

# Perancangan Struktur Data dan Desain Sistem Navigasi Sebagai Pemantauan Jarak Jauh Pada Sistem Operasi VDR on Demand

## *Design of Data Structure and Navigation System Design as Remote Monitoring in VDR Operating Systems on Demand*

Pandu Rama Satria Putranata<sup>1</sup>, Surya Michrani Nasution, S.T., M.T.<sup>2</sup>, Fairuz Azmi, S.T., M.T.<sup>3</sup>

<sup>1,2,3</sup>Prodi S1 Sistem Komputer, Fakultas Teknik Elektro, Universitas Telkom

<sup>1</sup> pramasatria13@gmail.com, <sup>2</sup> michrandi@telkomuniversity.ac.id, <sup>3</sup> fairuzazmi@telkomuniversity.ac.id

---

### Abstrak

Penggunaan alat Vehicle Data Recorder pada kendaraan automotif di jaman sekarang sudah menjadi hal yang biasa, dimana kendaraan seperti mobil sudah banyak di pasang alat Vehicle Data Recorder yang bervariasi terhadap manufaktur dan waktu kendaraan di buat. Kendaraan seperti mobil polisi dan ambulance umumnya sudah terpasang alat Vehicle Data Recorder yang lebih akurat dan lebih detail, alat ini di gunakan untuk memonitor penggunaan kendaraan, dan lalu di ambil datanya untuk keperluan investigasi atau pengembangan kendaraan. Akan tetapi jika data pada vehicle data recorder hilang dikarenakan kerusakan atau kecelakaan maka data pada Vehicle Data Recorder tidak bisa di ambil. Untuk Mengatasi masalah dimana Data Vehicle Data Recorder hilang maka penulis merancang dan membuat prototype sistem navigasi sebagai pemantau kendaraan jarak jauh berdasarkan Vehicle Data Recorder yang sudah terpasang pada mobil, di sistem ini Vehicle Data Recorder akan mengirim Data kepada server di mana server akan menyimpan Data Vehicle Recorder, sehingga Data tersimpan dan bisa digunakan pada waktu lain, Data ini lalu bisa di panggil pada sebuah program Komputer agar bisa memonitor kondisi kendaraan. Dalam memonitor kendaraan program akan menunjukkan kondisi kendaraan dalam bentuk tabel dan untuk menunjukkan posisi kendaraan maka dibuat peta pada Program agar pengguna bisa memonitor kondisi kendaraan. Dalam pengujian sistem ini di lakukan uji simulasi dimana sistem akan menerima data berdasarkan mobil yang berjalan berdasarkan arah yang sudah di tentukan untuk mengetahui perbedaan waktu disaat data di kirim pada mobil dan di tampilkan pada layar monitor, dari pengujian itu didapatkan bahwa sistem bisa membaca 300 Dataset per detik dan menerima 238 Dataset per detik.

**Kata Kunci:** *Internet Of Things, Vehicle Data Recorder, Web application, Vehicle monitoring*

---

### Abstract

*The use of Vehicle Data Recorder in automotif vehicle have become the norm, in which Vehicle like car have now been outfitted with Vehicle Data Recorder with varied accuracy and detail according to it's manufacturer and the time the vehicle is made. Vehicle such as Police car and ambulance are usually outfitted with a Vehicle Data Recorder device that has more accuracy and more detail, this device is used to monitor the vehicle usage, and then the data is recovered for investigational purpose or for vehicle development, but in a case where the data cannot be recovered because a malfunction or an accident happen then data has a risk of damage at which the data cannot be recovered. To fix this problem where the Data of Vehicle Data Recorder cannot be recovered, the writer has design and made a Prototype Navigation System as a long range Vehicle Data Recorder based on a onboard Vehicle Data Recorder that's been installed on a car. In this system the Vehilce Data Recorder will send its data to a Server that will store it's data, so the Data can be used in another time, this Data then can be called to a Computer Program to monitor the vehicle condition. In monitoring the Vehicle Data Recorder the program will show it's Vehicle Condition through a table function, as for the Vehicle position it will be shown through a map function. This two function will be used to monitor vehicle condition. As for the System test, the system will be test with a simulated condition where the system will accept a data based on a moving car with a predetermined route to show the system performance and diference in time when the vehicle send it's data and when the program is shown at the program. Based on the test it is found that the system will be abl to read 300 Data in a second and receive 238 Data in a sccond.*

**Keywords:** *Internet Of Things, Vehicle Data Recorder, Web application, Vehicle monitoring*

---

## 1. Pendahuluan

*Vehicle Data Recorder* adalah perangkat keras yang digunakan untuk merekam kejadian yang menyebabkan kerusakan pada kondisi kendaraan, *Vehicle Data Recorder* umumnya terpicu pada saat adanya perubahan arus elektronik pada mesin atau ada perubahan kecepatan secara mendadak[3], perangkat ini mirip dengan “*black box*” yang di pakai pada pesawat di mana perangkat mencatat kondisi kendaraan sebelum adanya kecelakaan.

