

## BAB 4

### IMPLEMENTASI DAN PENGUJIAN

#### 4.1 Implementasi Sistem

Pada tahapan ini, dilakukan implementasi Sistem Evaluasi Kendaraan Dinas SKK Migas dari hasil analisis dan perancangan yang telah dilakukan. Implementasi meliputi perangkat pengembang, perangkat server, perangkat *web client*, perangkat mikrokontroler dan sensor, basis data, *class* dan antarmuka. Pada tahap implementasi “Pembangunan Sistem Evaluasi Kendaraan Dinas SKK Migas Berbasis *Internet of Things IoT*” ini, semua komponen pada sistem sangat penting. Sistem dapat berjalan dengan baik apabila semua komponen memenuhi standar minimal dari segi perangkat keras (*hardware*), perangkat lunak (*software*) serta jaringan komunikasi data.

##### 4.1.1 Perangkat Keras Pengembang

Perangkat keras pengembang yang penulis gunakan untuk mengembangkan sistem yaitu sebuah laptop dengan spesifikasi pada tabel 4.1.

**Tabel 4.1 Spesifikasi Perangkat Keras Pengembang**

No	Spesifikasi	Versi
1.	Tipe	Asus X450JF
2.	Prosesor	Intel® Core™ i7 4700HQ Processor,
3.	Memori	8GB DDR3
4.	Kartu Grafis	NVIDIA® GeForce® GT 745M 2GB DDR3
5.	Hard disk	SSD 256GB

##### 4.1.1.1 Sistem Operasi

Sistem operasi yang digunakan penulis pada perangkat keras (laptop) pengembang yaitu seperti pada tabel 4.2.

**Tabel 4.2 Sistem Operasi Pengembang**

No	Spesifikasi	Versi
1.	Sistem Operasi	Linux Mint 18.3 Cinnamon 64-bit
2.	Linux Kernel	v4.13
3.	Arsitektur	64-bit

#### 4.1.1.2 Perangkat Lunak Pengembang

Berikut adalah perangkat lunak yang digunakan untuk pengembangan dan implementasi sistem. Perangkat lunak berupa GUI (*Graphical User Interface*) dan CLI (*Command-Line Interface*). Daftar perangkat lunak dapat dilihat pada tabel 4.3.

**Tabel 4.3 Perangkat Lunak Pengembangan**

No	Perangkat Lunak	Versi	Tipe	Keterangan
1.	Arduino IDE	v1.8.4	GUI	Untuk memprogram mikrokontroler dan sensor
2.	Visual Studio Code	v1.25.1	GUI	Untuk memprogram perangkat lunak <i>server</i> dan <i>client</i> berbasis web
3.	Mosquitto-client	v1.4.8	CLI	<i>Tools</i> Command-line Untuk melakukan pengecekan Komunikasi Data MQTT
4.	Node-cli	v8.11.3	CLI	Untuk menjalankan dan melakukan debug program berbasis NodeJS
5.	NPM (Node Package Manager)	v6.1.0	CLI	Sebagai sumber pustaka-pustaka Javascript dan perangkat/ <i>tools</i> untuk memprogram Javascript
6.	SQLElectron	v1.29.0	GUI	<i>Tools</i> untuk mengelola basis data
7.	Browser Google Chrome	v60.0.2	GUI	Untuk <i>debugging</i> antarmuka dan perangkat lunak <i>client</i>

#### 4.1.2 Implementasi Perangkat Server

Perangkat yang dibutuhkan pada server diantaranya perangkat keras, sistem operasi, perangkat lunak, dan koneksi Internet. Keseluruhan perangkat ini dapat mensimulasikan server aslinya sehingga ketika implementasi di tempat studi kasus (SKK Migas) dengan perangkat keras dan sistem operasi berbasis Linux dapat dilakukan dengan mudah.

##### 4.1.2.1 Perangkat Keras Server

Perangkat keras yang digunakan untuk implementasi *server* yaitu sebuah Raspberry PI versi 3B. Raspberry PI ini cukup mampu untuk dijadikan sebuah *server* berbasis Linux. Berikut spesifikasi lebih lengkap untuk server dapat dilihat pada tabel 4.4.

**Tabel 4.4 Spesifikasi Perangkat Keras Server**

No	Spesifikasi	Keterangan/Versi
1.	Nama Perangkat	Raspberry PI 3B
2.	CPU	BCM2837 Quad Cortex A53 @ 1.2GHz
3.	Arsitektur	ARMv8-A
4.	RAM	1GB SDRAM

5.	Media Penyimpanan	Sandisk MicroSD 32GB Class 10
6.	Konektifitas	Ethernet 10/100 Mbps
7.	Arsitektur	ARMv8-A

#### 4.1.2.2 Sistem Operasi Server

Sistem operasi yang digunakan pada Raspberry Pi 3B yaitu Raspbian yang berbasis pada Debian. Karena digunakan untuk server maka Raspbian yang digunakan adalah yang versi Lite yaitu versi tanpa GUI (Graphical User Interface). Dengan begitu server akan lebih ringan dan tidak memakan banyak media penyimpanan. Berikut spesifikasi sistem operasi yang digunakan untuk server pada tabel 4.5.

**Tabel 4.5 Spesifikasi Sistem Operasi Server**

No	Spesifikasi	Keterangan
1.	Nama Sistem Operasi	Raspbian Stretch Lite
2.	Versi	Juni 2018
3.	Tanggal Rilis	2018-06-27
4.	Arsitektur	Armv7l
5.	Kernel	Linux Raspberrypi 4.14

#### 4.1.2.3 Perangkat Lunak Server

Berikut adalah perangkat lunak yang dipasang di server guna berjalannya sistem. Perangkat lunak berikut tersedia dan kompatibel dengan arsitektur sistem operasi server yaitu Armv7l.

