TUGAS AKHIR – TJ141502

PERANGKAT PEMANTAU DAN IDENTIFIKASI KONDISI REL KERETA API SEBAGAI PEMANDU PETUGAS PERAWATAN REL DAN MASINIS KERETA API

Virbyansah Achmadan Nurrohman
NRP 07211440000020

Dosen Pembimbing
Arief Kurniawan, ST., MT.
Ahmad Zaini, ST., M.Sc.

DEPARTEMEN TEKNIK KOMPUTER
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 2018

Halaman ini sengaja dikosongkan

FINAL PROJECT – TJ141502

DEVICE OF RAILWAY CONDITION MONITORING AND IDENTIFICATION AS GUIDE FOR RAILWAY MAINTENANCE OFFICER AND TRAIN DRIVER

Virbyansah Achmadan Nurrohman
NRP 07211440000020

Advisor
Arief Kurniawan, ST., MT.
Ahmad Zaini, ST., M.Sc.

Departement of Computer Engineering
Faculty of Electrical Technology
Sepuluh Nopember Institute of Technology
Surabaya 2018

Halaman ini sengaja dikosongkan

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul "Perangkat Pemantau dan Identifikasi Kondisi Rel Kereta Api sebagai Pemandu Petugas Perawatan Rel dan Masinis Kereta Api" adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Juli 2018

Virbyansah Achmadan Nurrohman
NRP. 07211440000020

Halaman ini sengaja dikosongkan

LEMBAR PENGESAHAN

PERANGKAT PEMANTAU DAN IDENTIFIKASI KONDISI REL KERETA API SEBAGAI PEMANDU PETUGAS PERAWATAN REL DAN MASINIS KERETA API

Tugas Akhir ini disusun untuk memenuhi salah satu syarat memperoleh gelar Sarjana Teknik di Institut Teknologi Sepuluh Nopember Surabaya

Oleh: Virbyansah Achmadan Nurrohman (NRP: 07211440000020)

Tanggal Ujian : 29 Juni 2018 Periode Wisuda: September 2018

Disetujui oleh:

(Pembimbing I)

Arief Kurniawan, ST., MT.
NIP: 197409072002121001

.....

(Pembimbing II)

Ahmad Zaini, ST., M.Sc.
NIP: 197504192002121003

.....

(Penguji I)

Prof. Dr. Ir. Yoyon K. Suprapto, M.Sc.
NIP. 195409251978031001

.....

(Penguji II)

Dr. Eko Mulyanto Yuniarno, ST., MT.
NIP: 196806011995121009

.....

(Penguji III)

Muhtadin, ST., M.Sc.
NIP: 198106092009121003

.....

Mengetahui
Kepala Departemen Teknik Komputer

Dr. I Ketut Eddy Purnama, ST., MT.
NIP. 196907301995121001

Halaman ini sengaja dikosongkan

ABSTRAK

Nama Mahasiswa	:	Virbyansah Achmadan Nurrohman
Judul Tugas Akhir	:	Perangkat Pemantau dan Identifikasi Kondisi Rel Kereta Api sebagai Pemandu Petugas Perawatan Rel dan Masinis Kereta Api
Pembimbing	:	1. Arief Kurniawan, ST., MT. 2. Ahmad Zaini, ST., M.Sc.

Pemeriksaan rutin kondisi rel kereta api sudah dilaksanakan oleh PT Kereta Api Indonesia (PT KAI) menggunakan kereta ukur. Akibat tidak seimbangnya jumlah kereta ukur dengan panjang jalur rel kereta api yang harus diperiksa, pemeriksaan rel kereta api hanya bisa dilaksanakan sebanyak dua kali dalam satu tahun. Hal tersebut dirasa masih belum cukup untuk memperbarui data kondisi rel. Pada tugas akhir ini dibuatlah suatu sistem pemantau kondisi rel kereta api secara *real time* dengan menggunakan sensor vibrasi untuk mengidentifikasi kondisi rel kereta api berdasarkan lokasi. Data dapat diperbarui lebih cepat karena sistem dapat diterapkan pada semua jenis kereta api. Setelah dilakukan pengujian, didapatkan tingkat validasi data berdasarkan rata-rata selisih nilai geteran sebesar 0,06 g sumbu horizontal, 0,1 g sumbu lateral, 0,19 g sumbu vertikal dengan selisih jarak sebesar 34,3 meter, nilai galat rata-rata sensor adalah galat lokasi sebesar 12,2 meter, dengan *delay* gps 2,8 detik, galat kecepatan sebesar 17,36%, dan galat akselerometer sebesar 0,059 g dan interval waktu transmisi data sebesar 1,04 detik. Dengan dikembangkannya alat ini, petugas perwatan rel kereta api akan mendapat laporan kondisi rel kereta api beradasarkan warna *marker* sesuai dengan parameter nilai indeks dan dengan aplikasi peringatan dini, masinis kereta api mendapat notifikasi batas kecepatan maksimal pada lokasi rel tertentu.

Kata Kunci : Kereta Api, Peta Digital, Indeks Rel, Batas Kecepatan, peringatan dini

Halaman ini sengaja dikosongkan

ABSTRACT

Name : Virbyansah Achmadan Nurrohman
Title : *Device of Railway Condition Monitoring and Identification As Guide For Railway Maintenance Officer And Train Driver.*
Advisors : 1. Arief Kurniawan, ST., MT
2. Ahmad Zaini, ST., M.Sc.

Inspection routine of railroad conditions has been carried out by PT Kereta Api Indonesia (PT KAI) using a measurement train. Due to the unbalanced number of trains with the length of the railway tracks to be checked, the rail inspection can only be run two times a year. It is still not enough to update the rail condition data. In this final project a real time railway monitoring system by using vibration sensor to identify rail condition based on location is developed. Data can be updated faster because the system can be applied to all types of trains. After testing, data validation rate was obtained based on average difference of 0.06 g horizontal axis, 0.1 g lateral axis, 0.19 g vertical axis with distance difference of 34.3 meters, average error value the sensor is a location error of 12.2 meters, with a gps delay of 2.8 seconds, a speed error of 17.36 %, and an accelerometer error of 0.059 g and a data transmission time interval of 1.04 seconds. With the development of this tool, railway trainers will get reports on railway conditions based on marker colors according to index value parameters and with early warning applications, train drivers receive maximum speed limit notification at certain rail locations.

Keywords : Train, Digital Map, Path Index, Speed Limit, Early Warning

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Puji dan syukur kehadirat Tuhan Yang Maha Esa atas segala karunia-Nya, penulis dapat menyelesaikan penelitian ini dengan judul **Perangkat Pemantau dan Identifikasi Kondisi Rel Kereta Api sebagai Pemandu Petugas Perawatan Rel**.

Penelitian ini disusun dalam rangka pemenuhan bidang riset di Departemen Teknik Komputer ITS, Bidang Studi Telematika, serta digunakan sebagai persyaratan menyelesaikan pendidikan S1. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Keluarga, Ibu, Bapak dan Saudara tercinta yang telah memberikan dorongan spiritual dan material dalam penyelesaian buku penelitian ini.
2. Bapak Dr. I Ketut Eddy Purnama, ST., MT. selaku Kepala Departemen Teknik Komputer, Fakultas Teknik Elektro, Institut Teknologi Sepuluh Nopember.
3. Bapak Arief Kurniawan, ST., MT. selaku dosen pembimbing I dan Bapak Ahmad Zaini, ST., M.Sc. selaku dosen pembimbing II yang selalu memberikan arahan selama mengerjakan penelitian tugas akhir ini.
4. Bapak-ibu dosen pengajar Departemen Teknik Komputer, atas pengajaran, bimbingan, serta perhatian yang diberikan kepada penulis selama ini.
5. Secara khusus penulis mengucapkan terimakasih kepada PT Kereta Api Indonesia (Persero) Daop 8 Surabaya khususnya Bapak Yulianto, Bapak Jamiran, dan Bapak Djuri dari bagian Jalan Rel dan Jembatan yang telah memberi izin, arahan, dan bimbingan guna pengambilan data di lapangan.
6. Seluruh teman-teman dari Teknik Komputer, Laboratorium B401, dan B201 Teknik Komputer ITS.

Kesempurnaan hanya milik Allah SWT, untuk itu penulis memohon segenap kritik dan saran yang membangun. Semoga penelitian ini dapat memberikan manfaat bagi kita semua. Amin.

Surabaya, Juli 2018

Penulis

Halaman ini sengaja dikosongkan

DAFTAR ISI

Abstrak

Abstract

KATA PENGANTAR

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

NOMENKLATUR

1 PENDAHULUAN **1**
1.1 Latar belakang 1
1.2 Permasalahan 2
1.3 Tujuan 2
1.4 Batasan masalah 3
1.5 Sistematika Penulisan 3
1.6 Relevansi 4

2 TINJAUAN PUSTAKA **5**
2.1 Getaran Kereta Api 5
2.2 Nilai Indeks Jalur 6
2.3 Menetapkan Nilai Batas Kecepatan Maksimal 8
2.4 Sensor Akselerometer Digital 9
2.4.1 Kalibrasi Sensor Getar 10
2.5 *Global Positioning System* (GPS) 11
2.5.1 Segmen Penyusun GPS 12
2.5.2 *NMEA* 13
2.6 *Single Board Computer* Raspberry Pi 15
2.7 Modul MYSQLdb 16
2.8 *Cloud Storage* 16

3 DESAIN DAN IMPLEMENTASI SISTEM **19**
3.1 Desain Sistem 19
3.1.1 Sensor *Node* 20
3.1.2 Desain *Database* Pada *Cloud Storage* 21
3.1.3 *User Interface*(UI) 26
3.2 Implementasi Sistem 28
3.2.1 Pemasangan Sensor *Node* di Kereta Api . . . 29
3.2.2 Akusisi pada Sensor *Node* 30
3.2.3 Penyimpanan ke *Cloud Storage* 32
3.2.4 *User Interface* pada Sensor *Node* 34
3.2.5 Laporan Nilai Indek Rel Berbasis Web 36
3.2.6 Aplikasi Notifikasi Batas Kecepatan Maksimal 39

4 PENGUJIAN DAN ANALISA **43**
4.1 Pengujian *User Interface* 43
4.1.1 *User Interface* Laporan Nilai Indeks Rel . . . 43
4.1.2 *User Interface* Pada Sensor *Node* 46
4.2 Pengujian Perangkat 50
4.2.1 Pengujian Akurasi Lokasi Kondisi Statis . . . 50
4.2.2 Pengujian Akurasi Lokasi Kondisi Dinamis . 52
4.2.3 Pengujian Akurasi Kecepatan 54
4.2.4 Pengujian Akurasi Sensor Akselerometer . . . 57
4.3 Pengujian Interval Waktu Penyimpanan ke *Database* 59
4.4 Pengujian Sistem Peringatan Dini 60
4.5 Pengujian Validasi Data 62

5 PENUTUP **65**
5.1 Kesimpulan 65
5.2 Saran 65

DAFTAR PUSTAKA **67**

Biografi Penulis **69**

DAFTAR GAMBAR

1.1 Presentasi jenis kecelakaan perkeretaapian yang diinvestigasi KNKT (2010-2016).[1] 1

2.1 Korelasi antara kecepatan aktual dengan besar nilai getaran untuk getaran a. horizontal, b. vertikal.[4] . 5
2.2 *Pinout* sensor akselerometer digital ADXL345.[6] . . 9
2.3 Ilustrasi posisi untuk kalibrasi sensor akselerometer.[8] 11
2.4 Modul sensor Adafruit GPS *Ultimate*.[9] 11
2.5 Ilustrasi segmen GPS.[10] 12
2.6 Raspberry Pi 3 Model B pinout.[12] 15
2.7 Alur kerja MYSQLdb. 17

3.1 Gambaran umum kerja sistem. 19
3.2 Skema alur kerja sensor *node*. 20
3.3 Desain *Entity Relation Database*. 21
3.4 Tampilan desain UI pada sensor *node*. 26
3.5 Tampilan desain UI pada peta digital berbasis web. . 27
3.6 Tampilan desain aplikasi notifikasi batas kecepatan maksimal. 28
3.7 Skema pemasangan sensor *node* di *bogie* kereta penumpang. 29
3.8 Implementasi pemasangan sensor *node* di kereta penumpang. 30
3.9 Skema konfigurasi sensor *node*. 31
3.10 Diagram alir akusisi data ke SBC. 32
3.11 Diagram alir penyimpanan ke *cloud storage*. 33
3.12 Diagram alir visualisasi data pada *user ineterface* sensor *node*. 34
3.13 Implementasi *User Interface* pada sensor *node*. . . . 35
3.14 Visualisasi laporan nilai indeks dalam bentuk *marker* berwarna pada peta digital berbasis web. 37
3.15 Visualisasi laporan informasi detail berbasis web. . . 38
3.16 Diagram alir laporan indeks berbasis web. 39
3.17 Implementasi aplikasi notifikasi batas kecepatan maksimal kereta api. 40

3.18 Diagram alir aplikasi notifikasi batas kecepatan maksimal. 41
3.19 GET *request* dan *response* data. 42

4.1 *Marker* dengan indeks rel sebesar 569,73 berwarna hijau. 44
4.2 *Marker* dengan indeks rel sebesar 97,97 berwarna kuning. 44
4.3 *Marker* dengan indeks rel sebesar 79,8 berwarna jingga. 45
4.4 *Marker* dengan indeks rel sebesar 17,0 berwarna merah. 45
4.5 Visualisasi besar nilai getaran keadaan normal. . . . 46
4.6 Visualisasi besar nilai getaran keadaan abnormal. . . 46
4.7 Blok pada *User interface* sensor *node* yang diujikan. . 47
4.8 Tampilan menu *check* GPS. 47
4.9 Tampilan pada menu *check* akselerometer. 48
4.10 Tampilan pada menu proses kalibrasi sensor akselerometer. 48
4.11 Tampilan menu proses akuisisi sensor *node*. 49
4.12 Tampilan menu proses pengiriman data ke *cloud storage*. 49
4.13 Rata-rata dan standar deviasi galat akurasi lokasi GPS. 52
4.14 Rata-rata dan standar deviasi galat akurasi lokasi GPS. 55
4.15 Rata-rata dan standar deviasi galat kecepatan GPS. 57
4.16 Pengujian aplikasi peringatan dini batas kecepatan kereta api berdasarkan parameter jarak: (a)lebih dari 500 m, (b)499-400 m, dan (c) 399-300 m. 61
4.17 Pengujian aplikasi peringatan dini batas kecepatan kereta api berdasarkan parameter jarak: (a)299-200 m, (b)199-100 m, dan (c) 99-0 m. 61

DAFTAR TABEL

2.1 Hubungan batas nilai getaran dengan kecepatan aktual kereta api.[4] 6
2.2 Kualifikasi nilai indeks jalur rel.[5] 8
2.3 Struktur data dari $GPGGA.[11] 14

3.1 Struktur tabel data indeks. 23
3.2 Struktur tabel petugas. 23
3.3 Struktur tabel stasiun. 24
3.4 Struktur tabel kereta api. 25
3.5 Struktur tabel admin. 25
3.6 Struktur tabel rute. 25
3.7 Parameter warna *marker*. 27

4.1 Akurasi lokasi statis GPS pagi cerah. 50
4.2 Akurasi lokasi statis GPS siang cerah. 51
4.3 Akurasi lokasi statis GPS malam cerah. 51
4.4 Akurasi lokasi dinamis GPS pagi cerah. 53
4.5 Akurasi lokasi dinamis GPS pagi cerah. 53
4.6 Akurasi lokasi dinamis GPS malam cerah. 54
4.7 Akurasi kecepatan GPS pagi cerah. 55
4.8 Akurasi kecepatan GPS siang cerah. 56
4.9 Akurasi kecepatan GPS malam cerah. 56
4.10 Nilai sensor akselerometer sumbu X, Y, dan Z pada sensor *node* dan gawai. 58
4.11 Nilai minimum, rata-rata, dan maksimum galat sensor akselerometer. 58
4.12 Waktu dan tanggal pengiriman data (GPS) dan penerimaan data (*database*). 59
4.13 Nilai minimum, rata-rata, dan maksimum waktu pengiriman data. 60
4.14 Hasil pengujian aplikasi peringatan dini batas kecepatan maksimal kereta api. 62
4.15 Hasil pengujian validasi data. 63

Halaman ini sengaja dikosongkan

NOMENKLATUR

ah	:	Nilai getaran horizontal (g)
av	:	Nilai getaran vertikal (g)
V_{act}	:	Kecepatan aktual (km/jam)
V_{max}	:	Kecepatan maksimal (km/jam)
N	:	Nilai indeks jalur
N_x	:	Nilai indeks jalur horizontal
N_y	:	Nilai indeks jalur vertikal
A_{off}	:	*Offset error* dari sensor akselerometer (g)
A_{out}	:	Nilai sensor akselerometer sebelum kalibrasi (g)
A_{act}	:	Nilai sensor akselerometer setelah kalibrasi (g)
Gn	:	Gain dari sensor akselerometer (g)
A_{g+}	:	Nilai sensor akselerometer ketika posisi positif (g)
A_{g-}	:	Nilai sensor akselerometer ketika posisi negatif (g)

Halaman ini sengaja dikosongkan

BAB 1
PENDAHULUAN

Penelitian ini di latar belakangi oleh berbagai kondisi yang menjadi acuan. Selain itu juga terdapat beberapa permasalahan yang akan dijawab sebagai luaran dari penelitian.

1.1 Latar belakang

Seiring meningkatnya jumlah layanan kereta api, angka kecelakaan transportasi kereta api juga relative bertambah. Dijelaskan pada Gambar 1.1 yakni data investigasi kecelakaan perkeretaapian yang dilaksanakan oleh Komite Nasional Keselamatan Transportasi (KNKT) dari tahun 2010 hingga 2016 penyebab terbesar kecelakaan kereta api adalah anjloknya kereta api dari perlintasan rel kereta api sebesar 68% [1].