Pada saat *Vehicle Data Recorder* terpicu oleh sebuah kejanggalan, maka perangkat akan memulai merekam kondisi kendaraan pada ROM yang disiapkan hingga baterai pada perangkat habis, untuk melihat hasil perekaman tersebut maka *Vehicle Data Recorder* harus di ambil kembali dari tempat kecelakaan, dan jika terjadi kasus dimana *Vehicle Data Recorder* tidak bisa diambil kembali di karenakan terjadinya kerusakan pada *Vehicle Data Recorder* itu sendiri atau kendaraan yang tidak bisa di ambil kembali karena kerusakan, maka data pada *Vehicle Data Recorder* tidak bisa di ambil kembali.

Untuk mengatasi masalah tersebut maka penulis akan menggunakan teknologi nirkabel untuk menyimpan data pada tempat lain, dengan menggunakan gprs data pada *Vehicle Data Recorder* bisa dikirim pada suatu server untuk mencegah adanya kehilangan data akibat kecelakaan atau kerusakan.

Selain itu dengan menggunakan server sebagai tempat penyimpanan data, maka hal ini juga mempermudah proses pengambilan data *Vehicle Data Recorder*, dengan menggunakan server maka data bisa di lihat dilain waktu, hal ini mempermudah pengamatan data untuk melakukan analisis kecelakaan karena tidak harus mengambil data pada tempat kejadian.

## 2. Dasar Teori

Bagian membahas landasan teori, Berikut ini adalah landasan teorinya:

### 2.1 IVMS (In Vehicle Monitoring System)

IVMS merupakan sebuah system monitoring yang mengkombinasikan sebuah alat elektronik pada sebuah kendaraan, dengan sebuah software yang di desain untuk memonitor alat elektronik pada kendaraan yang bertujuan untuk mengumpulkan data yang di proses pada software.

IVMS melacak berbagai macam aktifitas di dalam kendaraan seperti posisi GPS, berbagai macam input seperti sabuk pengaman, kecepatan, hantaman, peristiwa mendadak(akselerasi dan deselerasi, umumnya dilakukan oleh akselerometer triaxial) dikarenakan pedal gas, [4]

### 2.2 Automotive Navigation System

Sistem navigasi otomotif adalah bagian dari kontrol mobil atau bagian dari sebuah service dari pihak ketiga yang digunakan untuk menemukan arah dalam mobil. Sistem ini biasanya menggunakan perangkat navigasi satelit untuk mendapatkan data posisi yang kemudian berkorelasi dengan posisi di jalan. Informasi lalu lintas dapat digunakan untuk menyesuaikan rute.

Secara matematis, navigasi otomotif didasarkan pada masalah jalur terpendek, dalam teori grafik, yang meneliti bagaimana mengidentifikasi jalur yang paling memenuhi beberapa kriteria (terpendek, termurah, tercepat, dll) antara dua titik dalam jaringan besar.

### 2.3 Openstreetmap

OpenStreetMap (OSM) adalah sebuah proyek kolaborasi untuk membuat sebuah peta dunia yang bisa di ubah secara gratis. Keluaran utama dari proyek ini bukanlah peta akan tetapi data. Pembuatan OSM termotivasi dari restriksi pada penggunaan atau pengadaan informasi pada seluruh dunia, dan juga peningkatan alat satelit navigasi. Data dari OSM bisa di gunakan pada aplikasi tradisional, dan peran seperti menggantikan data default yang termasuk pada GPS receiver.

Dibuat oleh Steve Coast di Inggris pada tahun 2004, ini terinspirasi oleh keberhasilan Wikipedia dan dominasi data peta eksklusif di Inggris dan di tempat lain. Sejak itu, telah berkembang menjadi lebih dari 2 juta pengguna terdaftar, yang dapat mengumpulkan data menggunakan manual survei, perangkat GPS, foto udara, dan sumber gratis lainnya. Data crowdsourced ini kemudian tersedia di bawah Lisensi Open Database. Situs ini didukung oleh OpenStreetMap Foundation, sebuah organisasi nirlaba yang terdaftar di Inggris dan Wales.