**Tabel 4.6 Spesifikasi Perangkat Lunak Server**

No	Perangkat Lunak	Versi	Arsitektur
1.	NodeJS	v8.11.3	Armv7l
2.	Mosquitto	v3.1	Armv7l
3.	PostgreSQL	v9.4	Armv7l
4.	PostGIS	v2.1	Armv7l

#### 4.1.3 Internet Server

Server memerlukan sambungan ke Internet agar dapat berkomunikasi dengan perangkat sensor. Perangkat sensor berkomunikasi dengan server dengan menggunakan IP yang dapat diakses secara *public*. Untuk mensimulasikan sambungan Internet ini, penulis menggunakan Internet yang telah terpasang di rumah. Perangkat Server (Raspberry PI) disambungkan menggunakan kabel

*Ethernet* langsung ke router. Di router dilakukan konfigurasi *port forwarding* untuk mengarahkan IP ke Server Raspberry PI.

**Tabel 4.7 Spesifikasi Internet Server**

No	Kategori	Spesifikasi
1.	ISP	Indihome Triple Play
2.	Tipe	Modem DSL/Fiber Optik
3.	Kecepatan Download	10 Mbps
4.	Kecepatan Upload	10 Mbps
5.	IP Static/Dynamic	Dynamic
6.	Mode	PPPoE

Dikarenakan IP yang disediakan ISP adalah *public dynamic* maka IP yang didapat dapat berubah-ubah. Jadi, jika sambungan menggunakan IP ke server akan sering terputus jika terjadi perubahan dan harus merubah alamat IP server pada perangkat mikrokontroler dan sensor setiap waktu secara manual. Untuk mengatasinya penulis menggunakan layanan *dynamic dns* dari No-IP dimana menggunakan nama domain untuk melakukan koneksi dengan *server* dengan IP yang terus diperbaharui secara otomatis.

#### 4.1.4 Implementasi Web Client

Perangkat keras sebagai client yang digunakan yaitu sebuah komputer laptop yang sama digunakan sebagai perangkat pengembangan dan dengan menggunakan aplikasi VirtualBox untuk menjalankan sistem operasi Windows yang digunakan sebagai implementasi menggunakan menggunakan perangkat lunak berbasis Windows. Adapun perangkat lunak yang digunakan sebagai perangkat *client* dilihat pada tabel 4.8 dan tabel 4.9.

**Tabel 4.8 Perangkat Lunak Client (Linux)**

No	Perangkat Lunak	Versi	Arsitektur
1.	Browser Chrome	v67.0	Linux 64-bit
2.	Browser Mozilla Firefox	v60.0.2	Linux 64-bit

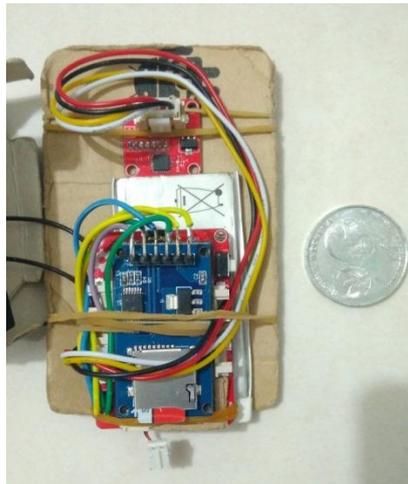
**Tabel 4.9 Perangkat Lunak Client (Windows)**

No	Perangkat Lunak	Versi	Arsitektur
1.	Browser Chrome	v67.0	Windows 32-bit

2.	Browser Internet Explorer	v8.0	Windows 32-bit
----	---------------------------	------	----------------

#### 4.1.5 Implementasi Mikrokontroler dan Sensor

Berikut adalah implementasi perangkat keras mikrokontroler dan sensor yang digunakan pada sistem. Dapat dilihat pada Gambar 4.1 dan Gambar 4.2 hasil implementasi dari perangkat-perangkat yang digunakan.



**Gambar 4.1 Implementasi Perangkat Mikrokontroler dan Sensor**



**Gambar 4.2 Implementasi Casing Perangkat Mikrokontroler**

#### 4.1.6 Implementasi Database

Pada tahap ini hasil dari perancangan *database* akan diimplementasikan secara langsung pada sistem yang dibangun. Karena DBMS (*Database Management System*) yang digunakan adalah PostgreSQL dan menggunakan

PostGIS sebagai ekstensi untuk menyimpan data spasial (dalam kasus ini adalah data lokasi GPS) maka pertama-tama ekstensi PostGIS haruslah diaktifkan pada *database* yang dibuat. Nama database yang dibuat sebagai implementasi yaitu “*vmsiot*” dan mengaktifkan ekstensi PostGIS dengan perintah SQL `CREATE EXTENSION POSTGIS`. Dengan begitu database yang dibuat mendukung tipe data spasial (Geometry).

#### 4.1.6.1 Implementasi Tabel

Berikut adalah tabel-tabel yang dibuat hasil dari implementasi dapat dilihat pada tabel 4.10.