Gambar 1.1: Presentasi jenis kecelakaan perkeretaapian yang diinvestigasi KNKT (2010-2016).[1]

Saat ini kereta api merupakan moda transportasi umum berbasis rel yang banyak digunakan oleh seluruh lapisan masyarakat di Indonesia. Hal tersebut dikarenakan kereta api menawarkan berbagai kemudahan, keamanan, kenyaman, dan ketepatan waktu merupakan kelebihan yang dimiliki oleh kereta api. Hal tersebut telah diatur dalam Undang-undang Republik Indonesia Nomor 22 Tahun

2009 tentang lalu lintas angkutan jalan[2]. Di Indonesia, PT Kereta Api Indonesia (Persero) atau disingkat PT KAI adalah perusahan satu-satunya milik pemerintah yang menyediakan jasa transportasi kereta api. Hal ini disebabkan kereta api yang melaju diatas rel melebihi batas kecepatan maksimal yang telah ditentukan berdasarkan tingkat kerusakan rel pada lokasi tersebut. Aditya menjabat sebagai *Junior Manager Track Maintenance* PT KAI mengatakan, pemeriksaan rel dengan kereta ukur dilakukan setiap enam bulan sekali [3]. Namun pemeriksaan berkala yang dilakukan oleh kereta ukur masih dirasa kurang cukup untuk memperbarui data kondisi rel dan batas kecepatan maksimum berdasarkan koordinat.

Berdasarkan uraian latar belakang, dibuatlah sebuah alat yang dapat mengintegrasikan sensor akselerometer sebagai pembaca getaran kereta api ketika melintas diatas rel, GPS, serta internet sebagai media komunuikasi untuk melakukan pemantauan terhadap kondisi lintasan rel kereta api berdasarkan geolokasi sehingga diharapkan dapat menjadi peringatan bagi petugas perwatan rel kereta api terhadap kerusakan rel dan peringatan dini batas kecepatan maksimal kepada masinis kerata api saat kereta akan melaju diatas rel tertentu.

1.2 Permasalahan

Rumusan masalah tugas akhir ini adalah kurangnya pemeriksaan kondisi rel kereta api secara berkala. Hal ini disebabkan tidak seimbangnya antara banyaknya jalur rel kereta api dengan jumlah kereta ukur. Proses pengolahan data pengukuran kondisi jalur rel kereta api hingga mejadi data akhir memerlukan waktu yang lama. Dan dibutuhkannya kondisi rel kereta api berdasarkan koordinat.

1.3 Tujuan

Berdasarkan rumusan permasalahan, tujuan dari tugas akhir ini adalah membuat suatu sistem purwarupa pemeriksaan kondisi rel kereta api berdasarkan lokasi serta identifikasi menggunakan sensor vibrasi untuk mengetahui tingkat getaran yang dihasilkan kereta api ketika melintas diatas rel kereta api yang bermasalah.

1.4 Batasan masalah

Batasan masalah yang timbul dari permasalahan Tugas Akhir ini adalah:

1. Pengujian dilakukan hanya pada satu jalur rel kereta api.
2. Menentukan nilai indeks jalur dengan memperhatikan nilai getaran-getaran rel terjadi apabila dilewati oleh kereta api.
3. Data hasil pengujian ditampilkan dalam bentuk web.

1.5 Sistematika Penulisan

Laporan penelitian Tugas akhir ini tersusun dalam sistematika dan terstruktur sehingga mudah dipahami dan dipelajari oleh pembaca maupun seseorang yang ingin melanjutkan penelitian ini. Alur sistematika penulisan laporan penelitian ini yaitu:

1. BAB I Pendahuluan
   Bab ini berisi uraian tentang latar belakang permasalahan, penegasan dan alasan pemilihan judul, sistematika laporan, tujuan dan metodologi penelitian.
2. BAB II Dasar Teori
   Pada bab ini berisi tentang uraian secara sistematis teori-teori yang berhubungan dengan permasalahan yang dibahas pada penelitian ini. Teori-teori ini digunakan sebagai dasar dalam penelitian, yaitu informasi terkait getaran kereta api, rumus pencarian nilai indeks rel, rumus pencarian batas kecepatan maksimal kereta api, dan *datasheet* setiap komponen penyusunan perangkat.
3. BAB III Perancangan Sistem dan Impementasi
   Bab ini berisi tentang penjelasan-penjelasan terkait eksperimen yang akan dilakukan dan langkah-langkah data diolah hingga menghasilkan visualisasi. Guna mendukung eksperimen pada penelitian ini, digunakanlah blok diagram atau *work flow* agar sistem yang akan dibuat dapat terlihat dan mudah dibaca untuk implementasi pada pelaksaan tugas akhir.
4. BAB IV Pengujian dan Analisa
   Bab ini menjelaskan tentang pengujian eksperimen yang dila-

kukan terhadap data dan analisanya. Beberapa teknik visualisasi akan ditunjukan hasilnya pada bab ini dan dilakukan analisa terhadap hasil visualisasi dan informasi yang didapat dari hasil mengamati visualisasi yang tersaji

5. BAB V Penutup
   Bab ini merupakan penutup yang berisi kesimpulan yang diambil dari penelitian dan pengujian yang telah dilakukan. Saran dan kritik yang membangun untuk pengembangkan lebih lanjut juga dituliskan pada bab ini.

1.6 Relevansi

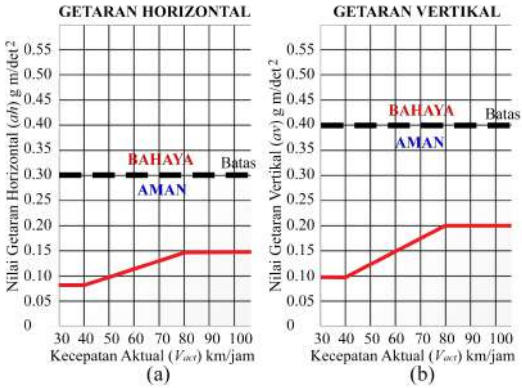
Penelitian mengenai *internet of things* merupakan bidang penelitian yang sangat dibutuhkan dan dipakai dalam pemenuhan kebutuhan informasi dan komputasi yang semakin beragam saat ini. Dalam tugas akhir ini, dilakukan penelitian mengenai pengembangan sistem indentifikasi kondisi rel kereta api dini guna pencegahan kecelakaan kereta api yang disebabkan oleh kerusakan rel.

BAB 2
TINJAUAN PUSTAKA

Demi mendukung penelitian ini, dibutuhkan beberapa teori penunjang sebagai bahan acuan dan referensi. Dengan demikian penelitian ini menjadi lebih terarah.

2.1 Getaran Kereta Api

Getaran kereta api menurut arahnya, dibedakan menjadi dua jenis getaran yang akan terjadi apabila rel dilewati oleh rangkaian kereta api, yaitu getaran horizontal dan getaran vertikal. Getaran-getaran yang ditimbulkan oleh kereta api memiliki dua parameter, yaitu berat beban kereta yang lewat dan kecepatan kereta yang lewat. Apabila kereta yang lewat merupakan kereta ukur dan yang digunakan sebagai kereta ukur adalah kereta dengan standar tertentu sehingga tidak timbul getaran kereta sebagai akibat kondisi keretanya sendiri, maka parameter yang digunakan hanya satu saja, yaitu kecepatan yang sedang dijalani oleh kereta ukur tersebut.[4] Gambar 2.1 menunjukan korelasi antara kecepatan aktual dengan besar nilai getaran yang terjadi.



Gambar 2.1: Korelasi antara kecepatan aktual dengan besar nilai getaran untuk getaran a. horizontal, b. vertikal.[4]

Berdasarkan gambar korelasi antara kecepatan aktual dengan nilai getaran yang terjadi, maka dapat disusun kedalam tabel batas nilai getaran yang diijinkan seperti Tabel 2.1.

Tabel 2.1: Hubungan batas nilai getaran dengan kecepatan aktual kereta api.[4]

Kecepatan aktual (Km/jam)	**Batas nilai getaran yang diijinkan**	
	Horizontal (ah) g ms^{-2}	Vertikal (av) g ms^{-2}
40	0.075	0.100
45	0.085	0.115
50	0.095	0.125
55	0.105	0.140
60	0.115	0.150
65	0.122	0.165
70	0.130	0.175
75	0.140	0.190
80	0.150	0.200
85	0.150	0.200
90	0.150	0.200
95	0.150	0.200
100	0.150	0.200

Penetapan batas bahaya untuk nilai getaran horizontal diperhatikan lebih ketat dari pada nilai getaran vertikal, karena getaran-getaran horizontal menjadi penyebab kereta api cenderung keluar dari jalur rel kereta api.

2.2 Nilai Indeks Jalur

Indeks jalur rel kereta api digunakan untuk menilai baik atau tidaknya kondisi dari suatu jalur rel kereta api dilakukan dengan menghitung besarnya nilai indeks N [5]. Berdasarkan Persamaan 2.1 dan 2.2 merupakan persamaan guna mencari nilai indeks saat kecepatan aktual lebih dari sama dengan 80 km/jam dan pada persamaan 2.3 dan 2.4 merupakan persamaan guna mencari nilai indeks saat kecepatan aktual antara 40 hingga 80 km/jam.

$$V_{act} \geq 80 \text{ km/jam} \rightarrow N_x = \frac{15}{ah} \tag{2.1}$$

$$V_{act} \geq 80 \text{ km/jam} \rightarrow N_y = \frac{20}{av} \tag{2.2}$$

$$40 \text{ km/jam} \leq vact < 80 \text{ km/jam} \rightarrow N_x = \frac{3 \times V_{act}}{16 \times ah} \tag{2.3}$$

$$40 \text{ km/jam} \leq vact < 80 \text{ km/jam} \rightarrow N_y = \frac{4 \times V_{act}}{16 \times av} \tag{2.4}$$

Persamaan 2.1 merupakan pencarian nilai indeks horizontal N_x dan Persamaan 2.2 merupakan pencarian nilai indeks vertikal N_y saat kecepatan aktual kereta api lebih dari sama dengan 80 km/jam dimana ah merupakan nilai getaran horizontal dan av merupakan nilai getaran vertikal. Sedangkan Persamaan 2.3 merupakan pencarian nilai N_x dan Persamaan 2.4 merupakan pencarian nilai N_y saat kecepatan aktual kereta api lebih besar sama dengan 40 km/jam hingga 79 km/jam dimana kedua persamaan tersebut menggunakan parameter kecepatan aktual kereta api V_{act}. Dan Jika V_{act} kurang dari 40 km/jam, maka nilai getaran diabaikan sehingga tidak perlu menghitung nilai indeks rel.

Setelah mendapatkan nilai N_x dan N_y, dilakukan pencarian nilai indeks rel N meggunakan Persamaan 2.5 dan Persamaan 2.6.

$$N_x < N_y \rightarrow N = N_x \tag{2.5}$$

$$N_y < N_x \rightarrow N = N_y \tag{2.6}$$

Persamaan 2.5 jika nilai N_x lebih kecil dari nilai N_y, maka nilai indek N sama dengan N_x. Dan Persamaan 2.6 jika nilai N_y lebih kecil dari nilai N_x, maka nilai indek N sama dengan N_y.

Berdasarkan besar nilai dari indek jalur, kondisi rel kereta api dapat dikualifikasikan kedalam empat kelas, berikut adalah susunan

kualifikasi kondisi rel kereta api berdasarkan nilai indeks rel kereta api yang dapat dilihat pada Tabel 2.2. Pengkualifikasian jalur kedalam nilai indeks jalur rel dapat diartikan bahwa secara normal jalur rel kereta api yang terpelihara baik harus dapat mencapai harga nilai indeks yang lebih besar atau paling rendah sama dengan 100 tidak lebih kecil dari harga ini, apalagi tidak lebih kecil dari 85. Jika terjadi nilai indeks kurang dari 85, maka perlu segera untuk diadakan perbaikan dan peningkatan kondisi. Namun pelu diingat disini, bahwa nilai index tersebut dihitung berdasarkan harga nilai getaran rata-rata yang terjadi disepanjang jalur rel kereta api. Pada titik-titik tertentu di sepanjang jalur rel tersebut masih terdapat nilai getaran yang besarnya mungkin melebihi batas yang diijinkan yaitu 0,3 g untuk nilai getaran horizontal dan 0,4 g untuk nilai vertikal, sehingga menimbulkan kejutan-kejutan horizontal atau vertikal yang keras dan berbahaya.

Tabel 2.2: Kualifikasi nilai indeks jalur rel.[5]

Nilai Indeks	**Kondisi Rel**
$N \geq 100$	Jalur dalam kondisi baik
$100 > N \geq 85$	Jalur dalam kondisi kurang baik
$85 > N \geq 70$	Jalur dalam kondisi buruk
$70 > N$	Jalur dalam kondisi sangat buruk

2.3 Menetapkan Nilai Batas Kecepatan Maksimal

Syarat kondisi jalur rel kereta api yang baik adalah nilai indeks $N \geq 100$. Apabila dengan mengambil pedoman syarat kondisi jalur yang baik adalah suatu harga yang pasti, misalnya 100. Maka pengukuran jalur dalam keadaan baik dengan kecepatan pengukuran akan menghasilkan nilai batas kecepatan maksimal seperti pada Persamaan 2.7.[5]

$$V_{max} = \frac{V_{act}}{100} \times N \tag{2.7}$$

Pada Persamaan 2.7 berfungsi untuk menentapkan batas kecepatan maksimal kereta api V_{max} yang diperoleh dari kecepatan aktual kereta api V_{act} dan nilai indeks rel N.

2.4 Sensor Akselerometer Digital

Sensor akselerometer yang digunakan dalam penelitian ini adalah sensor akselerometer digital ADXL345. Dilihat dari Gambar 2.2, Sensor ADXL345 adalah *Integrated Circuit Micro-Electro Mechanical System* atau IC MEMS memiliki ukuran fisik kecil dan tipis sehingga konsumsi daya yang dibutuhkan sangat rendah yakni 2.0 V hingga 3.6 V. Seperti yang dijelaskan pada *datasheet* [6], sensor ADXL345 memiliki jangkauan pengukuran (dengan resolusi tinggi 13-bit) hingga ± 16g (grafitasi). Sensor ADXl345 dapat mengukur tiga arah yaitu arah horizontal (± X), arah vertikal (± Z), dan arah lateral (± Y). Sinyal keluaran tegangan analog yang dihasilkan oleh sensor ADXL345 berbanding lurus dengan percepatan. Getar dapat mengukur percepatan statis gravitasi dalam aplikasi deteksi kemiringan dan juga mengukur percepatan dinamis yang dihasilkan dari gerakan, kejutan atau getaran.

Gambar 2.2: *Pinout* sensor akselerometer digital ADXL345.[6]

Komunikasi serial pada sensor ADXL345 tersedia ada dua macam yaitu I^2C dan SPI. Dengan kedua macam komunikasi, ADXL345 bekerja sebagai *slave*. Jika sensor ADXL345 menggunakan komunikasi I^2C, maka pin $\overline{CS}$ (*Chip Select*) harus dihubungkan ke tegangan atau pin $V_{DD\ I/O}$. Dengan komunikasi I^2C kecepatan standar sebesar 100 KHz dan mendapatkan I^2C *address* adalah 0x53.

2.4.1 Kalibrasi Sensor Getar

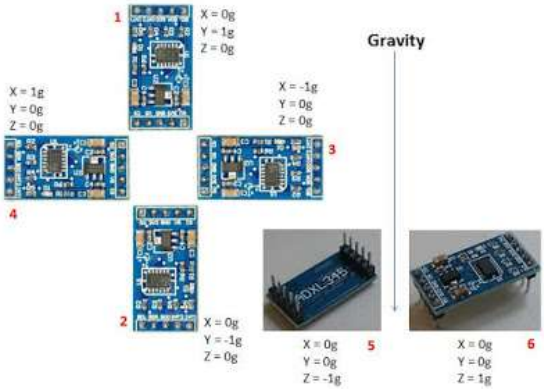
Nilai keluaran yang dihasilkan oleh sensor akselerometer seringkali tidak presisi dengan kondisi sebenarnya. Sehingga perlu dilakukan kalibrasi sensor untuk mendapatkan nilai *offset* dan *gain* tiap sumbu yang nantinya digabungkan untuk mendapatkan nilai yang valid. Pada kalibrasi sensor akselerometer digital, diasumsikan bahwa sensor yang digunakan adalah sensor akselerometer digital yang ideal. Sensor ideal yang dimaksud adalah perangkat tanpa *offset* 0 g dan dengan sensivitas sempurna (dinyatakan dengan mV/g untuk sensor analog dan LSB/g untuk sensor digital).[7] Pada Persamaan 2.8 digunakan untuk mencari nilai A_{off}, sedangkan pada Persamaan 2.9 digunakan untuk mencari *gain*. Dalam mendapatkan nilai setelah proses kalibrasi dapat menggunakan Persamaan 2.10.

$$A_{off} = 0.5 \times (A_{g+} + A_{g-}) \tag{2.8}$$

$$Gn = 0.5 \times (A_{g+} - A_{g-}) \tag{2.9}$$

$$A_{act} = \frac{(A_{out} - A_{off})}{Gn} \tag{2.10}$$

Berdasarkan Persamaan 2.8, A_{off} pada masing-masing sumbu sensor akselerometer dapat dihitung dengan cara membagi dua hasil dari penjumlahan nilai getaran pada sumbu positif A_{g+} dan sumbu negatif A_{g-}. Pada Persamaan 2.9, Gn pada masing-masing sumbu sensor akselerometer dengan cara membagi dua hasil dari selisih nilai A_{g+} dan A_{g-}. Nilai A_{g+} dan A_{g+} masing-masing sumbu sensor akselerometer diperoleh dengan cara mengubah posisi akselerometer menghadap ke arah gravitasi Bumi seperti diilustrasikan pada Gambar 2.3. Pada Persamaan 2.10, A_{act} diperoleh dengan cara mengurangi nilai sensor akselerometer yang belum terkalibrasi A_{out} dengan nilai *offset* A_{off}. Hasil dari selisih antara nilai A_{out} dengan nilai A_{off}, dibagi dengan nilai Gn sensor akselerometer.