### 2.4 Contraction hierachy

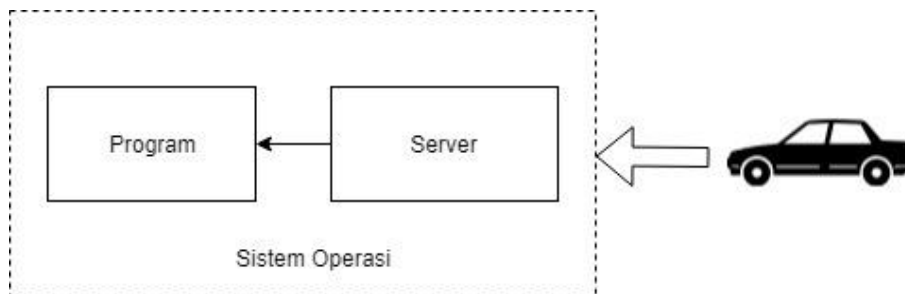
*Dalam pengaplikasian matematika, metode Contraction hirachies adalah sebuah teknik untuk mempercepat pencarian rute tercepat dengan membuat komputasi awal sebuah versi “contracted” dari sebuah connection graph [8],*

*Secara umum, algorima routing mempunyai 2 fase : proses preprocessing dari graph original dan queries, Contraction hierarchy (CH) adalah contoh ekstrim dari pendekatan hierarki. Hal ini bisa di capai dengan banyak cara, salah satu cara tersebut adalah dengan memberikan label untuk setiap node dalam sebuah urutan dari 1 hingga N, dimana N adalah nomor node di dalam sebuah graph*

### 3. Perancangan

#### 3.1. Gambaran Umum Sistem

Agar perangkat dapat berfungsi dengan baik dan menghasilkan keluaran yang diharapkan, sebuah perancangan system dilakukan. Berikut merupakan perancangan program secara umum yang ditunjukkan pada gambar 3.1.

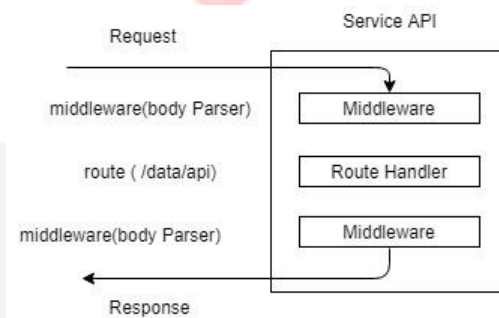


Gambar 3 1 Perancangan Program Secara Umum

Pertama sistem bekerja dengan mengambil data dari cloud server, lalu mengubah bentuk data pada cloud server, agar bisa di baca dengan format yang bisa di baca manusia, setelah itu system akan menampilkan data pada GUI, jika data bensin yang diterima dari server dibawah 20% maka sistem mulai mencari pom bensin terdekat berdasarkan lokasi data gps, lalu menentukan arah untuk mencapai lokasi pom bensin terdekat dan menampilkan arahnya pada GUI

#### 3.2. Cara Kerja Server

Server berfungsi untuk menyimpan data dari VDR dan memberikan data pada Program.