**Tabel 4.10 Implementasi Tabel**

No	Nama Tabel	Definisi Tabel
1.	driving_events	<pre>CREATE TABLE public.driving_events (   id integer NOT NULL,   "timestamp" timestamp with time zone,   event public.enum_driving_events_event,   gforce numeric,   location public.geometry(Point,4326),   "createdAt" timestamp with time zone NOT NULL,   "updatedAt" timestamp with time zone NOT NULL,   "vehicleId" integer,   FOREIGN KEY(vehicleId) REFERENCES vehicles(id) );</pre>
2.	vehicles	<pre>CREATE TABLE public.vehicles (   id integer NOT NULL,   license_reg character varying(255),   brand character varying(255),   "createdAt" timestamp with time zone NOT NULL,   "updatedAt" timestamp with time zone NOT NULL );</pre>
3.	locations	<pre>CREATE TABLE public.locations (   id integer NOT NULL,   "timestamp" timestamp with time zone,   coord public.geometry(Point,4326),   "createdAt" timestamp with time zone NOT NULL,   "updatedAt" timestamp with time zone NOT NULL,   "vehicleId" integer   FOREIGN KEY(vehicleId) REFERENCES vehicles(id) );</pre>
4.	usage_histories	<pre>CREATE TABLE public.usage_histories (   id integer NOT NULL,   propose_date timestamp with time zone,   pickup_time timestamp with time zone,   return_time timestamp with time zone,   distance numeric,   pickup_location public.geometry(Point,4326),   destination public.geometry(Point,4326),   status public.enum_usage_histories_status,</pre>

		<pre>"createdAt" timestamp with time zone NOT NULL, "updatedAt" timestamp with time zone NOT NULL, "vehicleId" integer, FOREIGN KEY(vehicleId) REFERENCES vehicles(id) );</pre>
--	--	---

#### 4.1.6.2 Skema Database

Berikut adalah skema *database* setelah tabel dibuat dilihat pada gambar 4.3. Terdapat tabel yang otomatis terbuat ketika mengaktifkan ekstensi PostGIS pada database yaitu *spatial\_ref\_sys*. Tabel *spatial\_ref\_sys* ini menyimpan informasi spasial yang digunakan oleh ekstensi PostGIS.



**Gambar 4.3 Implementasi Skema Database**

#### 4.1.7 Implementasi Class

Implementasi *Class* pada sistem yang telah dirancang merupakan implementasi yang terbentuk dari *class diagram* pada analisis sistem. Bahasa pemrograman yang digunakan yakni Javascript untuk antarmuka maupun *server*. Ekstensi file kelas yang dibuat adalah *.js*. Dapat dilihat pada tabel 4.11 implementasi *class* dan kode sumber dilampirkan di bagian lampiran.

**Tabel 4.11 Implementasi Class**

No	Nama Kelas	Nama File Fisik	Jenis Kelas
1.	App	App.js	Boundary
2.	DashboardUI	DashboardUI.js	Boundary
3.	KendaraanUI	KendaraanUI.js	Boundary
4.	LoginUI	LoginUI.js	Boundary
5.	LaporanUI	LaporanUI.js	Boundary

6.	DashboardController	dashboard.controller.js	Control
7.	LoginController	login.controller.js	Control
8.	KendaraanController	kendaraan.controller.js	Control
9.	ApiController	api.controller.js	Control
10.	LaporanController	laporan.controller.js	Control
11.	Kendaraan	kendaraan.service.js	Entity
12.	KejadianMengemudi	kejadian.service.js	Entity
13.	RiwayatPenggunaan	riwayat.service.js	Entity
14.	Location	location.service.js	Entity

#### 4.1.8 Implementasi Antarmuka

Berikut ini adalah implementasi antarmuka yang telah dirancang pada Sistem Evaluasi Kendaraan Dinas SKK Migas.

**Tabel 4.12 Implementasi Antarmuka**

No	Nama Antarmuka	Deskripsi	Nama File
1.	Login	Menampilkan kolom isian <i>username</i> dan <i>password</i> yang digunakan untuk masuk kedalam aplikasi.	LoginUI.js
2.	Dashboard	Menampilkan ringkasan informasi yang disajikan dalam bentuk grafis.	DashboardUI.js
3.	Kendaraan	Menampilkan riwayat kendaraan dalam bentuk visualisasi peta.	KendaraanUI.js
4.	Laporan	Menampilkan form untuk membuat laporan berdasarkan periode.	LaporanUI.js

## 4.2 Pengujian

Pengujian Sistem Evaluasi Kendaraan Dinas SKK Migas ini akan dilakukan dengan metode pengujian *Black Box*. Proses pengujian akan dilakukan terhadap semua kebutuhan fungsional yang telah dirancang pada tahap analisis dan perancangan sistem.

### 4.2.1 Pengujian Alpha Aplikasi Web

Pengujian *alpha* aplikasi web bertujuan untuk mengetahui fungsionalitas keseluruhan aplikasi web. Pada tabel 4.13 menjelaskan rencana pengujian aplikasi Web.

**Tabel 4.13 Rencana Pengujian Aplikasi Web**

Item Uji	Detail Uji	Jenis Pengujian
Login	Verifikasi Username dan Password	<i>Blackbox</i>
Dashboard	Menampilkan Ringkasan Informasi Grafis	<i>Blackbox</i>
Kendaraan	Menampilkan Riwayat Kendaraan	<i>Blackbox</i>
	Memilih Kendaraan	<i>Blackbox</i>
	Memilih Periode	<i>Blackbox</i>
Laporan	Menampilkan Formulir	<i>Blackbox</i>
	Memilih Periode	<i>Blackbox</i>
	Menampilkan Pratinjau Laporan	<i>Blackbox</i>
	Mendownload Laporan	<i>Blackbox</i>

Berdasarkan rencana pengujian maka dilakukan pengujian pada aplikasi web.

#### 4.2.1.1 Pengujian Login

Pengujian login dilakukan untuk mengecek fungsionalitas login pada aplikasi web. Pengujian login dengan memasukkan *username* dan *password* pada halaman.