Gambar 2.3: Ilustrasi posisi untuk kalibrasi sensor akselerometer.[8]

2.5 *Global Positioning System* (GPS)

GPS (*Global Positioning System*) yang digunakan adalah GPS bertipe Adafruit GPS *Ultimate* versi 3 yang dapat dilihat pada Gambar 2.4.

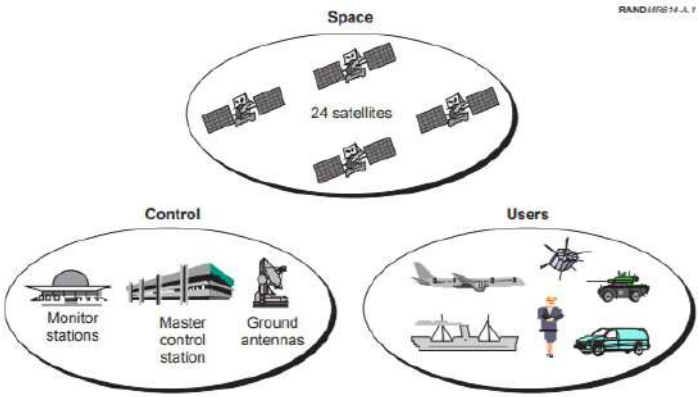
Gambar 2.4: Modul sensor Adafruit GPS *Ultimate*.[9]

Sensor GPS *Ultimate* versi 3 memiliki sembilan keluaran antara lain pin PPS (*Pulse-Per-Second*), Vin Tegangan input, GND (*Ground*), Tx (*transmitter*), Rx(*receiver*), FIX, VBAT, EN, dan

3.3V.[9] Sensor GPS Ultimate menggunakan standard NMEA 0183 dengan kecepatan *baud rate* 9600. Adafruit GPS *Ultimate* memiliki spesifikasi antara lain menggunakan *chipset* MTK3339 dan didukung antena *external* sehingga dapat melacak hingga 22 satelit pada 66 saluran. Waktu yang dibutuhkan sensor GPS untuk mendapatkan sinyal dari kondisi mati hingga hidup kurang lebih 34 detik. Selain itu, tingkat keakuratan posisi dan kecepatan oleh GPS Ultimate ini adalah sebesar 1.8 meter dan 0.1 meter/detik. GPS ini juga dilengkapi antena tambahan untuk menambah kekuatan sinyal dan baterai RTC (*Real Time Clock*) agar GPS tetap dapat melacak posisi satelit ketika kehilanagan daya. Sensor Adafruit GPS Ultimate dapat menggunakan GPSD (*open source GPS daemon*).

2.5.1 Segmen Penyusun GPS

Dilustrasikan pada Gambar 2.5, ada tiga segmen dalam sistem GPS yaitu Segmen Sistem Kontrol (*Control Segment*), Segmen Angkasa (*Space Segment*), dan Segmen Pengguna (*User Segment*).



Gambar 2.5: Ilustrasi segmen GPS.[10]

Segmen sistem kontrol GPS adalah otak dari GPS. Tugas segmen sistem kontrol adalah mengatur semua satelit GPS yang ada agar berfungsi sebagaimana mestinya. Amerika Serikat mengoperasikam sistem ini dari Sistem Kontrol Utama (*Master Station*) di

Falcon Air Force Base di Colorado Springs, Amerika Serikat. Segmen sistem kontrol ini juga termasuk empat stasiun monitor tak berawak (*Unmanned Station*) yang berlokasi menyebar di seluruh dunia terletak di Hawai, Ascension Island, Diego Garcia, dan Kwajelin Atoll yang merupakan daerah ekuator.[10]

Segmen angkasa adalah satelit-satelit GPS yang mengorbit di angkasa sebagai stasiun radio. Susunan dari satelit-satelit GPS di angkasa biasa disebut dengan Konstelasi Satelit GPS. Konstelasi standar dari satelit GPS terdiri dari 24 satelit yang menempati 6 bidang orbit. Satelit GPS tersebut dilengkapi antena-antena untuk mengirim dan menerima sinyal-sinyal gelombang. Gelombang tersebut selanjutnya dipancarkan ke Bumi dan dapat diterima oleh beberapa *receiver* GPS yang ada di Bumi dan dapat digunakan untuk menentukan informasi posisi, kecepatan, dan waktu.

Segmen pengguna adalah para pengguna satelit GPS di manapun berada, baik untuk kepentingan sipil ataupun kepentingan militer. Dalam hal ini alat penerima sinyal GPS (*GPS Receiver*) diperlukan untuk menerima dan memproses sinyal-sinyal dari satelit GPS untuk digunakan dalam penentuan posisi, kecepatan, dan waktu. Catatan yang penting adalah *GPS Receiver* tidak mengirimkan kembali sinyal apapun kepada satelit GPS dan juga jumlah pengguna dalam satu waktu adalah tidak terbatas serta tidak terjadi interferensi antar pengguna.

2.5.2 *NMEA*

NMEA *(National Marine Electronics Association)*-0183 dikembangkan secara spesifik untuk standar industri sebagai antar-muka bermacam-macam alat kelautan yang diperkenalkan sejak tahun 1983 [Pet06]. Standar tersebut diberikan untuk alat kelautan yang mengirimkan informasi ke komputer maupun alat lainnya. Contoh peralatan yang mengeluarkan data NMEA adalah *GPS (Global Positioning System)* NMEA-0183 berisi informasi yang berhubungan dengan geografi seperti tentang waktu, *longitude*, *latitude*, ketinggian, kecepatan dan masih banyak lagi. Untuk menampilkan informasi yang lebih dimengerti oleh user data NMEA-0183 perlu diolah lebih lanjut. Standar NMEA-0183 menggunakan format ASCII sederhana, masing-masing kalimat mendefinisikan isi masing-masing

tipe pesan yang dapat dipilah-pilah. Lima karakter pertama berupa setelah tanda $ disebut *field* alamat. Dua karakter pertama pada *address* disebut *Talker-ID*. Setelah *Talker-ID* mengikuti dibelakangnya tiga karakter yang menjelaskan tipe kalimat. Sedangkan tiap data dipisahkan dengan koma, jika ada *field* kosong maka tidak terisi apapun diantara dua koma dan diakhiri oleh *Carriage Return + Line Feed* (CR+LF). Salah satu tipe struktur dari NMEA-0183 adalah $GPGGA. Contoh *output* dari GPS bertipe $GPGGA adalah $GPGGA, 092204.999, 4250.5589, S,14718.5084, E, 1, 04, 24.4, 19.7, M, , , , 0000*1F. Struktur data dari contoh *output* $GPGGA tersebut dapat dilihat pada Tabel 2.3. [11]

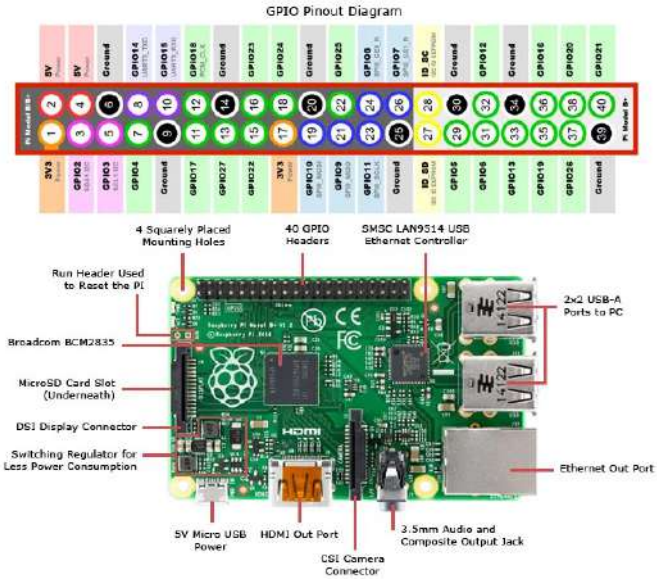
Tabel 2.3: Struktur data dari $GPGGA.[11]

Field	**Contoh isi**	**Deskripsi**
Sentence ID	$GPGGA	
UTC Time	092204.999	hhmmss.sss
Latitude	4250.5589	*ddmm.mmmm*
N/S Indicator	S	N = North, S = South
Longitude	14718.5084	*dddmm.mmmm*
E/W Indicator	*E*	E = East, W = West
Position Fix	*1*	*0 = Invalid, 1 = Valid SPS,* *2 = Valid DGPS, 3 = Valid PPS*
Satellites Used	*04*	*Satellites being used (0-12)*
HDOP	*24.4*	*Horizontal dilution of precision*
Altitude	19.7	*Altitude in meters according to WGS-84 ellipsoid*
Altitude Units	M	*M = Meters*
Geoid Seperation		*Geoid seperation in meters according to WGS-84 ellipsoid*
Seperation Units		*M = Meters*
DGPS Age		*Age of DGPS data in seconds*
DGPS Station ID	0000	
Checksum	*1F	
Terminator	CR/LF	

2.6 *Single Board Computer* Raspberry Pi

Raspberry Pi merupakan SBC (*Single Board Computer*) yang dapat digunakan untuk menjalankan perangkat lunak maupun proses pengolahan data. Selain itu Raspberry Pi dapat melakukan proses visualisasi data seperti halnya pada komputer personal. Raspberry Pi 3 Model B berbasis sistem operasi Raspbian, yaitu sistem operasi yang dikembangkan oleh perusahaan Raspberry Pi yang merupakan turunan dari sistem operasi Debian.

Gambar 2.6 merupakan *pinout* dari Raspberry Pi 3 Model B. Pada Raspberry Pi 3 Model B menggunakan CPU (*Central Processing Unit*) Broadcom BCM2837 64bit dengan kecepatan 1.2 Ghz dan GPU(*Graphic Processing Unit*) jenis *Broadcom Video Core* IV. Raspberry Pi 3 Model B memiliki kapasitas RAM (*Random Access Memory*) sebesar 1GB dan menggunakan microSD sebagai *storage*, sehingga *operating system* dan *file* tersimpan pada microSD tersebut.



Gambar 2.6: Raspberry Pi 3 Model B pinout.[12]

Selain itu Raspberry Pi 3 Model B dilengkapi dengan *ethernet port*, Wi-Fi *(Wireless Fidelity)* 802.11n 2.4 Ghz, dan bluetooth 4.1. Pada input/output, Raspberry Pi 3 Model B mempunyai GPIO (*General Purpose Input Output*) sebanyak 40 pin selain itu terdapat beberapa port seperti port HDMI (*High Definition Multimedia Interface*), audio jack 3.5mm, empat port USB (*Universal Serial Bus*), CSI (*Camera Serial Interface*), dan DSI (*Display Serial Interface*).[13]

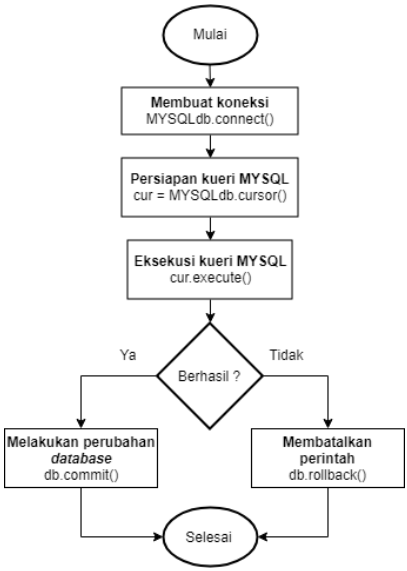
2.7 Modul MYSQLdb

Modul MYSQLdb merupakan antarmuka yang berfungsi untuk menghubungkan ke *database server* MYSQL dari Python. Modul MYSQLdb telah mendukung Python DB-API versi dua (merupakan standar antarmuka *database*) dan dibangun di atas MYSQL C API. Dengan menggunakan modul MYSQLdb kueri MYSQL dapat di jalankan pada program Python.[14]

Gambar 2.7 merupakan alur kerja sederhana menghubungkan antara *client* dan *server database* mengguakan MYSQLdb. Mula-mula program menjalankan perintah *MYSQLdb.connect()* berfungsi untuk membuat koneksi ke *server database*. Pada fungsi tersebut berisi parameter antaralain *host, user, password, database*, dan *port*. Program menjalankan perintah *cur=MYSQLdb.cursor()* digunakan untuk menyiapkan perintah kueri dalam bentuk MYSQL. Tahap selanjutnya adalah menjalankan program *cur.execute('berisi kueri MYSQL')* dan db.commit(). Bila terjadi kegagalan dalam menjalankan perintah, maka program menjalanperintah *db.rollback()* yang digunakan untuk membatalkan perintah. Lalu yang terakhir memutuskan koneksi dengan *server database*.

2.8 *Cloud Storage*

Cloud Storage merupakan media penyimpanan secara daring. Saat ini penggunaan *cloud storage* telah menjadi paradigma baru dalam dunia teknologi informasi seperti *internet of things*. Banyak data yang disimpan di *cloud storage* agar dapat diakses oleh pengguna dimana pun mereka berada. Saat ini telah banyak penyedia layanan *cloud storage* secara gratis maupun berbayar. Banyak perusahaan digital maupun bukan digital hingga pemerintahan te-



Gambar 2.7: Alur kerja MYSQLdb.

lah menggunakan penyimpanan cloud sebagai tempat penyimpanan data-data yang dikelola maupun sebagai *back up data*. Sesuai dengan penjelasan pada, dari segi privasi, *cloud storage* dibagi menjadi empat jenis yaitu:[15]

1. *Personal Cloud Storage*
   Storage jenis ini biasanya digunakan oleh perseorangan untuk *back up* data-data yang dimiliki. Beberapa smartphone bahkan telah menambahkan aplikasi *cloud* untuk mempermudah penggunanya ketika ingin melakukan *back up* terhadap datanya.
2. *Public Cloud Storage*
   Jenis *storage* ini sering digunakan oleh perusahaan yang memiliki jenis data tidak terstruktur. Perusahaan menyewa *cloud storage* dari provider *cloud* dan meminta untuk mengelola seluruh data yang dimiliki perusahaan. Selain itu jenis *storage* ini sering digunakan dalam hal berbagi secara publik.

3. *Private Cloud Storage*
   Jenis ini umumnya digunakan oleh perusahaan yang menginginkan fasilitas keamanan serta kecepatan lebih dibanding jenis *cloud storage* yang lain. Pada *storage* ini provider akan membentuk sebuah infrastruktur dalam pusat data perusahaan, dan keduanya akan berintegrasi didalamnya.
4. *Hybrid Cloud Storage*
   Merupakan gabungan antara *public* dan *private*. Data akan terbagi dua, yaitu data yang dianggap penting akan tersimpan dalam *private cloud storage* dan data lainnya di simpan dalam *public cloud storage*. Pada *storage* ini terdapat pilihan pengaturan data mana saja yang dijadikan *private* atau publik.

Sistem penyimpanan *cloud storage* dapat dianggap sebagai jaringan terdistribusi pusat data yang biasanya menggunakan teknologi *cloud computing* seperti virtualisasi dan menawarkan beberapa jenis antarmuka untuk menyimpan data.

Kelebihan *cloud storage* yaitu data dapat diakses dari berbagai lokasi (misalnya dari perangkat mobile) tanpa perlu menyamakan perangkat keras atau perangkat lunak yang sesuai, dan sinkronisasi fitur yang memungkinkan pelanggan untuk selalu memiliki akses ke versi terbaru dari perangkat (komputer personal, laptop, *smartphone*). Penggunaan *cloud storage* dapat dimanfaatkan sebagai layanan *Copy*, *Backup*, *Synchronization of Devices*, dan *Sharing Files*.

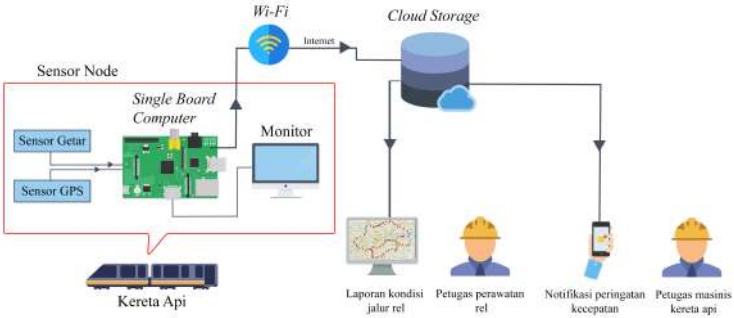
BAB 3
DESAIN DAN IMPLEMENTASI SISTEM

Penelitian ini dilaksanakan sesuai dengan desain sistem berikut dengan implementasinya. Desain sistem merupakan konsep dari pembuatan dan perancangan infrastruktur kemudian diwujudkan dalam bentuk blok-blok alur yang harus dikerjakan.

3.1 Desain Sistem

Pada sistem yang dibangun, digunakan sebuah sensor getar akselerometer untuk mendapatkan nilai getar saat kereta melaju dan sensor GPS untuk mendapatkan lokasi dan kecepatan aktual kereta api. Data-data tersebut divisualisasikan kedalam bentuk *marker* berwana pada peta digital berbasis web dan notifikasi peringatan dini batas kecepatan maksimal kereta api. Guna merealisasikan penelitian tugas akhir ini, dibuat sebuah desain sistem yang menggambarkan konsep dan kerja sistem.

