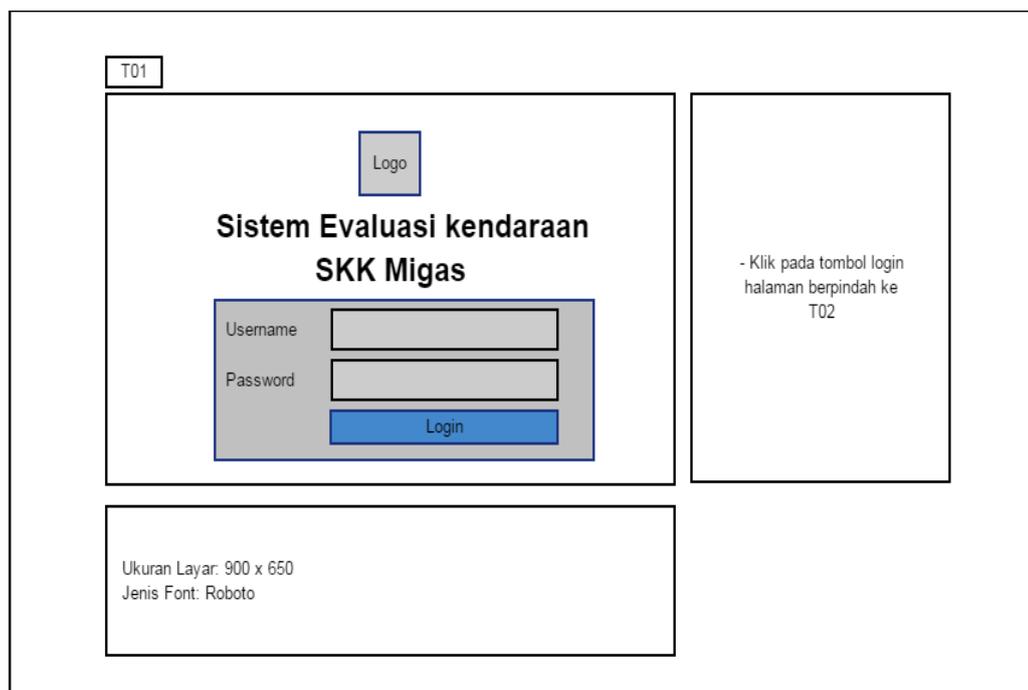
Perancangan sistem merupakan bagian dari metodologi pengembangan perangkat lunak yang dilakukan untuk memberikan gambaran secara terperinci. Berdasarkan penguraian yang sudah dilakukan sebelumnya perancangan sistem dibuat berdasarkan kebutuhan-kebutuhan fungsional dan menjadi landasan untuk implementasi dan pembangunan sistem yang menggambarkan bagaimana sistem dibentuk yang dapat berupa gambaran, perancangan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam suatu kesatuan yang utuh dan berfungsi.

3.2.1 Perancangan Antarmuka

Antar muka atau *interface* merupakan tampilan dari suatu perangkat lunak yang berperan sebagai media interaksi anatar manusia dan komputer. Perangkat lunak yang dibangun diharapkan menyediakan antarmuka yang mudah dipahami dan digunakan oleh pengguna.

3.2.1.1 Antarmuka Halaman Login

Antarmuka login menampilkan form untuk melakukan login dengan isian *username* dan *password*. Rancangan antarmuka login dapat dilihat pada gambar 3.36.



Gambar 3.36 Rancangan Antarmuka Halaman Login

3.2.1.2 Antarmuka Halaman Dashboard

Antarmuka halaman dashboard menampilkan informasi secara visual dengan menggunakan *Chart*. Pada bagian atas dashboard menampilkan informasi jumlah kesalahan mengemudi, jarak tempuh yang dihitung dari lokasi-lokasi yang didapat GPS, rata-rata jarak tujuan yang dihitung dari jarak setiap pengajuan dan jumlah pengajuan. Selanjutnya menampilkan grafik untuk pengajuan yang menampilkan perbandingan pengajuan yang ditolak, disetujui, dibatalkan dan selesai. Kejadian mengendara ditampilkan dalam grafik bar-chart secara horizontal yang dikelompokkan berdasarkan kendaraan dengan bentuk *stacked*. Top peminjam menampilkan siapa saja 10 peminjam atau pemakai kendaraan diurutkan dari peminjam yang sering melakukan peminjaman. Tujuan perjalanan menampilkan peta dengan menampilkan *heatmap* dan *marker-cluster* untuk