**Tabel 4.14 Pengujian Login**

Kasus dan Hasil Uji (Kondisi Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengetikkan <i>Username</i> “admin” dan <i>password</i> “admin”	Isian <i>username</i> terisi dan terlihat, Isian <i>password</i> terisi tetapi disensor	Isian “admin” tampil pada kotak isian <i>username</i> dan “*****” tampil pada kotak isian <i>password</i>	[ √ ] Diterima [ ] Ditolak
Klik tombol login	Aplikasi langsung mengarahkan ke halaman <i>dashboard</i>	Aplikasi mengarahkan ke halaman dashboard	[ √ ] Diterima [ ] Ditolak

**Tabel 4.15 Pengujian Login Kondisi Salah**

<b>Kasus dan Hasil Uji (Kondisi Salah)</b>			
<b>Data Masukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Kesimpulan</b>
Mengetikkan <i>Username</i> “testing” dan <i>password</i> “testing”	Isian <i>username</i> terisi dan terlihat, Isian <i>password</i> terisi tetapi disensor.	Isian “admin” tampil pada kotak isian <i>username</i> dan “*****” tampil pada kotak isian <i>password</i>	[ √ ] Diterima [ ] Ditolak
Klik tombol login	Muncul pesan gagal login.	Aplikasi memunculkan notifikasi bahwa <i>username</i> atau <i>password</i> salah	[ √ ] Diterima [ ] Ditolak

#### 4.2.1.2 Pengujian Halaman Dashboard

Pengujian dashboard dilakukan untuk mengetahui fungsionalitas halaman dashboard.

**Tabel 4.16 Pengujian Halaman Dashboard**

<b>Kasus dan Hasil Uji (Kondisi Normal)</b>			
<b>Data Masukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Kesimpulan</b>
Memilih menu Dashboard	Halaman Dashboard tampil dan semua grafik muncul beserta data nya	Halaman Dashboard tampil beserta data termuat dalam bentuk grafik	[ √ ] Diterima [ ] Ditolak

#### 4.2.1.3 Pengujian Halaman Kendaraan

Pengujian halaman kendaraan dilakukan untuk mengetahui fungsionalitas halaman kendaraan.

**Tabel 4.17 Pengujian Halaman Kendaraan**

<b>Kasus dan Hasil Uji (Kondisi Normal)</b>			
<b>Data Masukkan</b>	<b>Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Kesimpulan</b>
Memilih menu	Halaman kendaraan	Tampil riwayat	[ √ ] Diterima

Kendaraan	tampil	kendaraan pada peta	[ ] Ditolak
Klik pada salah satu pilihan kendaraan	Riwayat kendaraan yang tampil pada peta berubah	Peta memuat ulang data dan menampilkannya sesuai dengan kendaraan yang terpilih	[ √ ] Diterima [ ] Ditolak
Memilih periode	Riwayat kendaraan yang tampil pada peta berubah	Peta memuat ulang dan menampilkan sesuai dengan periode yang dipilih	[ √ ] Diterima [ ] Ditolak

#### 4.2.1.4 Pengujian Halaman Laporan

Pengujian halaman laporan bertujuan untuk mengetahui fungsionalitas halaman laporan dari memilih periode laporan sampai *browser* mendownload laporan. Hasil pengujian dapat dilihat pada tabel 4.18.

**Tabel 4.18 Pengujian Halaman Laporan**

Kasus dan Hasil Uji (Kondisi Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Memilih menu laporan	Halaman laporan tampil	Tampil halaman laporan beserta form untuk memilih periode dan tombol download	[ √ ] Diterima [ ] Ditolak
Memilih periode laporan	Periode terpilih	Pilihan periode laporan menampilkan periode terpilih	[ √ ] Diterima [ ] Ditolak
Klik tombol <i>preview</i>	Tampil draft laporan	Draft laporan tampil dibawah form	[ √ ] Diterima [ ] Ditolak
Klik tombol <i>download</i>	Browser mendownload	Browser langsung mendownload laporan.	[ √ ] Diterima [ ] Ditolak

#### 4.2.2 Pengujian Alpha Perangkat Mikrokontroler dan Sensor

Pengujian *Alpha* mikrokontroler dan sensor meliputi pengujian fungsionalitas apakah komponen mikrokontroler dan sensor sudah berfungsi sesuai dengan apa yang diharapkan dan dibutuhkan oleh sistem.

#### 4.2.2.1 Pengujian Modul GSM

Pengujian modul GSM dimaksudkan untuk mengetahui apakah modul GSM berfungsi sesuai dengan fungsionalitas yang dibutuhkan sistem yaitu sebagai jembatan komunikasi antara perangkat sensor dan *server*. Hasil pengujian modul GSM untuk dapat terkoneksi ke *server* dilihat pada tabel 4.19.