Pada Gambar 3.1 diuraikan secara singkat yaitu mulainya dilakukan pengambilan data oleh *Single Board Computer* (SBC) yang terhubung dengan sensor GPS dan getaran. SBC melakukan penyimpanan data pada data lokal dan divisualisasikan dengan tam-

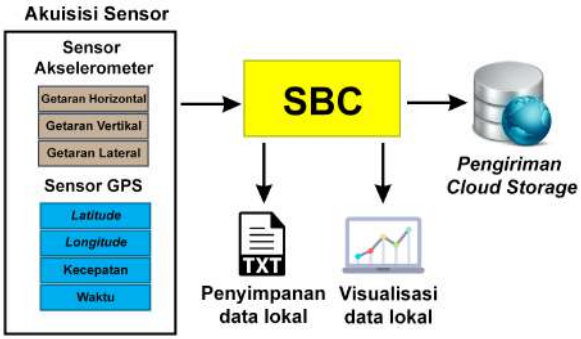


Gambar 3.1: Gambaran umum kerja sistem.

pilan interaktif pada monitor. Jika terdapat koneksi ke *server* maka dilakukan penyimpanan data dari data lokal ke *cloud storage*. Data disimpan kedalam bentuk *cloud storage* yang dapat diakses kembali melalui guna laporan indeks kondisi rel kereta api berbasis web dan notifikasi batas kecepatan kereta api. Terkait gambaran umum kerja sistem, sistem yang dibuat terbagi menjadi empat bagian yaitu sensor *node*, pengiriman data ke *cloud storage*, visualisasi data pada *sensor node* dan laporan nilai indeks rel kereta api berbasis web, dan sistem notifikasi peringatan dini batas kecepatan maksimal kereta api.

3.1.1 Sensor *Node*

Dijelaskan pada Gambar 3.2, sensor *node* terdiri dari sensor akselerometer digital (akusisi nilai getaran kereta), sensor GPS (akuisisi lokasi berupa koordinat *latitude* dan *longitude* aktual kereta api, waktu, dan kecepatan aktual), dan Raspberry pi 3 sebagai *Single Board Computer* (SBC).



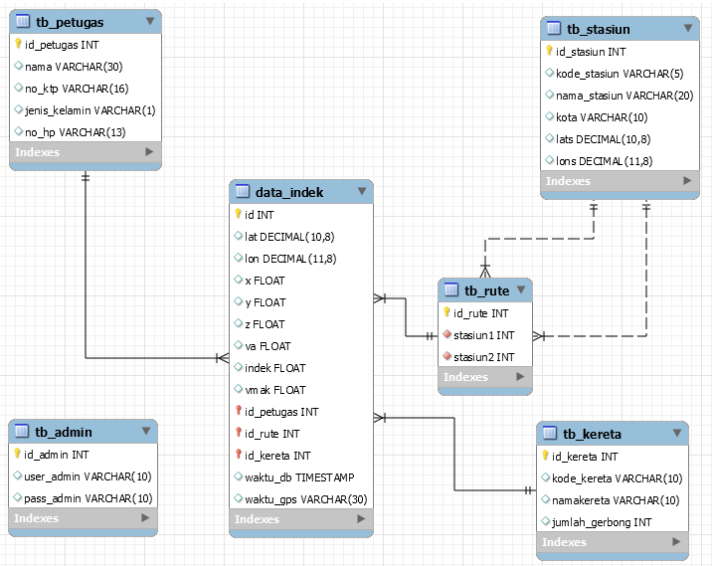
Gambar 3.2: Skema alur kerja sensor *node*.

Ada empat jenis data yang akan diakuisisi, yaitu data nilai getaran kereta api, data kecepatan aktual, waktu pengambilan data dan data lokasi. Data getaran diperoleh dengan menggunkan sensor akselerometer digital dengan tipe ADXL345. Ada tiga getaran yang diakuisisi berdasarkan arah getaran yaitu getaran horizontal (sumbu X), getaran lateral (sumbu Y), dan getaran vertikal (sumbu Z). Sedangkan untuk mengetahui lokasi, kecepatan aktual kereta api,

dan waktu pengambilan sensor yang digunakan adalah GPS bertipe GPS *Ultimate* versi tiga. Data-data yang diperoleh, dilakukan penyimpanan dan pengiriman data lokal oleh SBC. Penyimpanan dilakukan oleh SBC pada sebuah *local file* bertipe teks yang menyimpan seluruh data yang telah diakuisisi oleh sensor getar dan GPS. Data yang telah disimpan oleh SBC, divisualisasikan kedalam bentuk grafik meliputi grafik nilai getaran dan kecepatan kereta api. kemudian guna pengiriman data lokal ke *cloud storage*, SBC telah dilengkapi modul Wi-Fi sehingga pengiriman dapat dilakukan dengan menyambung Wi-Fi SBC dengan *Access Point* (AP) yang sebelumnya telah terhubung dengan internet.

3.1.2 Desain *Database* Pada *Cloud Storage*

Desain *database* dapat dijelaskan pada Gambar 3.3 dibagi menjadi enam tabel. Pada sistem ini, *database* digunakan untuk menyimpan data seluruh data yang telah dikirim dari sensor *node* ke *cloud storage*. *Database* yang digunakan jenis *database* relasi ERD



Gambar 3.3: Desain *Entity Relation Database*.

(*Entity Relation Database*), yang mana antara tabel dapat berelasi dengan tabel lain dan tabel tersebut tidak bersifat fleksibel. Desain *database* ini berguna untuk mengurangi tingkat redudansi data, salah satunya dengan cara normalisasi *database*.

Tabel data_indek adalah tabel data indeks dengan struktur tabel seperti pada Tabel 3.1. Tabel data indeks memiliki urutan struktur kolom antara lain *id*(*primary key*) sebagai lokasi penyimpanan nomor identitas setiap data indeks, lat untuk menyimpan titik koordinat *latitude* GPS, lon untuk menyimpan titik koordinat *longitude* GPS, x untuk menyimpan nilai rata-rata getaran horizontal (sumbu x), y untuk menyimpan nilai rata-rata getaran lateral (sumbu y), dan z untuk menyimpan nilai rata-rata getaran vertikal (sumbu z), va digunakan untuk menyimpan kecepatan aktual kereta api, indeks digunakan untuk menyimpan hasil nilai kondisi rel kereta api, vmak digunakan untuk menyimpan nilai batas kecepatan maksimal, id_petugas sebagai id dari petugas pada tabel data indeks, id_kereta sebagai id dari kereta api pada tabel data indeks, id_stasiun_berangkat sebagai id stasiun awal keberangkatan kereta pada tabel data indeks, id_stasiun_tujuan sebagai id stasiun tujuan kereta api pada tabel data indeks, dan waktu digunakan untuk menyimpan waktu data masuk ke *database server*.

Berdasarkan Tabel 3.1, terdapat 14 *field* pada struktur tabel data indeks yakni id menggunakan tipe data *integer*, lat menggunakan tipe data *decimal* dengan *point value* 10,8, lon menggunakan tipe data *decimal* dengan *point value* 11,8, x dengan tipe data *float*, y menggunakan tipe data float, z dengan tipe data *float*, va dengan tipe data *float*, indek menggunakan tipe data *float*, vmak menggunakan tipe data *float*, id_petugas menggunakan tipe data *integer*, id kereta menggunakan tipe data *integer*, id_stasiun_berangkat menggunakan tipe data *integer*, id_stasiun_tujuan menggunakan tipe data *integer*, dan waktu menggunakan tipe data *timestamp*.

Tabel tb_petugas adalah tabel petugas dengan struktur tabel seperti pada Tabel 3.2. Tabel petugas memiliki urutan struktur kolom antara lain id_petugas (*primary key*) untuk lokasi penyimpanan nomor tabel petugas, nama untuk menyimpan nama petugas, no_ktp untuk menyimpan informasi nomor Kartu Tanda Penduduk (KTP) petugas, jenis_kelamin digunakan untuk menyimpan infor-

Tabel 3.1: Struktur tabel data indeks.

Nama Field	Tipe Data	Ukuran	Keterangan
id	*int*		id data indeks
lat	*decimal*	10,8	latitude
lon	*decimal*	11,8	longitude
x	*float*		getaran horizontal
y	*float*		getaran lateral
z	*float*		getaran vertikal
va	*float*		kecepatan aktual
indek	*float*		nilai indek rel
vmak	*float*		kecepatan maksimal
id_petugas	*int*		id petugas
id_kereta	*int*		id kereta api
id_stasiun_berangkat	*int*		id stasiun awal
id _stasiun_tujuan	*int*		id stasiun akhir
waktu	*timestamp*		waktu data masuk
waktu_gps	*varchar*	30	waktu diterima gps

masi jenis kelamin petugas, dan no_hp digunakan untuk menyimpan informasi nomor *handphone* petugas.

Berdasarkan Tabel 3.2 terdapat enam *field* pada struktur tabel petugas yakni id_petugas menggunakan tipe data *integer*, nama menggunakan tipe data *varchar* dengan panjang 30, no_ktp menggunakan tipe data *varchar* dengan panjang 16, jenis_kelamin menggunakan tipe data *varchar* dengan ukuran satu, dan no_hp menggunakan tipe data *varchar* dengan ukuran 13.

Tabel 3.2: Struktur tabel petugas.

Nama Field	Tipe Data	Ukuran	Keterangan
id_petugas	*int*		id petugas
nama	*varchar*	30	nama petugas
no_ktp	*varchar*	16	nomor kependudukan
jenis_kelamin	*varchar*	1	jenis kelamin pegawai
no_hp	*varcahr*	13	nomor *handphone*

Tabel tb_stasiun adalah tabel stasiun dengan struktur tabel seperti pada Tabel 3.3. Tabel stasiun memiliki urutan struktur kolom antara lain id_stasiun *primary key* untuk lokasi penyimpanan nomor identitas stasiun, kode_stasiun digunakan untuk menyimpan kode dari stasiun, nama_stasiun sebagai label nama dari stasiun, kota untuk label nama kota stasiun berada, latitude, dan longitude stasiun.

Berdasarkan Tabel 3.3 terdapat enam *field* pada struktur tabel stasiun yakni id_petugas menggunakan tipe data *integer*, kode_stasiun menggunakan tipe data *varchar* dengan panjang lima, nama_stasiun menggunakan tipe data *varchar* dengan panjang 20, dan kota menggunakan tipe data *varchar* dengan panjang 10, *latitude* dan *longitude* menggunakan tipe data *decimal*.

Tabel 3.3: Struktur tabel stasiun.

Nama Field	Tipe Data	Ukuran	Keterangan
id_stasiun	*int*		id stasiun
kode_stasiun	*varchar*	5	kode stasiun
nama_stasiun	*varchar*	20	nama stasiun
kota	*varchar*	10	kota stasiun
lat	*decimal*	10,8	latitude patok
lon	*decimal*	11,8	longitude patok

Tabel tb_kereta adalah tabel yang berisi informasi seputar kereta api. Tabel kereta memiliki struktur tabel seperti pada Tabel 3.4. Tabel kereta memiliki urutan struktur kolom antara lain id_kereta *primary key* untuk lokasi penyimpanan nomor identitas kereta api, kode_kereta digunakan untuk label kode pada kereta api, nama_kereta digunakan untuk label nama kereta api, jumlah gerbong digunakan untuk menyimpan data jumlah_gerbong pada rangkaian kereta api.

Berdasarkan Tabel 3.4 terdapat empat *field* pada struktur tabel petugas yakni id_kereta api menggunakan tipe data *integer*, kode_kereta menggunakan tipe data *varchar* dengan panjang 10, nama kereta menggunakan tipe data *varchar* dengan panjang 10, dan jumlah_gerbong menggunakan tipe data *integer*.

Tabel 3.4: Struktur tabel kereta api.

Nama Field	Tipe Data	Ukuran	Keterangan
id_kereta	*int*		nomor id kereta
kode_kereta	*varchar*	10	kode kereta api
namakereta	*varchar*	10	nama kereta api
jumlah_gerbong	*int*		jumlah gerbong kereta api

Tabel admin memiliki urutan struktur kolom seperti pada Tabel 3.5. antara lain id_admin *primary key* untuk menyimpan nomor tabel, user_admin digunakan untuk menyimpan nama *user* sebagai admin, pass_admin untuk menyimpan kata sandi admin.

Berdasarkan Tabel 3.5 terdapat tiga *field* pada struktur tabel admin yakni id_admin menggunakan tipe data *integer*, dan user_admin dan pass_admin menggunakan tipe data *varchar* dengan panjang 10.

Tabel 3.5: Struktur tabel admin.

Nama Field	Tipe Data	Ukuran	Keterangan
id_admin	*int*		nomor id admin
user_admin	*varchar*	10	nama *user* admin
pass_admin	*varchar*	10	kata sandi admin

Tabel tb_rute adalah tabel rute kereta api dengan struktur tabel seperti pada Tabel 3.6. Tabel rute memiliki urutan struktur kolom antara lain id_rute (*primary key*) untuk menyimpanan nomor id rute, stasiun1 sebagai stasiun keberangkatan, dan stasiun2 sebagai stasiun tujuan.

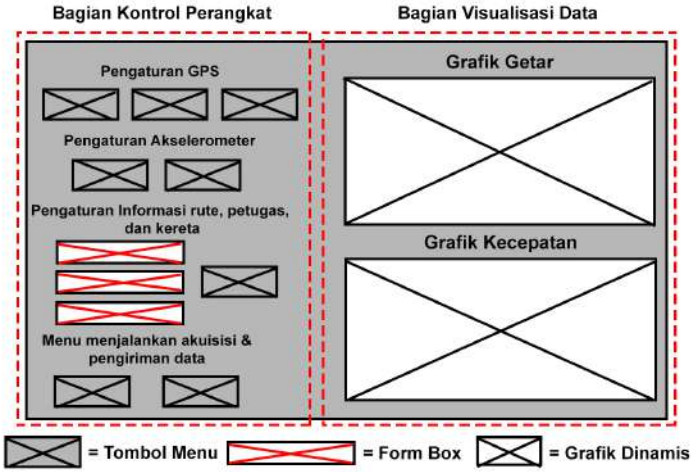
Berdasarkan Tabel 3.6 terdapat tiga *field* antara lain id_rute, stasiun1, dan stasiun2 menggunakan tipe data *integer*.

Tabel 3.6: Struktur tabel rute.

Nama Field	Tipe Data	Ukuran	Keterangan
id_rute	*int*		nomor id rute
stasiun1	*int*		stasiun keberangkatan
stasiun2	*int*		stasiun tujuan

3.1.3 *User Interface*(UI)

User interface (UI) dibagi menjadi tiga yaitu UI pada sensor *node*, UI laporan nilai indeks rel kereta api berbasis web berupa peta digital, dan UI aplikasi notifikasi batas kecepatan maksimal. Sensor *node* dilengkapi dengan layar sentuh dengan UI dibangun menggunakan Tkinter sebagai media kontrol perangakat pemantau kondisi rel kereta api. Pada sensor *node* data yang divisualisasikan adalah data aktual yang telah diakusisi oleh sensor getar dan GPS kedalam bentuk grafik bergerak. Sehingga petugas pengoperasi sensor *node* dapat mengoperasikan perangkat dengan menyentuh layar monitor dan sekaligus memantau kondisi aktual rel kereta api pada monitor sensor *node*. Berikut adalah desain visualisasi dan UI pada sensor *node* tertera pada Gambar 3.4.



Gambar 3.4: Tampilan desain UI pada sensor *node*.

Guna laporan nilai indeks kondisi rel kereta api, data divisualisaikan dengan *marker* berwarna pada peta digital berbasis web menggunakan *Google Maps API* sebagai sumber tampilan. Data-data yang ditampilkan pada UI peta digital adalah data yang berasal dari *cloud storage* yang sebelumnya dikirim oleh sensor *node*. Desain UI pada peta digital berbasis web tertera pada Gambar 3.5.

Gambar 3.5: Tampilan desain UI pada peta digital berbasis web.

Beberapa fitur yang digunakan dari *Google Maps API* yang kemudian dimodifikasi antara lain:

1. Fitur *marker* digunakan sebagai penanda data *node* yang telah tersimpan pada *cloud storage*.
2. Fitur geolokasi digunakan untuk menampilkan posisi *marker* pada peta digital sesuai data lokasi pengambilan yang telah tersimpan di *cloud storage*.

Pada Tabel 3.7 merupakan parameter warna *marker*, diantaranya adalah warna hijau menunjukkan besar nilai indeks lebih besar sama dengan 100 yang menandakan kondisi rel baik, warna *marker* kuning menunjukan besar nilai indeks sebesar 85 hingga 99 yang menandakan kondisi rel kurang baik, warna *marker* jingga menunjukkan nilai indeks sebesar 70 hingga 84 yang menandakan kondisi rel buruk, dan warna *marker* merah menunjukkan nilai indeks sebesar kurang dari 69 yang menandakan kondisi rel sangat buruk.

Tabel 3.7: Parameter warna *marker*.

Warna	Besar Indeks (N)	Kondisi Rel Kereta Api
Hijau	$N \geq 100$	Baik
Kuning	$100 > N \geq 85$	Kurang Baik
Jingga	$85 > N \geq 70$	Buruk
Merah	$70 > N$	Sangat Buruk

Guna informasi notifikasi batas kecepatan maksimal, data divisualisasikan kedalam bentuk aplikasi peta digital berbasis Android. Dilihat pada Gambar 3.6, tampilan desain aplikasi dibagi menjadi dua bagian. Bagian pertama adalah peta digital dan bagian kedua adalah informasi aktual. Bagian peta digital terdiri dari dua *marker* dengan warna yang berbeda yakni *marker* warna biru menunjukan lokasi aktual kereta api dan *marker* warna merah menunjukan lokasi batas kecepatan maksimal yang akan dilalui oleh kereta api. Bagian informasi aktual bertujuan memberikan informasi aktual meliputi kecepatan aktual kereta api, batas kecepatan, dan jarak.

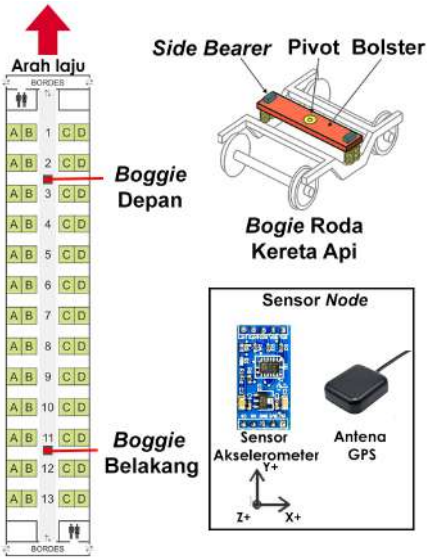
Gambar 3.6: Tampilan desain aplikasi notifikasi batas kecepatan maksimal.