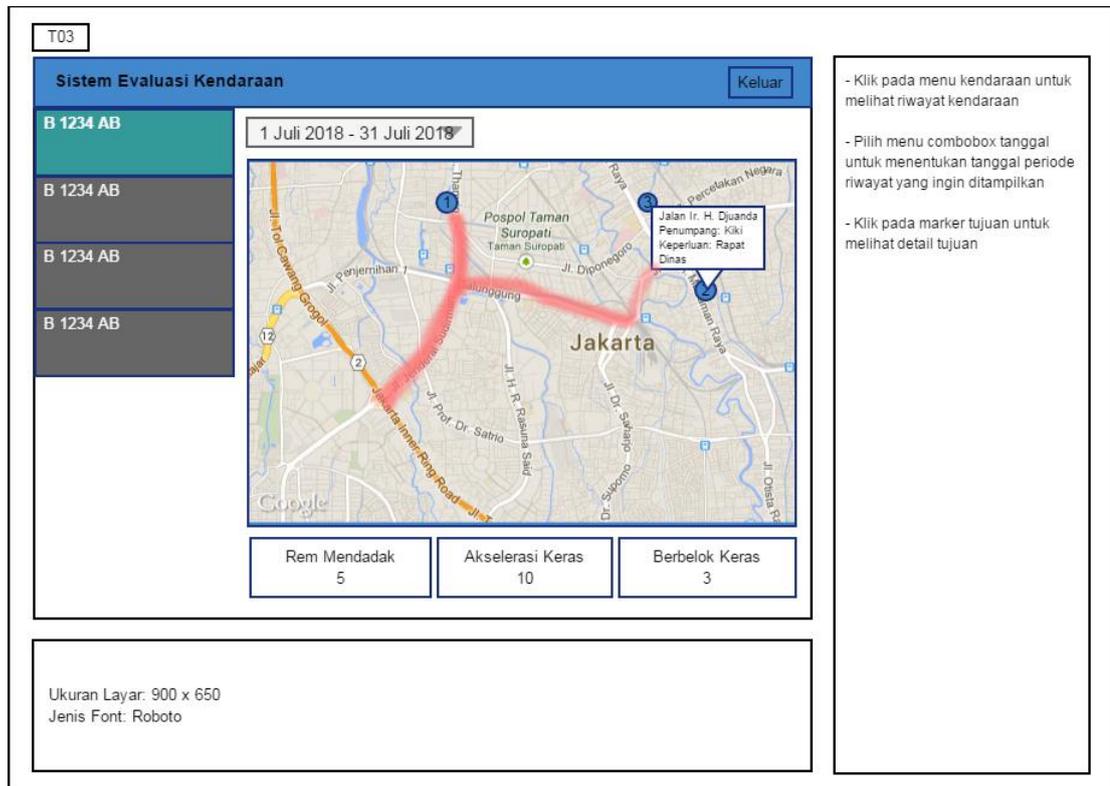
menyajikan informasi berdasarkan daerah mana yang sering dikunjungi peminjam. Rancangan antarmuka Dashboard dapat dilihat pada gambar 3.37.



Gambar 3.37 Rancangan Antarmuka Halaman Dashboard

3.2.1.3 Antarmuka Halaman Kendaraan

Antarmuka halaman kendaraan menampilkan list kendaraan, pilihan periode tanggal, peta dan informasi kejadian mengemudi. Item pada list kendaraan dapat dipilih dengan cara di-klik untuk menampilkan riwayat kendaraan tersebut. Pilihan periode tanggal dapat di klik untuk memilih rentang tanggal riwayat kendaraan yang ingin ditampilkan. Peta menampilkan marker titik-titik lokasi tujuan penggunaan kendaraan berdasarkan periode yang dipilih. Dengan klik pada marker akan muncul informasi detail nya. Peta juga menampilkan riwayat perjalanan jalur mana saja yang dilalui kendaraan dengan visualisasi *heatmap*. Pada bagian bawah peta terdapat informasi kejadian mengendara berdasarkan periode dipilih. Rancangan antarmuka halaman kendaraan dapat dilihat pada gambar 3.38.