**Tabel 4.19 Hasil Pengujian Modul GSM**

<b>Kasus dan Hasil Uji (Kondisi Normal)</b>			
<b>Skenario Pengujian</b>	<b>Hasil Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Kesimpulan</b>
Mikrokontroler pertama kali dihidupkan dengan daya baterai dan USB, perangkat berada di luar ruangan dengan SIM Card dan antena GSM terpasang.	Modul GSM dapat terhubung dengan <i>server</i> .	Modul GSM terkoneksi ke jaringan GSM, terkoneksi ke jaringan GPRS (Internet) dan terkoneksi ke server MQTT dalam waktu $\pm$ 20 detik.	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
Modul GSM tidak mendapatkan sinyal saat sudah terkoneksi Internet.	Modul GSM terputus dengan server dan melakukan koneksi ulang.	Modul GSM terus melakukan koneksi ulang ke jaringan GSM, jaringan GPRS (Internet) dan server MQTT.	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
Modul GSM terputus dari server setelah kehilangan sinyal dan mendapatkan sinyal kembali	Modul GSM dapat terkoneksi kembali ke server setelah terputus dan mencoba koneksi ulang.	Modul GSM terkoneksi kembali ke server.	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
<b>Kasus dan Hasil Uji (Kondisi Tidak Normal)</b>			
Modul GSM berada di luar ruangan dengan SIM Card terpasang tetapi tanpa antena.	Modul GSM tidak dapat terhubung ke jaringan, Internet dan server MQTT.	Modul GSM tidak mendapatkan sinyal dan tidak dapat terhubung ke server MQTT.	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
Modul GSM berada di	Modul GSM tidak dapat	Modul GSM tidak dapat	[ <input checked="" type="checkbox"/> ] Diterima

luar ruangan dengan antena terpasang tetapi tanpa SIM Card.	terhubung ke jaringan Internet dan server MQTT.	terhubung ke jaringan dan tidak melakukan koneksi ke server MQTT.	[ ] Ditolak
Modul GSM dihidupkan dengan daya USB tetapi tanpa baterai.	Modul GSM dapat terhubung ke jaringan GSM, Internet dan terkoneksi dengan server MQTT.	Modul GSM tidak dapat terhubung ke jaringan GSM dan tidak dapat terhubung ke server MQTT.	[ ] Diterima [ √ ] Ditolak

#### 4.2.2.2 Pengujian Modul GPS

Pengujian modul GPS dilakukan untuk mengetahui apakah modul GPS dapat berfungsi sesuai dengan fungsionalitas yang dibutuhkan oleh sistem, yaitu untuk merekam riwayat lokasi perjalanan kendaraan. Hasil pengujian modul GPS dapat dilihat pada tabel 4.20.

**Tabel 4.20 Hasil Pengujian Modul GPS**

<b>Kasus dan Hasil Uji (Kondisi Normal)</b>			
<b>Skenario Pengujian</b>	<b>Hasil Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Kesimpulan</b>
Mikrokontroler dan sensor pertama kali dihidupkan, antena GPS terpasang, perangkat mikrokontroler dan sensor berada di luar ruangan.	Modul GPS dapat menerima sinyal dan mendapatkan lokasi.	Setelah $\pm$ 45 detik modul GPS menerima sinyal dari satelit dan mendapatkan data lokasi.	[ √ ] Diterima [ ] Ditolak
Antena GPS terpasang, perangkat mikrokontroler dan sensor berada di luar ruangan.	Modul GPS dapat menerima sinyal dan mendapatkan lokasi setiap 5 detik.	Setiap $\pm$ 5 detik modul GPS mendapatkan data lokasi.	[ √ ] Diterima [ ] Ditolak
Terhubung ke Internet dan server, perangkat mikrokontroler dan	Modul GPS dapat menerima sinyal dan mendapatkan lokasi	Setiap rentang 5-10 detik perangkat mikrokontroler dan	[ √ ] Diterima [ ] Ditolak

sensor dengan antena GPS berada di luar ruangan.	serta mengirimkan lokasi ke server setiap 5 detik.	sensor mengirimkan lokasi dan diterima oleh server.	
<b>Kasus dan Hasil Uji (Kondisi Tidak Normal)</b>			
Modul GPS berada di luar ruangan tanpa antena.	Modul GPS tidak dapat menerima sinyal dan tidak mendapatkan data lokasi.	Data lokasi tidak diterima oleh modul GPS.	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
Modul GPS berada di dalam ruangan dengan antena.	Modul GPS tidak dapat menerima sinyal dan tidak mendapatkan data lokasi.	Data lokasi tidak diterima oleh modul GPS.	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak

#### 4.2.2.3 Pengujian Sensor Akselerometer

Pengujian sensor akselerometer dilakukan untuk mengetahui apakah sensor akselerometer berfungsi sesuai dengan fungsionalitas yang dibutuhkan oleh sistem yaitu untuk dapat mendeteksi *hard braking*, *hard accelerating* dan *hard cornering*.

Pengujian yang dilakukan yaitu simulasi dengan menggunakan mobil *remote-control* dengan panjang 32cm, tinggi 13cm, lebar 12cm dan berat 682gram.



**Gambar 4.4 Mobil Remote-Control Untuk Simulasi Pengujian**

Adapun nilai *threshold* G-Force untuk pendeteksian disesuaikan yaitu 0.3 dari *threshold* normal. Nilai *threshold* yang digunakan pada simulasi menggunakan mobil *remote-control* ini dilihat pada tabel 4.21.

**Tabel 4.21 Nilai *Threshold* G-Force Yang Digunakan Untuk Simulasi**

No	Tipe	<i>Threshold</i> (g)
1.	Akselerasi Keras	0.09
2.	Rem Mendadak	0.18
3.	Berbelok Keras	0.15

a. Pengujian Sensor Akselerometer

Berikut adalah hasil pengujian akselerometer untuk mendeteksi kejadian dilihat pada tabel 4.22.