3.2 Implementasi Sistem

Secara keseluruhan, alur implementasi sistem terbagi menjadi enam, yaitu pemasangan sensor *node*, akusisi data pada sensor *node*, penyimpanan ke *cloud storage*, *user interface* sensor *node*, laporan nilai indek rel berbasis web, dan aplikasi notifikasi batas kecepatan.

3.2.1 Pemasangan Sensor *Node* di Kereta Api

Gambar 3.7 merupakan skema pemasangan sensor *node*. Pemasangan sensor *node* dilakukan di dalam kereta penumpang pada posisi depan atau belakang kereta penumpang yang dekat dengan sambungan *bogie* (gabungan roda kereta). Pada pemasangan sensor *node*, sensor akselerometer dipasang dengan merekatkan sensor dengan sambungan *pivot bogie* yang berada di lantai kereta penumpang. Pada pemasangan sensor akselerometer, sumbu $Z+$ sensor menghadap berlawanan arah gravitasi guna mendeteksi getaran vertikal pada kereta api. Sedangkan sumbu $Y+$ sensor menghadap searah dengan arah laju kereta api guna mendeteksi getaran lateran kereta api. Sehingga sumbu X pada sensor akselerometer berfungsi untuk mendeteksi getaran horizontal kereta api. Dan pemasangan antena GPS dilakukan di dalam kereta penumpang. Posisi pemasangan antena GPS harus menghadap kearah atas agar sensor GPS dapat menerima informasi lokasi dan kecepatan kereta api.



Gambar 3.7: Skema pemasangan sensor *node* di *bogie* kereta penumpang.

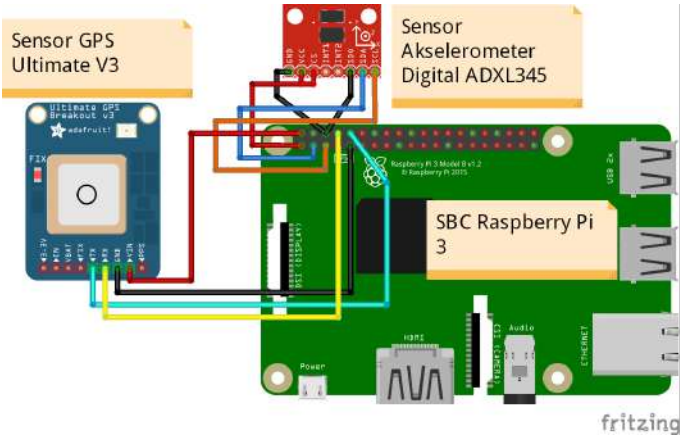
Gambar 3.8 merupakan implementasi pemasangan sensor *node* di lantai kereta penumpang. Pada implementasi pemasangan sensor *node* didalam kereta penumpang, sumber listrik yang digunakan berasal dari kereta penumpang yang tersedia di setiap tempat duduk penumpang.

Gambar 3.8: Implementasi pemasangan sensor *node* di kereta penumpang.

3.2.2 Akusisi pada Sensor *Node*

Konfigurasi pemasangan sensor akselerometer dan GPS pada SBC, ditunjukan pada Gambar 3.9. Sensor *node* tersusun atas beberapa komponen yakni sensor akselerometer digital ADXL345 dan sensor GPS *Ultimate* yang terhubung dengan SBC Raspberry Pi 3. Sensor ADXL345 dihubungkan ke Raspberry pi melalui pin GPIO (*General Purpose Input Output*) menggunakan komunikasi I^2C dimana untuk menggunakan komunikasi I^2C pin $\overline{CS}$ dan pin 3.3v sensor akselerometer dihubungkan ke pin 3.3V SBC, pin SDO dan pin GND sensor akselerometer dihubungkan ke GND SBC, pin SDA pada sensor akselerometer dihubungkan ke pin 5 SBC, dan pin SCL sensor akselerometer dihubungkan ke pin 7 SBC. Sedangkan GPS *Ultimate* dihubungkan ke SBC menggunkan komunikasi seri-

al UART (*Universal Asynchronous Receiver Transmitter*). Pada pemasangan sensor GPS dengan SBC, pin vin sensor GPS dihubungkan dengan pin 5V SBC, pin GND sensor GPS dihubungkan dengan pin GND SBC, pin TX (*transmitter*) sensor GPS dihubungkan dengan pin RX (*receiver*) SBC, dan begitu sebaliknya pin RX sensor GPS dihubungkan dengan pin TX SBC. Program sensor pada SBC menggunakan beberapa *library* diantaranya adxl345python untuk *library* sensor akselerometer dan GPSController untuk *library* sensor GPS.



Gambar 3.9: Skema konfigurasi sensor *node*.

Proses awal sebelum akusisi data yakni proses kalibrasi pada setiap sumbu sensor akselerometer. Kalibrasi sensor akselerometer dilakukan hanya sekali setiap pemasangan baru pada kerata api. Proses berikutnya menunggu sensor GPS mendapatkan lokasi. Sensor GPS memerlukan waktu kurang lebih 1 menit dari kondisi GPS yang baru dihidupkan untuk mendapatkan lokasi dengan status minimal *FIX-2D* (*latitude* dan *longitude*). Setelah sensor GPS mendapatkan sinyal, proses akuisisi oleh sensor akselerometer dan GPS dapat dimulai. Hasil dari proses akusisi sensor berupa data mentah yang kemudian diproses dengan menggunakan *library* masing-masing sensor. Data disusun berupa deret data dengan tipe *string* dengan ditambahkan *delimiter* koma (,) guna membedakan

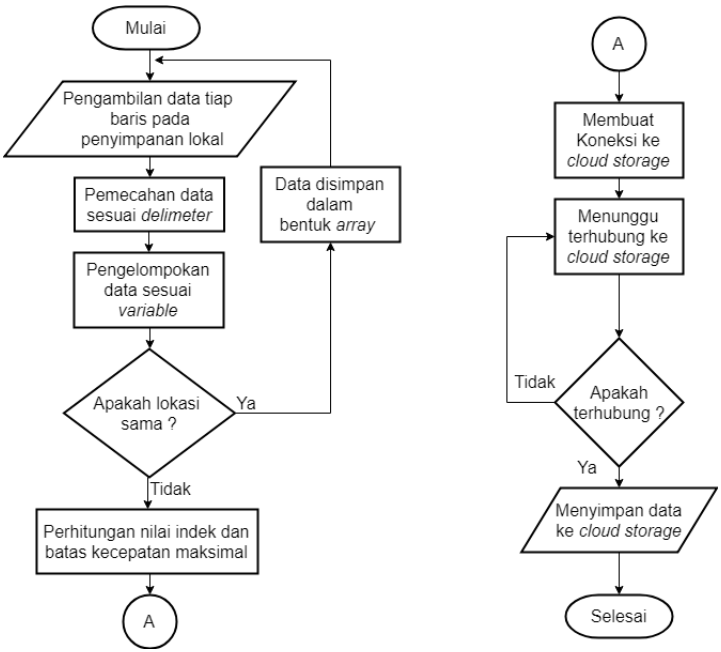
data satu dengan yang lain. Kemudian data disimpan ke data lokal SBC dalam bentuk *file* bertipe teks yang nantinya dapat diakses untuk dikirim ke *cloud storage* dan visualisasi data. Akusisi oleh sensor *node* dilakukan setiap 0,5 detik. Alur kerja akuisisi data oleh sensor dijelaskan pada Gambar 3.10.

Gambar 3.10: Diagram alir akusisi data ke SBC.

3.2.3 Penyimpanan ke *Cloud Storage*

Pada penyimpanan ke *cloud storage* dikirim melalui sambungan GPRS (*General Packet Radio Service*) yang terhubung ke SBC melalui Wi-Fi *Wireless Fidelity*. Penyimpanan tersebut menggunakan MYSQLdb untuk berkomunikasi langsung dengan *server database* MYSQL. Selain terdapat proses penyimpanan ke *cloud storage*, program juga dibuat untuk menghitung nilai rata-rata data getaran dan kecepatan aktual pada lokasi yang sama guna pencarian nilai indeks dan batas kecepatan maksimal.

Gambar 3.11 menjelaskan alir kerja penyimpanan data lokal ke *cloud storage*. Pada proses pertama ialah membuka data yang tersimpan pada penyimpanan lokal. Data dibaca setiap baris hingga data terakhir yang tersimpan. Data lokal yang telah dibaca setiap baris, dipisah dan dikelompokan sesuai dengan masing-masing variabel. Apabila terdapat lokasi yang sama, maka nilai data getar disimpan di *array* sementara dan kemudian dilakukan perhitungan nilai rata-rata guna perhitungan nilai indek dan batas kecepatan maksimal. Kemudian sensor *node* membuat koneksi pengirim-



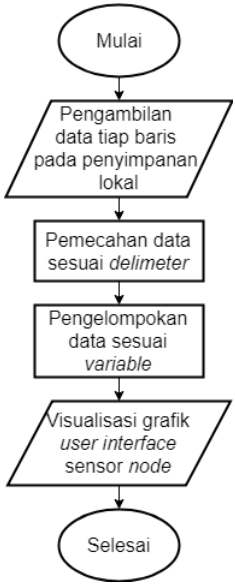
Gambar 3.11: Diagram alir penyimpanan ke *cloud storage*.

an menggunakan modul MYSQLdb yang bekerja pada mesin lokal menggunakan standar UNIX *Socket*. Koneksi tersebut berisi informasi mengenai pengiriman ke *cloud storage* seperti nama *host*, *user name*, *password*, nama *database*, dan *port*. Jika koneksi berhasil terhubung dengan *cloud storage*, data dikirim sesuai dengan perintah

kueri MYSQL yang telah dibuat. Proses pengiriman data dilakukan dengan interval 100 milidetik setiap pengiriman.

3.2.4 *User Interface* pada Sensor *Node*

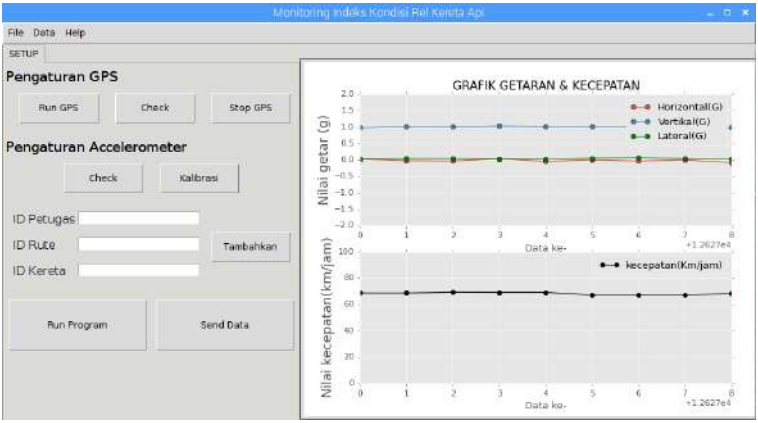
Tampilan *user interface* pada sensor *node* memiliki dua fungsi yaitu fungsi pemantauan dan fungsi kontrol. Fungsi pemantauan terdiri dari grafik-grafik yang memvisualkan data aktual sensor. Sedangkan fungsi kontrol digunakan untuk mempermudah petugas dalam menjalankan alat. Gambar 3.12 menjelaskan cara kerja visualisasi data pada *user interface* sensor *node*.



Gambar 3.12: Diagram alir visualisasi data pada *user ineterface* sensor *node*.

Pada proses visualisasi data ke grafik adapaun beberapa proses yakni membaca tiap baris data yang tersiman pada data lokal. Data yang telah dibaca, kemudian dilakukan pemecahan (*split data)* sesuai delimeter yang telah ditentukan. Dari hasil pemecahan data, didapat data getaran kereta api dan kecepatan aktual yang nilainya divisualisasikan ke tampilan UI sensor *node*.

User interface dibuat dari *layout* yang dibangun menggunakan *library* Tkinter dengan bahasa Python. Gambar 3.13 merupakan implementasi tampilan utama user interface yang dibuat. *User interface* tersebut dibuat seinteraktif mungkin agar dapat dengan mudah dioperasikan dan dipantau oleh petugas perwatan rel kereta api.



Gambar 3.13: Implementasi *User Interface* pada sensor *node*.

Pada *user interface* terdapat beberapa komponen utama guna pengoperasian alat dan pemantauan kondisi rel kereta api yaitu:

1. Kontrol alat - Kontrol alat mempresentasikan media yang digunakan untuk mengoperasikan alat. Komponen kontrol alat meliputi *Run GPS* berfungsi sebagai inisialisasi pemasangan antara *UART GPS* dengan *GPS Daemon*, *Check* berfungsi sebagai pemeriksa GPS telah menerima informasi lokasi, *Stop GPS* berfungsi untuk memutus komunikasi *UART GPS* dengan *GPS Daemon*, kalibrasi berfungsi sebagai memulai proses kalibrasi sensor getar, *check* pada getar berfungsi sebagai memeriksa sensor getar telah terhubung dengan SBC, *run program* berfungsi untuk menjalankan akusisi sensor node dan visualisasi data aktual, *send data* berfungsi untuk menyimpan data dari data lokal ke *cloud server*, dan pengisian informasi id petugas, id kereta, id rute.

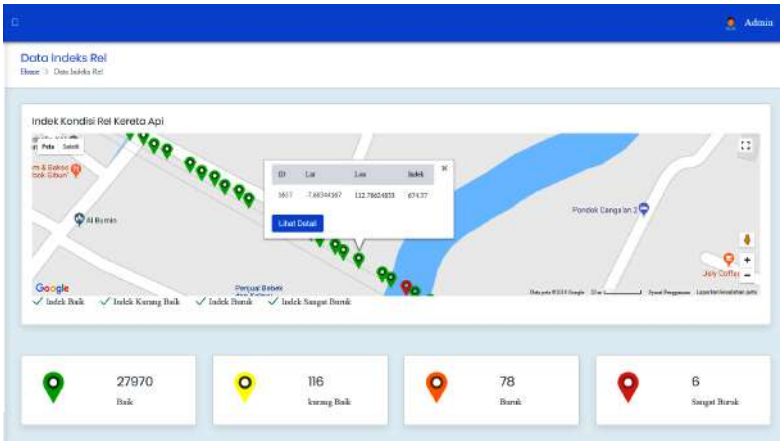
2. Pemantauan aktual - Pemantaun aktual mempresentasikan visualisasi data yang telah diakusisi oleh sensor *node* kedalam bentuk grafik garis. Terdapat dua jenis data yang divisualisasikan yaitu data getaran dan kecepatan aktual. data getaran meliputi getaran vertikal, getaran horizontal, dan getaran lateral. Sumbu X pada grafik getaran merupakan urutan data getaran dan sumbu Y merupakan nilai keluaran getaran dalam satuan g (gravitasi). Masing-masing jenis getaran ditandai dengan warna yang berbeda seperti getaran horizontal dengan warna merah, getaran vertikal dengan warna biru, dan getaran lateral dengan warna hijau. Pada grafik kecepatan, sumbu X merupakan urutan data kecepatan sedangkan sumbu Y adalah nilai kecepatan dengan satuan km/jam. kecepatan aktual divisualisasikan kedalam gravik berwarna kuning.
3. Blok menu - Blok menu berisi menu *file*, menu data, dan menu *help* yang digunakan untuk menunjang pemantauan dan pengoperasian sensor *node*.

3.2.5 Laporan Nilai Indek Rel Berbasis Web

Pembuatan laporan nilai indeks rel kereta api digunakan untuk memvisualisasikan laporan kondisi rel kereta api kepada petugas perawatan rel. Sistem pelaporan nilai indeks rel kereta api dibuat dengan pengambilan data-data sensor dari *cloud storage* yang telah menyimpan data kondisi rel kereta api dari sensor *node*. Data yang diambil kemudian dikelompokkan berdasarkan nilai indeks jalur rel sesuai dengan Tabel 2.3 yakni kualifikasi nilai indeks rel. Data yang telah dikelompokkan, kemudian disajikan kedalam bentuk peta digital dengan ditambahkan *marker* berwarna guna memvisualisasikan kondisi jalur rel kereta api berdasarkan lokasi saat akuisisi data. Warna pada *marker* mempresentasikan nilai dari nilai indeks rel kereta api.

Gambar 3.14 merupakan hasil implementasi laporan nilai indeks rel berbasis web yang dapat diakses oleh petugas perawatan rel. Laporan nilai indeks rel berbasis web dibuat menggunakan pemrograman PHP (*HyperText Preprocessor*) sebagai *back end* dan divisualkan dalam bentuk peta digital menggunakan API (*Application Programming Interface*) *Google Maps*. Informasi laporan nilai

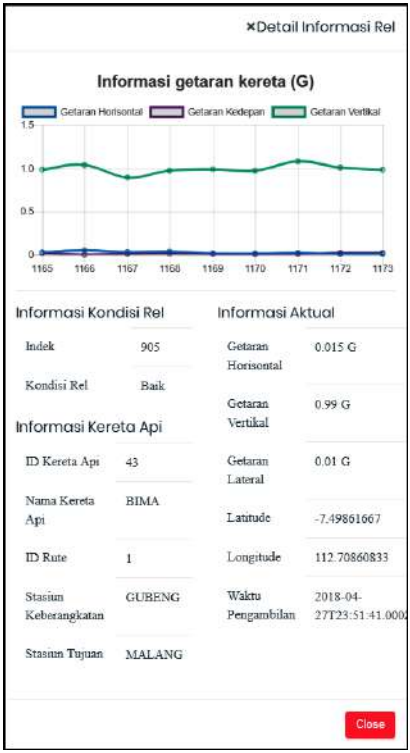
indeks jalur rel kereta api dapat dilihat dengan cara memilih salah satu *marker* yang nantinya akan muncul sebuah kota informasi (*info box*) yang berisi beberapa informasi antara lain informasi ID *marker*, nilai indeks jalur rel kereta api, informasi lokasi *marker* (*latitude* dan *longitude*), dan menu informasi detail. Pada bagian bawah peta digital laporan nilai indeks rel berbasis web menyajikan informasi jumlah data yang telah disimpan di *cloud storage* berdasarkan kualifikasi kondisi rel kereta api.