Gambar 3.38 Rancangan Antarmuka Halaman Kendaraan

3.2.1.4 Antarmuka Halaman Laporan

Antarmuka halaman laporan menampilkan pilihan periode laporan yang akan dibuat dan menampilkan *preview* dari laporan berdasarkan periode yang terpilih. Setelah selesai memilih periode dapat memilih tombol *download* untuk mengunduh laporan. Rancangan antarmuka laporan dapat dilihat pada gambar 3.39.



Gambar 3.39 Rancangan Antarmuka Halaman Laporan

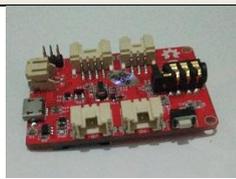
3.2.2 Perancangan Perangkat Mikrokontroler dan Sensor

Perancangan meliputi komponen-komponen yang digunakan dan perancangan detail berupa diagram skematik.

3.2.2.1 Komponen Yang Digunakan

Berikut pada tabel 3.29 komponen-komponen yang digunakan pada perangkat mikrokontroler dan sensor yang dibutuhkan sistem.

Tabel 3.29 Komponen Perancangan Perangkat Mikrokontroler dan Sensor

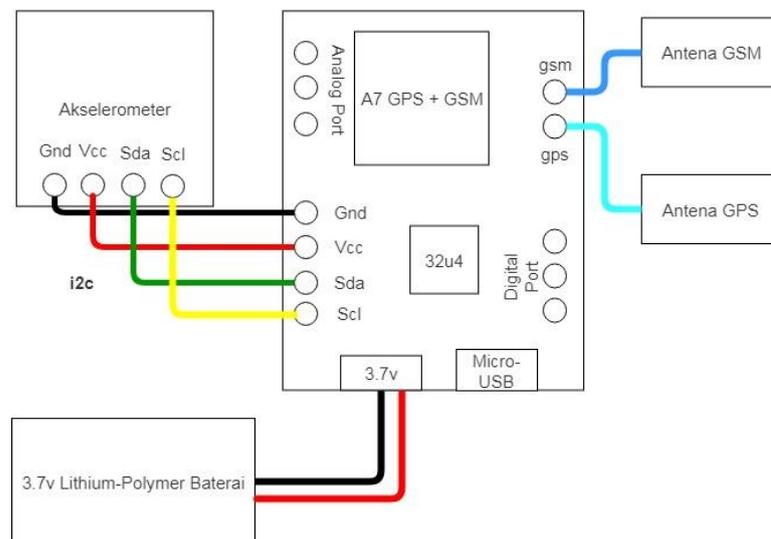
Perangkat	Gambar	Nama	Spesifikasi
Mikrokontroler		Elecrow 32u4 With A7 GSM/GPS	Fitur: GPS, GSM dan GPRS. Daya: 3.7V sampai 5V Penyimpanan: Flash 28Kb
Antena GSM		SIM800L GSM/GPRS Antenna	Konektor: RP-SMA

Antena GPS		Active GPS Antenna A7 External untuk Mobil	Konektor: RP-SMA Frequency: 1575.42MHz Voltage: 3-5V DC current: Max:10mA Polarization: Circular (RH) Waterproof: YES Panjang Kabel: 3M Dimensi: 45x39x13mm (main unit) Berat: 60g
Baterai		TCL Li-Polymer Battery	Kapasitas: 820Mah Voltage: 3.7v Konektor: JST
Konektor Antena		RP-SMA to U.FL Connector	Panjang Kabel: 10cm
Kabel Modul		4 Pin Crowtail Cable	4 Pin (i2c) Type: JST-PH I2C
Kartu SIM 3G		Kartu Telkomsel HALO	APN:Internet Ukuran: Micro-SIM
Akselerometer		Crowtail MPU6050 Accel+Gyro	Range Sensitifitas: +- 16g Konektor: i2c
Casing		Custom	Bahan: Cardboard / Dus Panjang: 10.0cm Lebar: 6.3cm

			Tinggi: 2.5cm
Kabel data dan Power		Kabel Micro-USB Standar	Panjang: 100cm

3.2.2.2 Diagram Blok Alat

Pada gambar 3.40 menampilkan rancangan mikrokontroler dan sensor dengan bentuk diagram blok. Sensor akselerometer dihubungkan dengan sensor mikrokontroler menggunakan kabel komunikasi i2c. Perangkat mikrokontroler diberi daya baterai 3.7v dan dipasangkan antena GSM dan GPS.



Gambar 3.40 Diagram Blok Alat