**Tabel 4.22 Hasil Pengujian Sensor Akselerometer**

<b>Kasus dan Hasil Uji (Kondisi Normal)</b>			
<b>Skenario Pengujian</b>	<b>Hasil Yang Diharapkan</b>	<b>Pengamatan</b>	<b>Kesimpulan</b>
Internet terhubung, GPS mendapat lokasi, mobil <i>RC</i> dijalankan maju lurus dengan kecepatan penuh dari posisi diam.	Perangkat mikrokontroler mendeteksi akselerasi keras dan mengirimkan kejadian mengendara beserta posisinya ke server dan terlihat di aplikasi web.	Kejadian mengendara dan lokasinya terkirim ke server, tersimpan di database dan nilai kejadian mengendara pada aplikasi web bertambah.	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
Internet terhubung, GPS belum mendapatkan lokasi, mobil <i>RC</i> dijalankan maju lurus dengan kecepatan penuh dari posisi diam.	Perangkat mikrokontroler mendeteksi akselerasi keras dan mengirimkan kejadian mengendara ke server dan terlihat di aplikasi web.	Kejadian mengendara terkirim ke server, tersimpan di database dan nilai jumlah akselerasi keras pada aplikasi web bertambah.	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
Internet terhubung, mobil <i>RC</i> dijalankan maju lurus dengan kecepatan lambat dari posisi diam.	Perangkat mikrokontroler tidak mendeteksi akselerasi keras, mikrokontroler tidak mengirimkan ke	Akselerasi keras tidak terdeteksi, server tidak menerima pesan akselerasi keras dan pada aplikasi web nilai	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak

	server dan nilai akselerasi keras pada aplikasi web tidak berubah	akselerasi keras tidak berubah.	
<b>Kasus dan Hasil Uji (Kondisi Tidak Normal)</b>			
Internet tidak terhubung, mobil <i>RC</i> dijalankan maju lurus dengan kecepatan penuh dari posisi diam.	Perangkat mikrokontroler tidak mengirimkan pesan akselerasi keras ke server.	Tidak ada pesan akselerasi keras terkirim ke server dan server tidak menerima pesan akselerasi keras.	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
Internet terhubung, posisi menghadap mikrokontroler diputar 90derajat dari posisi seharusnya, mobil <i>RC</i> dijalankan maju lurus dengan kecepatan penuh dari posisi diam.	Perangkat mikrokontroler tidak akan mendeteksi akselerasi keras melainkan mendeteksi berbelok keras.	Kejadian yang terdeteksi yaitu berbelok keras, terkirim ke server dan tersimpan di basis data.	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak

b. Pengujian Deteksi Akselerasi Keras (*Hard Accelerating*)

Berikut adalah data hasil pengujian pada mikrokontroler dan sensor untuk mendeteksi akselerasi keras pada kendaraan dilihat pada tabel 4.23.

**Tabel 4.23 Data Hasil Pengujian Deteksi Akselerasi Keras**

Percobaan Ke	Terkirim Ke Server	G-Force Terdeteksi	Kesimpulan
1	Ya	0.11g	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
2	Ya	0.13g	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
3	Ya	0.10g	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
4	Ya	0.14g	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
5	Tidak	-	[ <input type="checkbox"/> ] Diterima [ <input checked="" type="checkbox"/> ] Ditolak
6	Tidak	-	[ <input type="checkbox"/> ] Diterima [ <input checked="" type="checkbox"/> ] Ditolak

7	Ya	0.14g	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
8	Ya	0.12g	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
9	Ya	0.12g	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
10	Ya	0.09g	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak

Persentase keberhasilan perangkat mikrokontroler dan sensor untuk mendeteksi akselerasi keras (*hard accelerating*) pada pengujian ini yaitu (jumlah terdeteksi/total percobaan) x 100%. Jadi persentase keberhasilannya adalah  $8/10 \times 100\% = 80\%$ .

c. Pengujian Deteksi Rem Mendadak (*Hard Braking*)

Berikut adalah data hasil pengujian pada mikrokontroler dan sensor untuk mendeteksi rem mendadak pada kendaraan dilihat pada tabel 4.24.

**Tabel 4.24 Data Hasil Pengujian Deteksi Rem Mendadak**

Percobaan Ke	Terkirim Ke Server	G-Force Terdeteksi	Kesimpulan
1	Ya	0.19g	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
2	Ya	0.18g	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
3	Tidak	-	[ <input type="checkbox"/> ] Diterima [ <input checked="" type="checkbox"/> ] Ditolak
4	Ya	0.23g	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
5	Ya	0.20g	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
6	Ya	0.20g	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
7	Ya	0.21g	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
8	Ya	0.21g	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
9	Ya	0.18g	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak

10	Ya	0.22g	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
----	----	-------	--

Persentase keberhasilan perangkat mikrokontroler dan sensor untuk mendeteksi rem mendadak (*hard braking*) dan mengirimnya ke *server* pada pengujian ini yaitu (jumlah terdeteksi/total percobaan) x 100%. Jadi persentase keberhasilannya adalah  $9/10 \times 100\% = 90\%$ .

d. Pengujian Deteksi Berbelok Keras (*Hard Cornering*)

Berikut adalah data hasil pengujian pada mikrokontroler dan sensor untuk mendeteksi apabila kendaraan berbelok keras (*hard cornering*) dilihat pada tabel 4.25.

**Tabel 4.25 Data Hasil Pengujian Deteksi Berbelok Keras**

Percobaan Ke	Terkirim Ke Server	G-Force Terdeteksi	Kesimpulan
1	Ya	0.16g	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
2	Tidak	-	[ <input type="checkbox"/> ] Diterima [ <input checked="" type="checkbox"/> ] Ditolak
3	Ya	0.15g	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
4	Ya	0.16g	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
5	Ya	0.17g	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
6.	Tidak	-	[ <input type="checkbox"/> ] Diterima [ <input checked="" type="checkbox"/> ] Ditolak
7.	Tidak	-	[ <input type="checkbox"/> ] Diterima [ <input checked="" type="checkbox"/> ] Ditolak
8.	Ya	0.15	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
9.	Ya	0.15	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak
10.	Ya	0.17	[ <input checked="" type="checkbox"/> ] Diterima [ <input type="checkbox"/> ] Ditolak

Persentase keberhasilan perangkat mikrokontroler dan sensor untuk mendeteksi berbelok keras (*hard cornering*) pada pengujian ini yaitu (jumlah

terdeteksi/total percobaan) x 100%. Jadi persentase keberhasilannya adalah  $7/10 \times 100\% = 70\%$ .