Gambar 3.14: Visualisasi laporan nilai indeks dalam bentuk *marker* berwarna pada peta digital berbasis web.

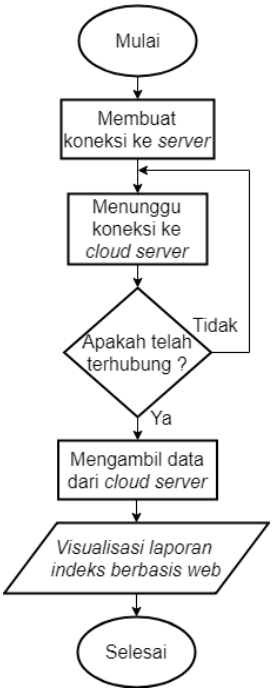
Menu informasi detail pada laporan nilai indeks berbasis web terdiri dari beberapa bagian informasi antara lain informasi grafik getaran kereta api, informasi kondisi rel, informasi kereta api, dan informasi aktual. Informasi kondisi rel meliputi beberapa komponen seperti nilai indeks rel dan jenis kondisi rel kereta api berdasarkan parameter nilai indeks rel. Informasi aktual berisi informasi seperti nilai getaran horizontal (X), nilai getaran vertikal (Z), nilai getaran lateral (Y), *latitude*, *longitude*, dan waktu saat saat berlangsungnya proses akuisisi data. Dan informasi kereta api berisi informasi terkait kereta api yang digunakan saat dilakukan pengukuran kondisi rel kereta api. Informasi kereta api meliputi id kereta api, nama kereta api, id rute perjalanan kereta api, stasiun keberangkatan,

dan stasiun tujuan. Hasil implementasi menu informasi detail pada laporan nilai indeks berbasis web dapat dilihat pada Gambar 3.15.



Gambar 3.15: Visualisasi laporan informasi detail berbasis web.

Pada proses laporan nilai indeks rel berbasis web, *web client* membuat koneksi ke *cloud storage* Jika koneksi berhasil terhubung, sistem mengambil data dari *cloud storage*. Data yang diambil merupakan data yang diinginkan oleh pengguna (petugas perawatan rel kereta api). Program memilih sesuai dengan kueri yang diminta. Data yang telah didapatkan, divisualisasikan kedalam laporan kondisi berbasis web. Proses kerja pembuatan laporan nilai indeks rel berbasis web dijelaskan pada sebuah diagram alir pada Gambar 3.16



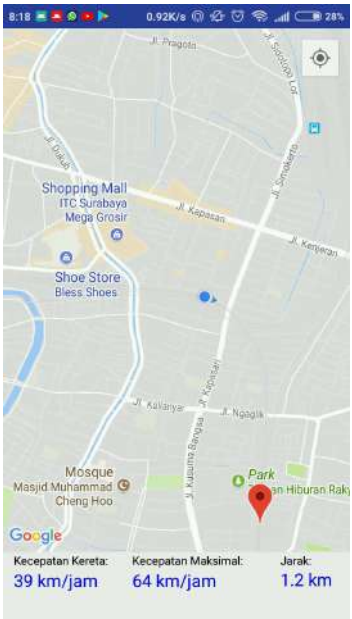
Gambar 3.16: Diagram alir laporan indeks berbasis web.

3.2.6 Aplikasi Notifikasi Batas Kecepatan Maksimal

Pembuatan aplikasi notifikasi batas kecepatan maksimal berfungsi untuk memberi peringatan dini kepada masinis agar dapat mengatur kecepatan kereta api sesuai batas kecepatan maksimal yang telah ditentukan. Nilai batas kecepatan maksimal didapatkan dengan mengambil nilai indeks kondisi rel kereta api, nilai kecepatan aktual, dan informasi lokasi titik batas kecepatan maksimal kereta api.

Pembuatan aplikasi notifikasi batas kecepatan maksimal dibangun dengan menggunakan *platform* Android dan*API Google Maps*. Aplikasi notifikasi batas kecepatan maksimal memiliki beberapa fi-

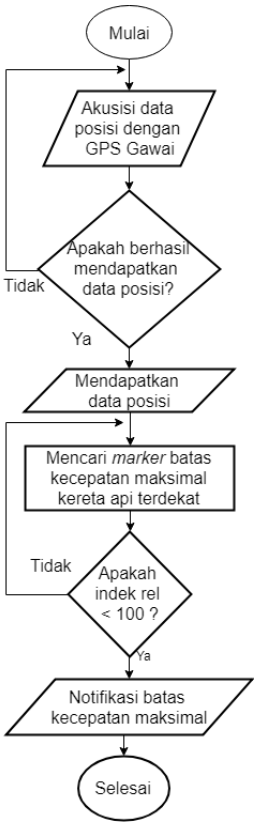
tur seperti mengetahui posisi aktual kereta api, kecepatan aktual kereta api, dan titik lokasi batas kecepatan maksimal yang akan dilewati oleh kereta api. Dengan aplikasi notifikasi batas kecepatan maksimal, masinis cukup menjalankan aplikasi yang telah terpasang digawai. Gambar 3.17 merupakan hasil implemetasi aplikasi notifikasi batas kecepatan maksimal kereta api.



Gambar 3.17: Implementasi aplikasi notifikasi batas kecepatan maksimal kereta api.

Pada aplikasi notifikasi batas kecepatan maksimal, *marker* batas kecepatan maksimal akan muncul saat kereta api akan mendekati *marker* batas kecepatan maksimal yang memiliki nilai indeks kondisi rel kereta api kurang dari 100. Diagram alir sistem kerja aplikasi batas kecepatan maksimal kereta api dapat dilihat pada Gambar 3.18. Nilai batas kecepatan maksimal yang akan dilewati oleh kereta api, diperoleh dengan melakukan *request* ke *cloud storage* menggunakan metode GET pada *port* 80. *Request* data dilakukan dengan cara me-

nyisipkan data ke alamat URL (*Uniform Resource Locator*) *cloud storage* yang dituju. Data yang disisipkan berupa data posisi aktual kereta api berupa nilai *latitude* dan *longitude*. Data yang telah diunggah, aplikasi akan mendapat data *respone* dari *cloud storage*. *Respone* data yang didapat menggunakan format JSON *(JavaScript Object Notation*) terdiri dari data lokasi *latitude* dan *longitude* titik *marker*, batas kecepatan maksimal, nilai indeks rel, dan jarak aktual antara kereta api dengan titik batas kecepatan maksimal.



Gambar 3.18: Diagram alir aplikasi notifikasi batas kecepatan maksimal.

Gambar 3.19 menjelaskan proses *request* data ke *cloud storage* dengan metode *GET* menggunakan protokol HTTP (*Hypertext Transfer Protocol*) dengan data lokasi aktual kereta api yang disisipkan pada URL dan hasil *respone* data dalam format JSON.

Gambar 3.19: GET *request* dan *response* data.

BAB 4
PENGUJIAN DAN ANALISA

Pada bab ini dipaparkan hasil pengujian serta analisa dari desain sistem dan implementasi. Pengujian dibagi menjadi lima bagian antara lain:

1. Pengujuan *user interface* - meliputi pengujian *user interface* pada laporan nilai indeks rel berbasis web dan *user ineterface* pada sensor *node*.
2. Pengujian perangkat yang digunakan - meliputi pengujian akurasi lokasi perangkat GPS, akurasi kecepatan perangkat GPS, dan akurasi sensor akselerometer.
3. Pengujian interval waktu penyimpanan ke *Database* - meliputi pengujian waktu rata-rata jeda pengiriman ke *database*.
4. Pengujian sistem peringatan dini - meliputi pengujian notifikasi batas kecepatan maksimal pada aplikasi.
5. Pengujian validasi data - meliputi pengujian validasi nilai indeks rel dari hasil pengukuran dengan berbeda kereta dan kecepatan.

Dengan dilaksanakannya beberapa pengujian tersebut, sehingga dapat ditarik kesimpulan dari pelaksanaan tugas akhir ini.

4.1 Pengujian *User Interface*

Pada pengujian *user interface*, terdapat dua pengujian yakni pengujian *user interface* pada laporan nilai indeks rel pada peta digital dan pengujian *user interface* pada sensor *node*.

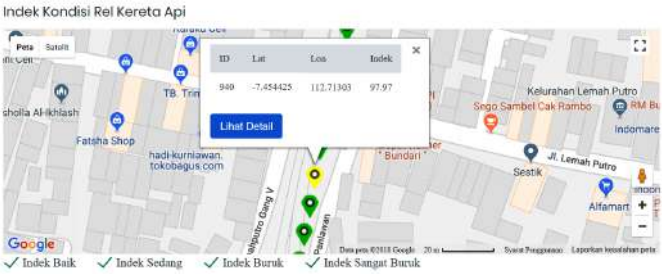
4.1.1 *User Interface* Laporan Nilai Indeks Rel

Pada bagian ini, dilakukan pengujian untuk melihat apakah warna *marker* yang ditampilkan oleh peta digital telah sesuai dengan dengan parameter nilai indeks rel yang telah ditentukan. Hasil yang didapatkan pada pengujian ini adalah warna *marker* pada peta digital sudah dapat menampilkan sesuai dengan parameter dan dapat menampikan informasi meliputi nilai indeks rel dan lokasi. Pada pengujian *user interface* laporan nilai indeks, dibagi menjadi empat pengujian berdasarkan parameter nilai indeks rel.

Pada Gambar 4.1 pada *ID* indeks dengan nomor 1125 menunjukkan nilai indeks rel sebesar 569,73. 569,73 merupakan angka yang berada diatas parameter indeks rel 100, sehingga warna yang seharusnya tertera pada *marker* adalah warna hijau. Pada pengujian ini, warna yang tertera pada *marker* adalah warna hijau, sehingga proses penampilan warna *marker* sudah benar.

Gambar 4.1: *Marker* dengan indeks rel sebesar 569,73 berwarna hijau.

Pada Gambar 4.2 pada *ID* indeks dengan nomor 940 menunjukkan nilai indeks rel sebesar 97,97. 97,97 merupakan angka yang berada diantara parameter indeks rel 85-99, sehingga warna yang seharusnya tertera pada *marker* adalah warna kuning. Pada pengujian ini, warna yang tertera pada *marker* adalah warna kuning, sehingga proses penampilan warna *marker* sudah benar.



Gambar 4.2: *Marker* dengan indeks rel sebesar 97,97 berwarna kuning.

Pada Gambar 4.3 pada *ID* indeks dengan nomor 1157 menunjukkan nilai indeks rel sebesar 79,8. 79,8 merupakan angka yang berada diatas parameter indeks rel 70-84, sehingga warna yang seharusnya tertera pada *marker* adalah warna jingga. Pada pengujian ini, warna yang tertera pada *marker* adalah warna jingga, sehingga proses penampilan warna *marker* sudah benar.

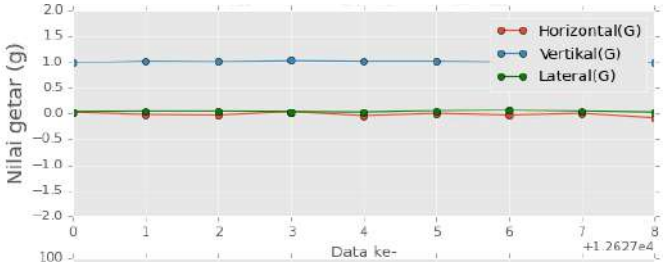
Gambar 4.3: *Marker* dengan indeks rel sebesar 79,8 berwarna jingga.

Pada Gambar 4.4 pada *ID* indeks dengan nomor 1659 menunjukkan nilai indeks rel sebesar 17. 17 merupakan angka yang berada dibawah parameter indeks rel 69, sehingga warna yang seharusnya tertera pada *marker* adalah warna merah. Pada pengujian ini, warna yang tertera pada *marker* adalah warna merah, sehingga proses penampilan warna *marker* sudah benar.

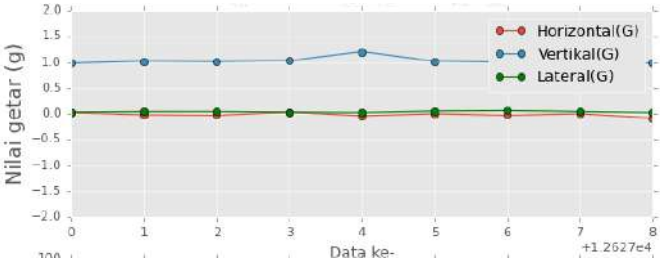
Gambar 4.4: *Marker* dengan indeks rel sebesar 17,0 berwarna merah.

4.1.2 *User Interface* Pada Sensor *Node*

Pada pengujian ini dilakukan guna mengetahui seberapa baik data divisualisasikan dan pengoperasian alat dengan menggunakan *user interface* pada sensor *node*. Visualisasi dilakukan pada kondisi getaran yang berbeda yaitu kondisi normal yang ditunjukkan oleh Gambar 4.5 dan kondisi abnormal getaran yang ditunjukkan oleh Gambar 4.6. Pada kondisi abnormal tampilan *user interface* parameter besar grafik getaran vertikal lebih dari 1.10 g dan kurang dari 0.90 g dan getaran horizontal lebih dari 0.075 g dan kurang dari -0.075 g. Sedangkan pada kondisi normal ditunjukkan dengan tampilan *user interface* parameter besar grafik getaran vertikal berada dintara 0.90 hingga 1.10 g dan getaran horizontal berada diantara -0.075 g hingga 0.075 g.

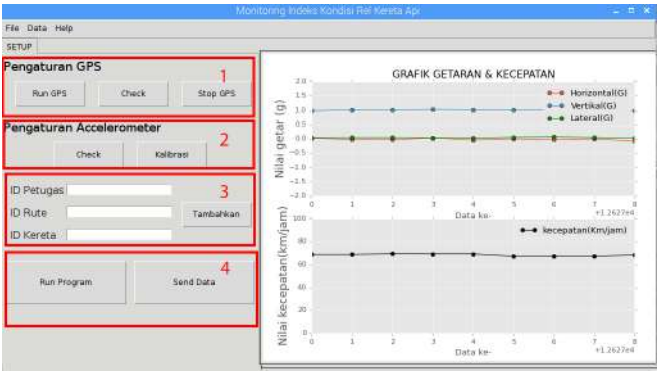


Gambar 4.5: Visualisasi besar nilai getaran keadaan normal.



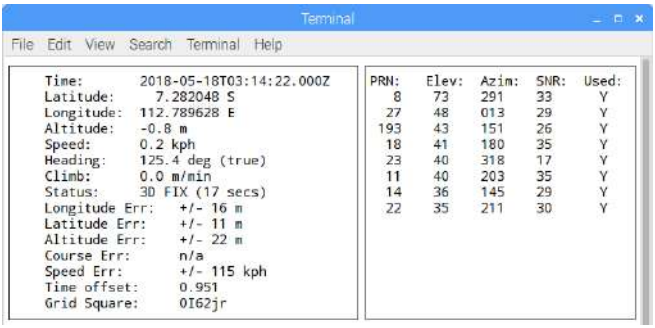
Gambar 4.6: Visualisasi besar nilai getaran keadaan abnormal.

User interface sensor *node* dapat dilihat pada Gambar 4.7 terdapat beberapa menu dalam bentuk tombol dan *form* yang diuji



Gambar 4.7: Blok pada *User interface* sensor *node* yang diujikan.

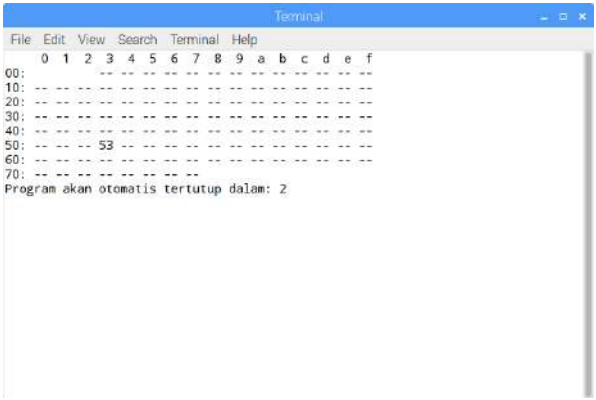
fungsionalitasnya. Menu pertama yakni menu pengaturan GPS meliputi tombol '*Run GPS*' berfungsi untuk menghubungkan *UART GPS* dengan *GPS Daemon*, '*Stop GPS*' berfungsi untuk memutus sambungan *UART GPS* dengan *GPS Daemon*, dan '*Check*, berfungsi untuk memeriksa perangkat GPS telah menerima informasi lokasi aktual seperti ditunjukan oleh Gambar 4.8.



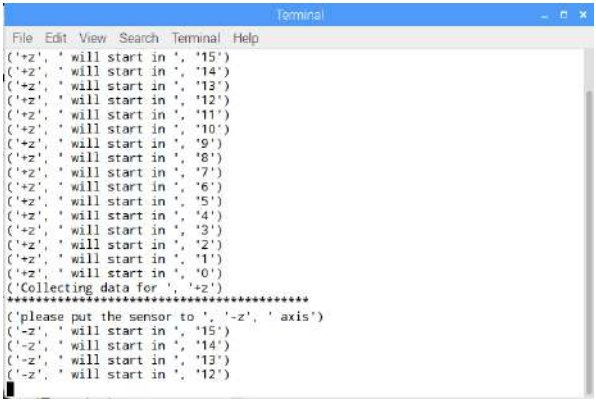
Gambar 4.8: Tampilan menu *check* GPS.