### 4.2.3 Kesimpulan Hasil Pengujian Alpha

Berdasarkan hasil pengujian dengan skenario dan kasus uji sampel diatas, maka dapat ditarik kesimpulan bahwa secara fungsionalitas sistem dapat bekerja sesuai dengan yang diharapkan. Akan tetapi terdapat faktor-faktor yang memungkinkan terjadinya kesalahan pada sistem. Salah satu faktor bahwa pendeteksian kejadian dan pengiriman riwayat lokasi perjalanan membutuhkan jaringan internet dan koneksi yang stabil. Persentase keberhasilan dalam mendeteksi kejadian mengendara (*hard braking*, *hard cornering* dan *hard accelerating*) dan terkirim ke *server* pada hasil percobaan adalah 80%.

### 4.2.4 Pengujian Beta

Pengujian *beta* dilakukan untuk mengetahui sejauh mana kualitas sistem yang sudah dibangun, apakah sudah memenuhi harapan atau belum. Pengujian dilakukan dengan secara langsung mempresentasikan hasil pekerjaan kepada responden dari pihak SKK Migas.

Metode yang digunakan dalam kuesioner adalah skala likert karena umum digunakan dalam kuesioner dan banyak digunakan dalam survei. Lima pilihan umpan balik yaitu:

1. Sangat Tidak Setuju
2. Tidak Setuju
3. Cukup Setuju
4. Setuju
5. Sangat Setuju

Berikut adalah 6 pernyataan yang diajukan kepada pihak SKK Migas sebagai responden atas umpan balik pada sistem yang dibangun:

1. Dengan diterapkannya sistem ini mempermudah dalam memantau riwayat perjalanan kendaraan dinas.

2. Dengan diterapkannya sistem ini mempermudah dalam memantau apabila kendaraan sering rem mendadak, berakselerasi keras atau berbelok keras.
3. Dengan diterapkannya sistem ini mendukung evaluasi pada kendaraan dinas.
4. Dengan diterapkannya sistem ini penyajian informasi menjadi lebih lengkap dan jelas.
5. Dengan diterapkannya sistem ini laporan yang dihasilkan menjadi lebih baik dan jelas.
6. Secara Keseluruhan Sistem Ini Sesuai Dengan Apa Yang Diharapkan.

Berikut adalah jawaban yang disediakan dalam pengisian kuesioner dengan menggunakan skala likert dari skala 1-5.

**Tabel 4.26 Skala Jawaban**

Nilai	Keterangan
1	Sangat Tidak Setuju
2	Tidak Setuju
3	Cukup Setuju
4	Setuju
5	Sangat Setuju

Untuk mengetahui hasil akhir dari skala yang ada, dibutuhkan rumus untuk menghitung hasil akhir tersebut, rumusnya adalah: hasil akhir = total skor / Y \* 100. Dengan Y adalah nilai tertinggi dengan responden. Adapun kriteria interpretasi skornya berdasarkan interval adalah sebagai berikut:

**Tabel 4.27 Rentang Nilai**

Rentang Nilai	Keterangan
0%-19.99%	Sangat tidak setuju
20%-39.99%	Tidak setuju
40%-59.99%	Cukup Setuju
60%-79.99%	Setuju
80%-99.99%	Sangat setuju

#### 4.2.4.1 Kasus dan Hasil Pengujian Beta

Berikut adalah hasil persentasi masing-masing nilai jawaban kuisisioner yang diberikan kepada 3 staf SKK Migas di Divisi Fasilitas Kantor dan Keuangan dan Divisi IT.

1. Dengan diterapkannya sistem ini mempermudah dalam memantau riwayat perjalanan kendaraan dinas.

**Tabel 4.28 Hasil Kuesioner Pernyataan Nomor 1**

Kategori Jawaban	Skor	Frekuensi Jawaban	Total Skor	Jumlah Persentasi	Keputusan
Sangat Tidak Setuju	1	0	0	100%	Sangat Setuju
Tidak Setuju	2	0	0		
Kurang Setuju	3	0	0		
Setuju	4	0	0		
Sangat Setuju	5	3	15		
		3	15		

Jumlah skor ideal (kriterium) untuk seluruh item =  $5 \times 3 = 15$  (seandainya semua menjawab sangat setuju). Total jumlah skor yang diperoleh untuk pertanyaan nomor 1 adalah 15 dengan nilai persentase =  $(15/15) \times 100\% = 100\%$  dari yang diharapkan (100%). Jadi kriteria interpretasi skor berdasarkan interval pada tabel 4.25 menyimpulkan bahwa responden sangat setuju bahwa sistem yang dibangun mempermudah dalam memantau riwayat perjalanan kendaraan dinas.

2. Dengan diterapkannya sistem ini mempermudah dalam memantau apabila kendaraan sering rem mendadak, berakselerasi keras atau berbelok keras.

**Tabel 4.29 Hasil Kuesioner Pernyataan Nomor 2**

Kategori Jawaban	Skor	Frekuensi Jawaban	Total Skor	Jumlah Persentasi	Keputusan
Sangat Tidak Setuju	1	0	0	93.3%	Sangat Setuju
Tidak Setuju	2	0	0		
Kurang Setuju	3	0	0		

Setuju	4	1	4		
Sangat Setuju	5	2	10		
		3	14		

Jumlah skor ideal (kriterium) untuk seluruh item yaitu  $5 \times 3 = 15$  (seandainya semua menjawab sangat setuju). Total jumlah skor yang diperoleh untuk pernyataan nomor 2 adalah 14 dengan nilai persentase =  $(14/15) \times 100\% = 93.3\%$  dari yang diharapkan (100%). Jadi kriteria interpretasi skor berdasarkan interval pada tabel 4.26 menyimpulkan bahwa responden sangat setuju bahwa sistem yang dibangun mempermudah dalam memantau apabila kendaraan sering rem mendadak, berakselerasi keras atau berbelok keras.

3. Dengan diterapkannya sistem ini mendukung evaluasi pada kendaraan dinas.

**Tabel 4.30 Hasil Kuesioner Pernyataan Nomor 3**

Kategori Jawaban	Skor	Frekuensi Jawaban	Total Skor	Jumlah Persentasi	Keputusan
Sangat Tidak Setuju	1	0	0	100%	Sangat Setuju
Tidak Setuju	2	0	0		
Kurang Setuju	3	0	0		
Setuju	4	0	0		
Sangat Setuju	5	3	15		
		3	15		

Jumlah skor ideal (kriterium) untuk seluruh item yaitu  $5 \times 3 = 15$  (seandainya semua menjawab sangat setuju). Total jumlah skor yang diperoleh untuk pernyataan nomor 3 adalah 15 dengan nilai persentase =  $(15/15) \times 100\% = 100\%$  dari yang diharapkan (100%). Jadi kriteria interpretasi skor berdasarkan interval pada tabel 4.27 menyimpulkan bahwa responden sangat setuju bahwa sistem yang dibangun mendukung evaluasi pada kendaraan dinas.

4. Dengan diterapkannya sistem ini penyajian informasi menjadi lebih lengkap dan jelas.

**Tabel 4.31 Hasil Kuesioner Pernyataan Nomor 4**

Kategori Jawaban	Skor	Frekuensi Jawaban	Total Skor	Jumlah Persentasi	Keputusan
Sangat Tidak Setuju	1	0	0	93.3%	Sangat Setuju
Tidak Setuju	2	0	0		
Kurang Setuju	3	0	0		
Setuju	4	1	4		
Sangat Setuju	5	2	10		
		3	14		

Jumlah skor ideal (kriterium) untuk seluruh item yaitu  $5 \times 3 = 15$  (seandainya semua menjawab sangat setuju). Total jumlah skor yang diperoleh untuk pernyataan nomor 4 adalah 14 dengan nilai persentase yaitu  $(14/15) \times 100\% = 93.3\%$  dari yang diharapkan (100%). Jadi kriteria interpretasi skor berdasarkan interval pada tabel 4.31 menyimpulkan bahwa responden sangat setuju bahwa sistem yang dibangun dapat menyajikan informasi yang lebih lengkap dan jelas.

5. Dengan diterapkannya sistem ini laporan yang dihasilkan menjadi lebih baik dan jelas.

**Tabel 4.32 Hasil Kuesioner Pernyataan Nomor 5**

Kategori Jawaban	Skor	Frekuensi Jawaban	Total Skor	Jumlah Persentasi	Keputusan
Sangat Tidak Setuju	1	0	0	93.3%	Sangat Setuju
Tidak Setuju	2	0	0		
Kurang Setuju	3	0	0		
Setuju	4	1	4		
Sangat Setuju	5	2	10		
		3	14		

Jumlah skor ideal (kriterium) untuk seluruh item yaitu  $5 \times 3 = 15$  (seandainya semua menjawab sangat setuju). Total jumlah skor yang diperoleh untuk pernyataan nomor 5 adalah 14 dengan nilai persentase yaitu  $(14/15) \times 100\% = 93.3\%$  dari yang diharapkan (100%). Jadi kriteria interpretasi skor berdasarkan

interval pada tabel 4.32 menyimpulkan bahwa responden sangat setuju bahwa sistem yang dibangun dapat menghasilkan laporan yang lebih baik dan jelas.

6. Secara keseluruhan sistem ini sesuai dengan apa yang diharapkan.

**Tabel 4.33 Hasil Kuesioner Pernyataan Nomor 6**

Kategori Jawaban	Skor	Frekuensi Jawaban	Total Skor	Jumlah Persentasi	Keputusan
Sangat Tidak Setuju	1	0	0	93.3%	Sangat Setuju
Tidak Setuju	2	0	0		
Kurang Setuju	3	0	0		
Setuju	4	1	4		
Sangat Setuju	5	2	10		
		3	14		

Jumlah skor ideal (kriterium) untuk seluruh item yaitu  $5 \times 3 = 15$  (seandainya semua menjawab sangat setuju). Total jumlah skor yang diperoleh untuk pernyataan nomor 6 adalah 14 dengan nilai persentase yaitu  $(14/15) \times 100\% = 93.3\%$  dari yang diharapkan (100%). Jadi kriteria interpretasi skor berdasarkan interval pada tabel 4.33 menyimpulkan bahwa responden sangat setuju bahwa sistem yang secara keseluruhan sesuai dengan apa yang diharapkan.

#### **4.2.4.2 Kesimpulan Hasil Pengujian Beta**

Berdasarkan hasil persentase jawaban setiap responden terhadap pernyataan yang diajukan pada pengujian *Beta* dapat ditarik kesimpulan bahwa secara keseluruhan sistem yang dibangun sudah memenuhi kebutuhan dan sesuai dengan apa yang diharapkan dengan hasil 95.5% responden setuju pada sistem yang dibangun.