Menu kedua yakni menu pengaturan sensor akselerometer meliputi tombol '*check*' yang berfungsi untuk memeriksa sensor akselerometer telah tersambung dengan perangkat sensor *node*. Perangkat yang telah tersambung ditandai dengan munculnya informasi

alamat I^2C yaitu 53 seperti ditunjukan oleh Gambar 4.9. Tombol 'kalibrasi' berfungsi untuk memulai proses kalibrasi sensor akselerometer yang ditunjukan oleh gambar 4.10.



Gambar 4.9: Tampilan pada menu *check* akselerometer.

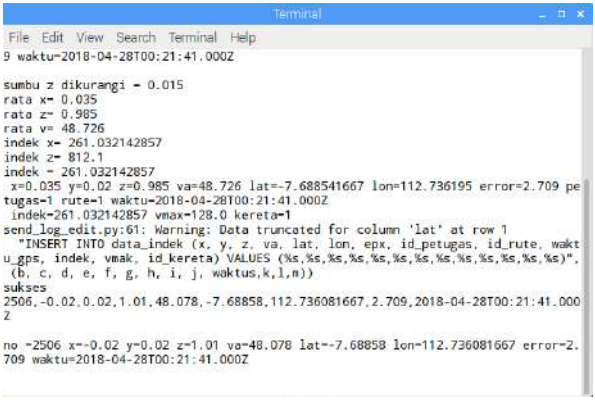


Gambar 4.10: Tampilan pada menu proses kalibrasi sensor akselerometer.

Menu ketiga yaitu pengaturan nomor *ID* yang digunakan meliputi *ID* petugas yang megoperasikan sensor *node*, *ID* rute perjalanan, dan *ID* kereta api yang digunakan untuk pengukuran.

Menu Keempat yaitu menu pogram utama meliputi tombol '*run program*' berfungsi untuk menjalankan program akuisisi sensor *node* yang ditunjukan oleh Gambar 4.11 dan tombol '*send data*' berfungsi untuk menjalankan program pengiriman data ke *cloud storage* yang ditunjukan oleh Gambar 4.12.

Gambar 4.11: Tampilan menu proses akuisisi sensor *node*.



Gambar 4.12: Tampilan menu proses pengiriman data ke *cloud storage*.

4.2 Pengujian Perangkat

Pada penelitian ini, dilakukan pengujian pada perangkat yang digunakan dalam sensor *node* yakni sensor akselerometer dan GPS untuk mengetahui nilai akurasi sensor akselerometer dan sensor GPS. Ada dua macam pengujian untuk sensor GPS yakni pengujian akurasi lokasi dan pengujian akurasi kecepatan dan pengujian nilai akurasi sensor akselerometer.

4.2.1 Pengujian Akurasi Lokasi Kondisi Statis

Pada pengujian ini, dilakukan perbandingan antara nilai *latitude* dan *longitude* yang didapatkan oleh perangkat GPS sensor *node* dengan lokasi patok *Benchmark* ITS 01. Pengujian ini digunakan untuk melihat nilai galat dari perangkat GPS yang digunakan didalam sensor *node* saat kondisi statis atau diam, jika dibandingkan dengan patok *Benchmark* ITS 01 yang sudah memiliki tingkat keakuratan yang tinggi dengan nilai *latitude* -7.2820129062839944 dan *longitude* 112.79415777941604. Ada tiga jenis pengujian akurasi lokasi berdasarkan waktu pengambilan dan kondisi cuaca yakni pagi cerah, siang cerah, dan malam cerah.

1. Pagi cerah

   Pada Tabel 4.1 hasil akuisisi data perangkat GPS sensor *node* dan nilai galat saat kondisi statis, dapat diketahui bahwa nilai galat terbesar adalah 7 meter, nilai galat terkecil adalah 3 meter, nilai rata-rata galat adalah 5,6 meter, dan nilai standar deviasi galat adalah 1,5 meter.

Tabel 4.1: Akurasi lokasi statis GPS pagi cerah.

No	GPS		Galat (m)
	Lat	Long	
1	-7.282013333	112.794133333	3
2	-7.281993333	112.794105	6
3	-7.282003335	112.794106667	6
4	-7.281993333	112.794105	6
5	-7.281995	112.794095	7
Rata-rata			5,6

2. Siang cerah

   Pada Tabel 4.2 hasil akuisisi data perangkat GPS sensor *node* dan nilai galat saat kondisi statis, dapat diketahui bahwa nilai galat terbesar adalah 5 meter, nilai galat terkecil adalah 4 meter, nilai rata-rata galat adalah 4,6 meter, dan nilai standar deviasi galat adalah 0,5 meter.

Tabel 4.2: Akurasi lokasi statis GPS siang cerah.

No	GPS		Galat (m)
	Lat	Long	
1	-7.282003333	112.794198333	5
2	-7.282003334	112.794196667	4
3	-7.282003335	112.794195	4
4	-7.282003333	112.7942	5
5	-7.282003333	112.79421	5
Rata-rata			4,6

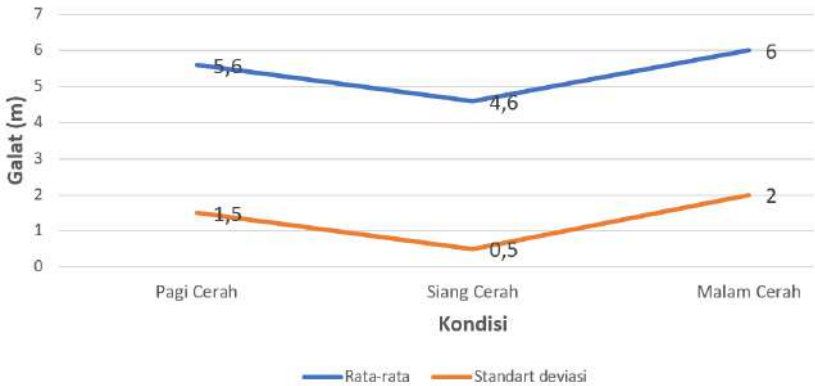
3. Malam cerah

   Pada Tabel 4.3 hasil akuisisi data perangkat GPS sensor *node* dan nilai galat saat kondisi statis, dapat diketahui bahwa nilai galat terbesar adalah 8 meter, nilai galat terkecil adalah 4 meter, nilai rata-rata galat adalah 6,0 meter, dan nilai standar deviasi galat adalah 2,0 meter.

Tabel 4.3: Akurasi lokasi statis GPS malam cerah.

No	GPS		Galat (m)
	Lat	Long	
1	-7.282025	112.794086667	8
2	-7.282023333	112.794088333	8
3	-7.282028333	112.794106667	6
4	-7.282018333	112.794123333	4
5	-7.282016667	112.794125	4
Rata-rata			6,0

Dari hasil tiga pengujian akurasi lokasi kondisi statis yakni pada pagi cerah, siang cerah, dan malam cerah dapat disimpulkan kedalam sebuah grafik garis seperti pada Gambar 4.13, bahwa nilai galat rata-rata tertinggi adalah 6,0 meter pada waktu malam cerah dengan nilai standar deviasi galat adalah 2,0 meter. Sedangkan pada nilai galat rata-rata terendah adalah 4,6 meter pada waktu siang cerah dengan nilai standar deviasi galat adalah 0,5 meter. Dan pada pagi cerah nilai galat rata-rata adalah 5,6 meter dengan nilai standar deviasi galat adalah 1,5 meter.



Gambar 4.13: Rata-rata dan standar deviasi galat akurasi lokasi GPS.

4.2.2 Pengujian Akurasi Lokasi Kondisi Dinamis

Pada pengujian ini, dilakukan perbandingan antara nilai *latitude* dan *longitude* yang didapatkan oleh perangkat GPS sensor node dengan lokasi patok *Benchmark* 02 GRAHA. Pengujian ini digunakan untuk melihat besar *delay* saat proses akusisi data dari perangkat GPS yang digunakan di dalam sensor node saat kondisi dinamis atau bergerak, jika dibandingkan dengan patok *Benchmark* 02 GRAHA yang sudah memiliki tingkat keakuratan yang tinggi dengan nilai *latitude* -7.276424504554041 dan *longitude* 112.790235103 26242. Ada tiga jenis pengujian akurasi lokasi berdasarkan waktu pengambilan dan kondisi cuaca yakni pagi cerah, siang cerah, dan malam cerah. Masing-masing pengujian dibagi menjadi lima pa-

rameter berdasarkan kecepatan yakni 30 km/jam, 40 km/jam, 50 km/jam, 60 km/jam, dan 70 km/jam.

1. Pagi cerah

   Pada Tabel 4.4 hasil akuisisi data perangkat GPS sensor *node* dan nilai galat saat kondisi dinamis, dapat diketahui bahwa nilai *delay* terbesar adalah 4,0 detik, nilai *delay* terkecil adalah 2,0 detik, nilai rata-rata *delay* adalah 3,0 detik, dan nilai standar deviasi galat adalah 0,7 detik.

   Tabel 4.4: Akurasi lokasi dinamis GPS pagi cerah.

No	Kecepatan (km/jam)	GPS		Delay (detik)
		Lat	Long	
1	30	-7.276426667	112.790283333	2,0
2	40	-7.276453333	112.790265	3,0
3	50	-7.276453333	112.790278333	3,0
4	60	-7.276463333	112.790261667	3,0
5	70	-7.276396667	112.790276667	4,0
Rata-rata				3,0

2. Siang cerah

   Pada Tabel 4.5 hasil akuisisi data perangkat GPS sensor *node* dan nilai galat saat kondisi dinamis, dapat diketahui bahwa nilai *delay* dari seluruh parameter kecepatan adalah 2,0 detik, nilai rata-rata *delay* adalah 2,0 detik, dan nilai standar deviasi galat adalah 0 detik.

   Tabel 4.5: Akurasi lokasi dinamis GPS pagi cerah.

No	Kecepatan (km/jam)	GPS		Delay (detik)
		Lat	Long	
1	30	-7.276411667	112.790228333	2,0
2	40	-7.276411667	112.790235	2,0
3	50	-7.276445	112.790231667	2,0
4	60	-7.276458333	112.790223333	2,0
5	70	-7.27641	112.790256667	2,0
Rata-rata				2,0

3. Malam cerah

   Pada Tabel 4.6 hasil akuisisi data perangkat GPS sensor *node* dan nilai galat saat kondisi dinamis, dapat diketahui bahwa nilai *delay* terbesar adalah 4,0 detik, nilai *delay* terkecil adalah 3,0 detik, nilai rata-rata *delay* adalah 3,4 detik, dan nilai standar deviasi galat adalah 0,5 detik.

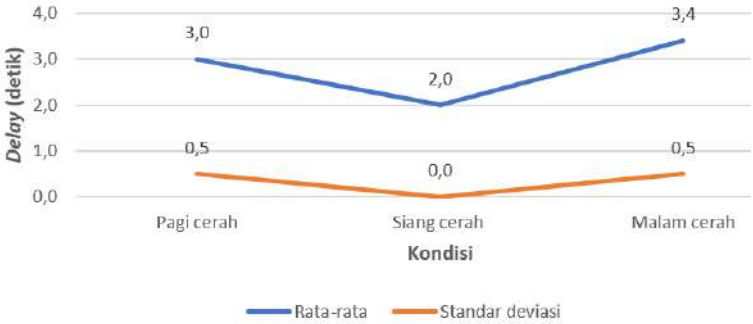
Tabel 4.6: Akurasi lokasi dinamis GPS malam cerah.

No	Kecepatan (km/jam)	GPS		Delay (detik)
		Lat	Long	
1	30	-7.276645	112.790223333	3,0
2	40	-7.27646	112.79024833	3,0
3	50	-7.276433333	112.790275	3,0
4	60	-7.276463333	112.79024	4,0
5	70	-7.27642	112.790258333	4,0
Rata-rata				3,4

Dari hasil tiga pengujian akurasi lokasi kondisi dinamis yakni pada pagi cerah, siang cerah, dan malam cerah dapat disimpulkan kedalam sebuah grafik garis seperti pada Gambar 4.14, bahwa nilai *delay* rata-rata tertinggi adalah 6,0 meter pada waktu malam cerah dengan nilai standar deviasi galat adalah 2,0 meter. Sedangkan pada nilai galat rata-rata terendah adalah 4,6 meter pada waktu siang cerah dengan nilai standar deviasi galat adalah 0,5 meter. Dan pada pagi cerah nilai galat rata-rata adalah 5,6 meter dengan nilai standar deviasi galat adalah 1,5 meter.

4.2.3 Pengujian Akurasi Kecepatan

Pada pengujian akurasi kecepatan, dilakukan perbandingan antara nilai kecepatan yang didapatkan oleh perangkat GPS yang digunakan didalam sensor *node* dengan *speedometer* sepeda motor dengan kecepatan yang ditentukan. Ada lima parameter kecepatan yang diuji yaitu kecepatan 20 Km/jam, 30 Km/jam, 40 Km/jam, 50 Km/jam, dan 60 Km/jam. Pengujian ini dilakukan untuk melihat nilai galat dari perangkat GPS yang digunakan pada sensor *node*, jika dibandingkan dengan *speedometer* sepeda motor yang sudah memiliki tingkat keakuratan yang tinggi. Ada tiga jenis penguji-



Gambar 4.14: Rata-rata dan standar deviasi galat akurasi lokasi GPS.

an akurasi kecepatan berdasarkan waktu pengambilan dan kondisi cuaca yakni pagi cerah, siang cerah, dan malam cerah.

1. Pagi cerah

   Dari hasil akuisisi data dari perangkat GPS sensor *node* dan *speedometer* sepeda motor dengan lima parameter kecepatan pada waktu pagi kondisi cerah dapat diketahui dari Tabel 4.7 bahwa nilai galat terbesar adalah 9,41% , nilai galat terkecil adalah 4,50%, nilai rata-rata galat adalah 7,11%, dan nilai standar deviasi galat adalah 2,05%.

Tabel 4.7: Akurasi kecepatan GPS pagi cerah.

No	Rata-rata GPS (Km/jam)	*Speedometer* (Km/jam)	Galat (%)
1	19,10	20	4,50
2	28,03	30	6,56
3	36,42	40	8,96
4	45,30	50	9,41
5	56,33	60	6,12
Rata-rata			7,11

2. Siang cerah

   Dari hasil akuisisi data dari perangkat GPS sensor *node* dan

speedometer sepeda motor dengan lima parameter kecepatan pada waktu siang kondisi cerah dapat diketahui dari Tabel 4.8 bahwa nilai galat terbesar adalah 12,57%, nilai galat terkecil adalah 4,50%, nilai rata-rata galat adalah 7,10%, dan nilai standar deviasi galat adalah 3,59%.

Tabel 4.8: Akurasi kecepatan GPS siang cerah.

No	Rata-rata GPS (Km/jam)	*Speedometer* (Km/jam)	Galat (%)
1	17,49	20	12,57
2	28,06	30	6,47
3	36,63	40	8,43
4	51,77	50	3,53
5	57,30	60	4,50
Rata-rata			7,10

3. Malam cerah

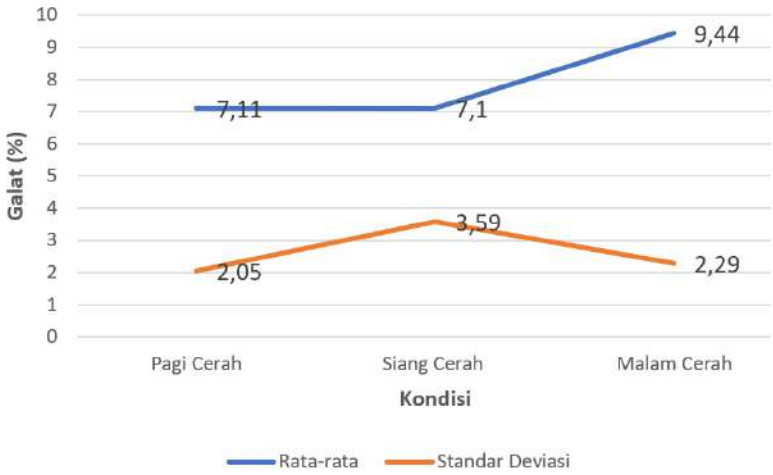
   Dari hasil akuisisi data dari perangkat GPS sensor *node* dan *speedometer* sepeda motor dengan lima parameter kecepatan pada waktu siang kondisi cerah dapat diketahui dari Tabel 4.9 bahwa nilai galat terbesar adalah 11,03%, nilai galat terkecil adalah 6,75%, nilai rata-rata galat adalah 9,44%, dan nilai standar deviasi galat adalah 2,29%.

Tabel 4.9: Akurasi kecepatan GPS malam cerah.

No	Rata-rata GPS (Km/jam)	*Speedometer* (Km/jam)	Galat (%)
1	18,65	20	6,75
2	26,30	30	12,35
3	36,21	40	9,33
4	46,12	50	7,76
5	53,38	60	11,03
Rata-rata			9,44

Gambar 4.15 merupakan grafik hasil pengujian kecepatan yang dideteksi GPS, dapat dilihat bahwa nilai galat rata-rata terendah

adalah sebesar 7,10% pada kondisi siang cerah dengan nilai standar deviasi adalah 3,59%. Sementara nilai galat rata-rata tertinggi adalah sebesar 9,44% pada kondisi malam cerah dengan nilai standar deviasi adalah 1,57%. Dan pada kondisi pagi cerah, besar nilai galat rata-rata adalah 7,11% dengan nilai standar deviasi adalah 2,32%.



Gambar 4.15: Rata-rata dan standar deviasi galat kecepatan GPS.

4.2.4 Pengujian Akurasi Sensor Akselerometer

Pada pengujian akurasi sensor akseleromter, dilakukan dengan membandingkan selisih nilai getaran masing-masing sumbu horizontal(X), vertikal (Z), dan lateral (Y) sensor akselerometer yang digunakan pada sensor *node* dengan gawai pada kereta yang sama. Sensor akselerometer yang digunakan pada sensor *node* adalah tipe ADXL345. Sedangkan gawai yang diguanakan adalah Sony Xperia Z3+ dengan tipe sensor akselerometer yakni *Invensense Accelerometer* versi 1. Pengujian ini dilakukan guna mengetahui nilai galat dari sensor akselerometer yang digunakan pada sensor *node* jika dibandingkan dengan sensor akselerometer dari gawai yang sudah memiliki tingkat akurasi yang tinggi. Pada pengujian akurasi sensor akselerometer, pengambilan data pada masing-masing sensor dila-

kukan sebanyak 100 kali. Pada Tabel 4.10 terdapat sepuluh sampel nilai sumbu horizontal (X), vertikal(Z), lateral (Y) dan sensor akselerometer gawai dan sensor akselerometer pada sensor *node*.

Tabel 4.10: Nilai sensor akselerometer sumbu X, Y, dan Z pada sensor *node* dan gawai.

Gawai			**Sensor *Node***		
X (g)	**Y (g)**	**Z (g)**	**X (g)**	**Y (g)**	**Z (g)**
0,02	0,01	1,00	0,00	0,02	1,00
0,02	0,00	0,96	0,01	0,00	0,99
0,01	0,01	1,01	0,00	0,01	1,04
0,02	0,00	1,02	0,02	0,00	1,05
0,00	0,01	1,01	0,01	0,01	0,97
0,02	0,00	0,98	0,02	0,02	0,99
0,02	0,01	0,97	0,01	0,01	1,00
0,02	0,01	0,97	0,01	0,02	1,02
0,01	0,02	1,01	0,02	0,01	1,01
0,05	0,01	1,01	0,02	0,01	0,98

Dari Tabel 4.11, nilai galat rata-rata getaran masing-masing sumbu adalah 0.012 g pada sumbu X, 0.014 g pada sumbu Y, dan 0.033 g pada sumbu Z. Sedangkman nilai galat maksimal masing-masing sumbu adalah 0.040 g pada sumbu X, 0.060 g pada sumbu Y, dan 0.280 g pada sumbu Z. Dan besar nilai galat minimal masing-masing sumbu memiliki nilai sama yaitu 0.00 g.

Tabel 4.11: Nilai minimum, rata-rata, dan maksimum galat sensor akselerometer.

Nilai Galat	**Sumbu**		
	X (g)	**Y (g)**	**Z (g)**
Rata-rata	0,012	0,014	0,033
Maksimal	0,040	0,060	0,280
Minimal	0,000	0,000	0,000

4.3 Pengujian Interval Waktu Penyimpanan ke *Database*

Pada bagian ini, dilakukan pengujian untuk mengetahui waktu rata-rata jeda pengiriman dari sensor *node* sebagai pengirim ke *database* sebagai penerima. Pengujian dilakukan dengan melihat perbedaan waktu yang didapat oleh waktu sensor *node* yang sebelumnya telah disinkronisasi oleh GPS sebagai pengirim dan waktu yang didapat *database* sebagai penerima. Waktu jeda dihitung dengan cara mengurangi waktu penerimaan data dengan waktu pengiriman data. Pengambilan data pengujian dilakukan sebanyak 203 kali.

Pada Tabel 4.12 terdapat sepuluh sampel pengujian interval waktu penyimpanan data dari sensor *node* sebagai pengirim dan *database* sebagai penerima. Umumnya, tanggal dan waktu yang didapatkan oleh *database* dan GPS menggunakan satuan UTC (*Coordinated Universal Time*). Indonesia menggunakan zona waktu (UTC+07:00) sehingga nilai dari waktu pada *database* dan GPS pada sensor *node* sebelumnya telah ditambah dengan 07:00 oleh sistem.

Tabel 4.12: Waktu dan tanggal pengiriman data (GPS) dan penerimaan data (*database*).

Pengirim		**Penerima**	
Tanggal GPS	**Waktu GPS**	**Tanggal Database**	**Waktu Database**
2018-05-22	15:41:12	2018-05-22	15:41:13
2018-05-22	15:41:14	2018-05-22	15:41:16
2018-05-22	15:41:38	2018-05-22	15:41:39
2018-05-22	15:42:05	2018-05-22	15:42:06
2018-05-22	15:42:23	2018-05-22	15:42:25
2018-05-22	15:42:31	2018-05-22	15:42:32
2018-05-22	15:47:21	2018-05-22	15:47:23
2018-05-22	15:47:35	2018-05-22	15:47:37
2018-05-22	15:49:33	2018-05-22	15:49:34
2018-05-22	15:50:32	2018-05-22	15:50:32

Dari Tabel 4.13 pengujian waktu pengiriman data dapat dilihat bahwa nilai waktu pengiriman tercepat adalah selama 1 detik. Nilai rata-rata pengiriman data adalah selama 1,04 detik. Sementara waktu pengiriman terlama adalah selama 2 detik.

Tabel 4.13: Nilai minimum, rata-rata, dan maksimum waktu pengiriman data.

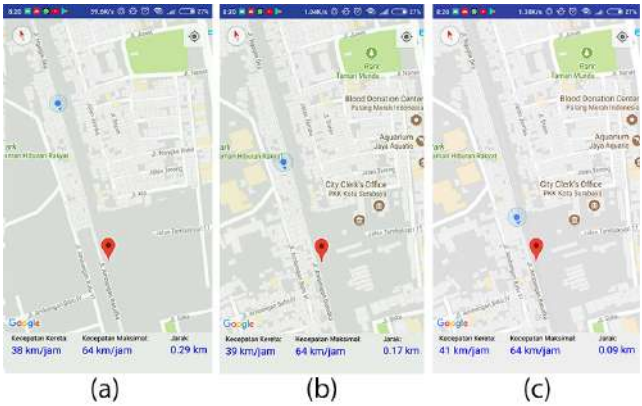
Nilai	Interval (detik)
Minimum	1,00
Rata-rata	1,04
Maksimum	2,00

4.4 Pengujian Sistem Peringatan Dini

Pada pengujian sistem peringatan dini batas kecepatan maksimal kereta api, dilakukan pengujian aplikasi peringatan batas kecepatan kereta api guna mengetahui aplikasi dapat memberikan notifikasi sesuai batas kecepatan maksimal yang telah dirancang. Pada pengujian sistem peringatan dini, aplikasi akan memberikan notifikasi berupa kecepatan aktual kereta api, batas kecepatan maksimal, jarak antara posisi aktual kereta api dengan *marker* titik batas kecepatan maksimal, poisisi aktual kereta, dan titik batas kecepatan maksimal kereta api dengan nilai indeks rel kurang dari 100 berupa *marker* berwarna merah. Pengujian dibagi menjadi enam berdasarkan parameter jarak antara posisi aktual kereta api dengan lokasi batas kecepatan maksimal. Pembagian parameter jarak dibagi menjadi enam antara lain lebih dari 500 m, 499-400 m, 399-300 m, 299-200 m, 199-100 m, dan 99-0 m. Terdapat enam sampel hasil pengujian aplikasi peringatan dini batas kecepatan maksimal kereta api yang dapat dilihat pada Gambar 4.16 dan Gambar 4.17.

Berdasarkan hasil pengujian pada Gambar 4.16 dan Gambar 4.17, dapat diketahui bahwa pada paremeter jarak lebih dari 500 m, posisi aktual terdeteksi dengan jarak 660 m pada kecepatan aktual 41 km/jam. Pada paremeter jarak antara 499-499 m, posisi aktual terdeteksi dengan jarak 460 m pada kecepatan aktual 39 km/jam. Pada parameter jarak antara 399-300 m, posisi aktual terdeteksi dengan jarak 320 m pada kecepatan 36 km/jam. Pada parameter

Gambar 4.16: Pengujian aplikasi peringatan dini batas kecepatan kereta api berdasarkan parameter jarak: (a)lebih dari 500 m, (b)499-400 m, dan (c) 399-300 m.



Gambar 4.17: Pengujian aplikasi peringatan dini batas kecepatan kereta api berdasarkan parameter jarak: (a)299-200 m, (b)199-100 m, dan (c) 99-0 m.

jarak antara 299-200 m, posisi aktual terdeteksi dengan jarak 290 m pada kecepatan 38 km/jam. Pada parameter jarak antara 199-100 m, posisi aktual terdeteksi dengan jarak 170 m pada kecepatan 39

km/jam. Dan pada parameter jarak antara 99-0 m, posisi aktual terdeteksi dengan jarak 90 m pada kecepatan 41 km/jam. Hasil seluruh pengujian sistem peringatan dini batas kecepatan maksimal kereta api, dapat dilihat pada Tabel 4.14.

Tabel 4.14: Hasil pengujian aplikasi peringatan dini batas kecepatan maksimal kereta api.

Parmeter jarak (m)	Jarak aktual (m)	Kecepatan aktual (km/jam)
>500	660	41
499-400	460	39
399-300	320	36
299-200	290	38
199-100	170	39
99-0	90	41

4.5 Pengujian Validasi Data

Pada bagian ini, dilakukan pengujian guna mengetahui tingkat validasi data yang digunakan untuk mengidentifikasi kondisi rel. Pengujian dilakukan dengan melihat perbedaan nilai getaran dan letak lokasi pengukuran menggunakan kereta yang berbeda. Perbedaan nilai getaran dihitung dengan cara mencari selisih nilai getaran dengan parameter warna marker yang sama dan lokasi terdekat antara kedua marker. Dan perbedaan letak lokasi pengukuran dihitung dengan cara mencari jarak antara kedua lokasi pengukuran kondisi rel kereta api. Pada pengujian ini, kereta api yang digunakan ada dua yakni kereta eksekutif Bima (kode kereta 43) dengan rute Surabaya gubeng (SGU) - Malang (ML) dan kereta ekonomi Jayabaya (kode kereta 143) dengan rute Malang (ML) - Surabaya gubeng (SGU). Pengambilan data pengukuran kondisi rel kereta api dilakukan sebanyak enam kali.

Berdasarkan Tabel 4.15 hasil pengujian validasi data dapat diketahui bahwa rata-rata jarak adalah 34,3 meter dan rata-rata selisih nilai getaran pada sumbu horizontal (X) adalah 0,06 g, pada sumbu vertikal (Z) adalah 0,19 g, dan pada sumbu lateral (Y) adalah 0,01 g.

Tabel 4.15: Hasil pengujian validasi data.

No	Kode kereta	Indeks	Nilai Getaran (g)			Kecepatan (Km/jam)	Selisih Jarak (m)	Selisih nilai getaran (g)		
			X	Y	Z			X	Y	Z
1	43	77	0,17	0,04	1,10	69,8	4	0,14	0,00	0,33
	143	75	0,03	0,04	0,77	69,6				
2	43	73	0,07	0,03	1,23	67,0	10	0,02	0,02	0,01
	143	72	0,09	0,05	1,24	69,4				
3	43	80	0,04	0,05	1,22	70,2	31	0,02	0,04	0,43
	143	84	0,02	0,01	0,79	70,4				
4	43	87	0,00	0,02	1,14	49,1	119	0,12	0,00	0,26
	143	89	0,12	0,02	0,88	56,8				
5	43	90	0,09	0,035	1,05	40,8	30	0,02	0,00	0,05
	143	86	0,11	0,035	1,10	48,3				
6	43	98	0,05	0,05	0,88	46,9	27	0,02	0,01	0,26
	143	90	0,03	0,04	1,14	50,6				
7	43	91	0,06	0,01	1,14	49,2	19	0,05	0,01	0,00
	143	96	0,01	0,02	1,14	53,6				
Rata-rata							34,3	0,6	0,01	0,19

Halaman ini sengaja dikosongkan

BAB 5
PENUTUP

5.1 Kesimpulan

Dari hasil pengujian yang sudah dilakukan dapat ditarik beberapa kesimpulan sebagai berikut:

1. Tingkat validasi data berdasarkan rata-rata selisih nilai getaran dan selisih jarak yang diperoleh dari jenis kereta dan kecepatan yang berbeda adalah sebesar 0,6 g untuk sumbu horizontal, 0,01 g untuk sumbu lateral, 0,19 g untuk sumbu vertikal, dan 34,3 meter untuk selisih jarak.
2. Tingkat keakurasian masing-masing sensor jika dilihat dari nilai galat rata-rata pada sensor GPS dalam menentukan lokasi serta besar kecepatan adalah 5,4 meter dengan *delay* gps 2,8 detik, dan 17,36% dan pada sensor akselerometer sebesar 0,059 g.
3. Proses pengiriman data oleh sensor sebagai pengirim *node* menuju *database server* sebagai penerima membutuhkan interval waktu rata-rata sebesar 1,04 detik.
4. Aplikasi peringatan dini batas kecepatan maksimal kereta api dapat menampilkan *marker* terdekat yang memiliki nilai indeks kurang dari 100 serta menampilkan informasi kecepatan aktual kereta api, batas kecepatan maksimal, dan jarak antara posisi kereta api dengan posisi *marker*.
5. Aplikasi dapat digunakan oleh petugas PT KAI untuk memantau kondisi rel kereta api berdasarkan warna pada *marker* (hijau, kuning, jingga, dan merah) sesuai dengan parameter nilai indeks kondisi rel kereta api yang telah ditetapkan.

5.2 Saran

Untuk pengembangan penelitian selanjutnya terdapat beberapa saran sebagai berikut :

1. Pemasangan sensor akselerometer digital dipasang pada bagian dalam *bogie* gerbong kereta api dan pemasangan antena GPS di luar gerbong kereta api guna mendapatkan nilai ge-

taran dan posisi yang lebih akurat.

2. Penambahan data relasi antara patok kilometer rel kereta api dengan data geolokasi sehingga data yang divisualisasikan pada laporan indek rel dapat menampilkan dalam satuan kilometer rel.

DAFTAR PUSTAKA

[1] "Data investigasi kecelakaan perkeretaapian tahun 2010-2016." Terakhir diakses pada tanggal 27 April 2018. (Dikutip pada halaman , 1).

[2] "Lalu lintas dan angkutan jalan nomor 22 tahun 2009." Terakhir diakses pada tanggal 27 April 2018. (Dikutip pada halaman 2).

[3] "Deteksi keselamatan perjalanan kereta api." https://travel.kompas.com/read/2010/09/02/03165727/Deteksi.Keselamatan.Perjalanan.Kereta. Terakhir diakses pada tanggal 3 Juli 2018. (Dikutip pada halaman 2).

[4] I. Karyadi, "Studi kemungkinan penggunaan mikroprosesor untuk meenetapkan kecepatan maksimum kereta api yang dijinkan berdasarkan kondisi track," graduate's thesis, Institut Teknologi Bandung, Januari 1986. (Dikutip pada halaman , 5, 6).

[5] M. Subyanto, Dinamika Kendaraan Rel. Bandung: CV Komala, 1982. (Dikutip pada halaman , 6, 8).

[6] "Datasheet digital accelerometer adxl345." http://www.analog.com/media/en/technical-documentation/data-sheets/ADXL345.pdf. Terakhir diakses pada tanggal 28 April 2018. (Dikutip pada halaman , 9).

[7] "Using an accelerometer for inclination sensing." http://www.analog.com/media/en/technical-documentation/application-notes/AN-1057.pdf. Terakhir diakses pada tanggal 28 April 2018. (Dikutip pada halaman 10).

[8] "How to calibrate adxl345 triple axis accelerometer." http://wei48221.blogspot.com/2015/07/how-to-calibrate-adxl345-triple-axis_43.html. Terakhir diakses pada tanggal 4 Juni 2018. (Dikutip pada halaman , 11).

[9] "Adafruit ultimate gps breakout - 66 channel w/10 hz updates - version 3." https://cdn-learn.adafruit.com/downloads/pdf/adafruit-ultimate-gps.pdf. Terakhir diakses pada tanggal 28 April 2018. (Dikutip pada halaman , 11, 12).

[10] "Gps technologies and alternatives." https://www.rand.org/content/dam/rand/pubs/monograph_reports/MR614/MR614.appa.pdf. Terakhir diakses pada tanggal 28 April 2018. (Dikutip pada halaman , 12, 13).

[11] "National marine electronics association (nmea)-0183 tipe rmc." http://infocomputerscience.blogspot.co.id/2011/03/national-marine-electronics-association.html. Terakhir diakses pada tanggal 28 April 2018. (Dikutip pada halaman 14).

[12] "Raspberry pi pinout diagram circuit notes.." www.jameco.com/Jameco/workshop/circuitnotes/raspberry-pi-circuit-note.html. Terakhir diakses pada tanggal 20 April 2018. (Dikutip pada halaman , 15).

[13] "Raspberry pi 3 is out now! specs, benchmarks and more.." https://www.raspberrypi.org/magpi/raspberry-pi-3-specs-benchmarks/. Terakhir diakses pada tanggal 20 April 2018. (Dikutip pada halaman 16).

[14] "Mysqldb users guide." http://mysqlclient.readthedocs.io/user_guide.html#introduction. Terakhir diakses pada tanggal 20 Mei 2018. (Dikutip pada halaman 16).

[15] "V. beal, what is cloud storage?." https://www.webopedia.com/TERM/C/cloud_storage.html. Terakhir diakses pada tanggal 20 April 2018. (Dikutip pada halaman 17).

BIOGRAFI PENULIS

Virbyansah Achmadan Nurrohman yang disapa Byan, lahir di Lamongan Jawa Timur pada tanggal 21 Februari 1996. Merupakan anak pertama dari dua bersaudara. Penulis lulus dari SMP Negeri 1 Babat dan melanjutkan ke SMA Negeri 2 Lamongan. Penulis melanjutkan ke jenjang strata satu di Jurusan Teknik Multimedia dan Jaringan ITS yang sekarang telah berganti nama menjadi Departemen Teknik Komputer ITS. Dalam masa kuliah, penulis tertarik dengan pengembangan *Web Apps* dan *Internet of Things* (IoT). Selain itu penulis aktif berorganisasi, yaitu dengan menjadi wakil ketua Divisi Workshop Elektro 2016-2017, ketua bidang Media dan Informasi IMM ITS Surabaya, serta menjabat sebagai koordinator asisten Lab B401 Komputasi Multimedia Departemen Teknik Komputer ITS 2016-2017.

Halaman ini sengaja dikosongkan