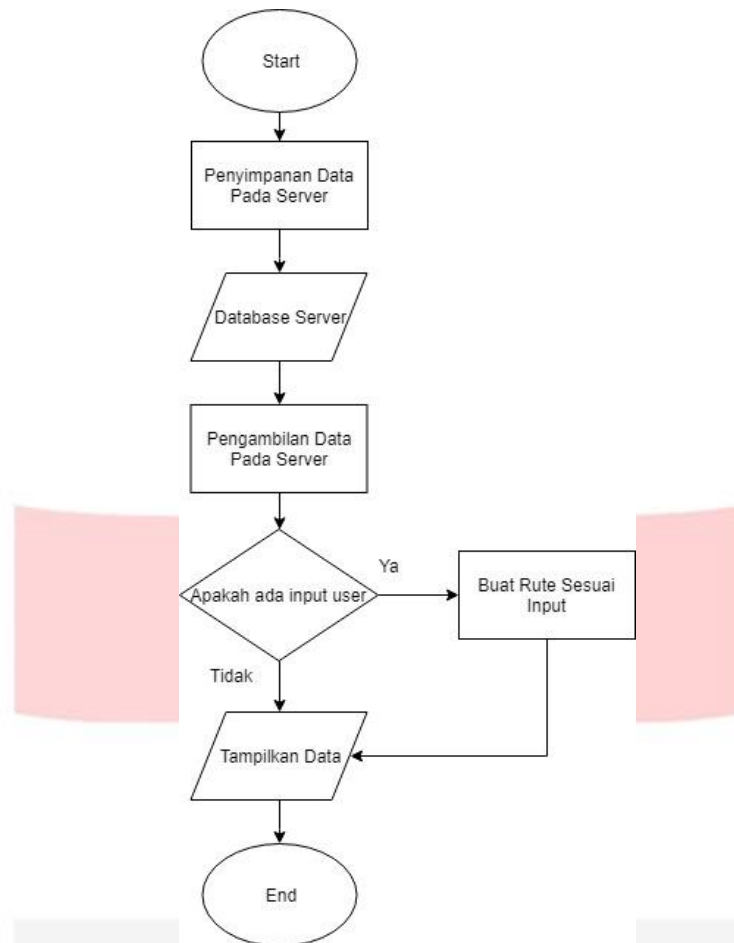


Gambar 3 2 API Server

Pada Gambar 3 2 Berikut merupakan cara kerja dari server, setiap request yang masuk pada server akan melalui parser agar bisa di baca program, lalu program akan mencocokkan request pada salah satu rute API dan memberikan response yang sesuai dengan request, lalu program akan melalui parser kembali untuk mengubah bentuk response sesuai request.

#### 3.3 Cara Kerja Program

Pada proses ini akan di jelaskan bagaimana cara kerja program dari tahap pengambilan data dari server, lalu tahap parsing data, memilih rute navigasi dan penampilan data :



Gambar 3 3 Flowchart Proses Program

Berikut merupakan penjelasan alur flowchart pada proses monitoring :

- 1) Sistem dimulai dengan menerima data kendaraan pada *Server*
- 2) Program akan melakukan request pada server untuk mobil dengan id tertentu dan *Server* akan memberikan data berdasarkan id tersebut.
- 3) Setelah mendapatkan data stream dari server program akan melakukan parsing pada stream tersebut agar bisa di tampilkan.
- 4) Di jendela Program peta akan muncul posisi mobil pada peta, jika Pengguna memberikan input untuk posisi tujuan yang diinginkan maka program akan menambahkan rute berdasarkan posisi yang diinginkan

#### 4. Implementasi Tampilan Aplikasi

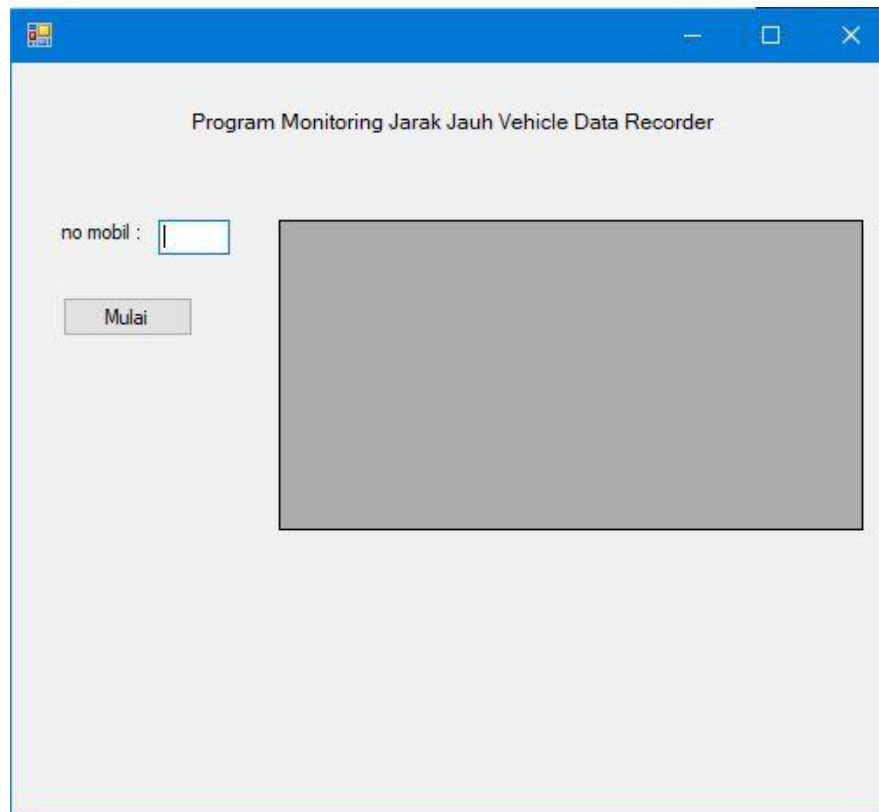
Pada bagian ini akan menjelaskan mengenai tampilan dari Monitoring Jarak Jauh yang dijelaskan sebagai berikut:

##### 1) Jendela Tabel Kondisi Kendaraan

Pada jendela ini User akan di hadapkan pada Teks Judul Program, Tabel sebuah table kosong dan juga tombol mulai. Pada jendela ini user bisa mengontrol aplikasi monitoring. Berikut adalah penjelasan setiap aspek pada wind

- a. Tabel Kondisi Kendaraan, Tabel akan berisi Data mobil yang diminta berdasarkan input box, Tabel akan berisikan dengan Data data mobil seperti Kecepatan, Panas Mesin, DLL.
- b. Input Box, Di Bagian Input Box User bisa memsukan nama mobil untuk mengambil Data mobil yang akan diinginkan, jika nama yang dimasukan tidak ada pada database maka server akan mengirimkan pesan pada

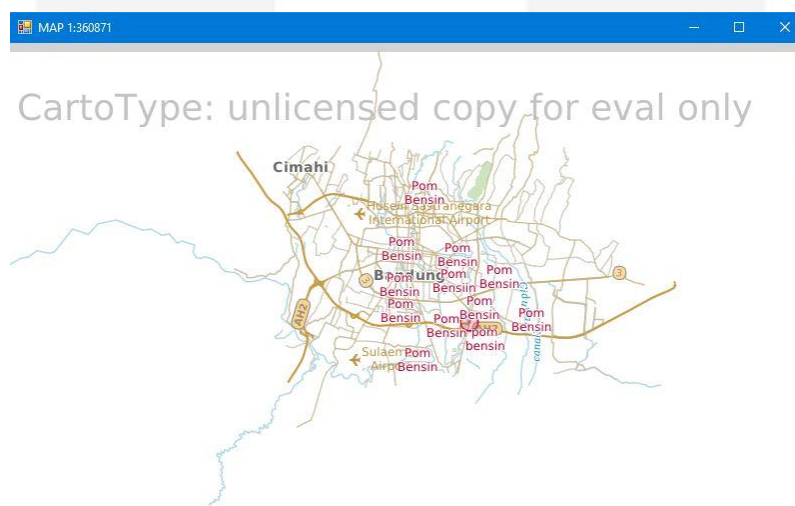
program bahwa database dengan nama yang dimasukan tidak ada, dan jika nama yang dimasukan ada maka server akan memberikan data tersebut.



**Gambar 4 1 Tampilan Jendela Tabel Kondisi Kendaraan**

## 2) Jendela Peta Posisi Kendaraan

Pada halaman ini terdapat teks pada gambar 4.2 yang menunjukkan informasi terkait Posisi Kendaraan. Halaman ini akan mulai menunjukkan Posisi Kendaraan setelah Jendela Tabel Kondisi Kendaraan meminta data pada Server.



**Gambar 4 2 Tampilan Jendela Peta Posisi Kendaraan**

## 5.1. Kesimpulan

Berdasarkan hasil analisis terhadap skenario pengujian dan implementasi program diambil beberapa kesimpulan yaitu:

1. Sistem mampu membaca data yang disimpan pada Server dengan proses penyimpanan data sebesar 220 Dataset per detik dan dengan proses pengambilan data sebesar 1081 Dataset per detik.
2. Program menggunakan resource computer dengan jumlah 100 mb memori dan 10% dari cpu processing
3. Routing akan selalu mencari rute paling tercepat berdasarkan jalur terpendek, dan tidak berdasarkan waktu tempuh tercepat ataupun kondisi jalan.

## 5.2. Saran

Berikut beberapa saran yang dapat diberikan untuk pengembangan penelitian tugas akhir selanjutnya, yaitu :

1. Program bisa di ubah dalam bentuk *web* agar bisa di baca di *Computer* ataupun di *Smartphone*.



## 1. DAFTAR PUSTAKA

- [1] Tomer, Toledo. Tsippy, Lotan. 2006. *In-Vehicle Data Recorder for Evaluation of Driving Behavior and Safety*. Transportation Research Record Journal of the Transportation Research Board 1953(1):112-119
- [2] Joe T. Correia, Ken . Iliadis, Ed s. McCarron, Mario A. Smolej 2001 *Utilizing Data From Automotive Event Data Recorder*. Canadian Multidisciplinary Road Safety Conference XII;. London. Ontario.
- [3] Dennis F. Andrews, Rudy Limpert, 2013, *ELECTRONIC CONTROL MODULE DATA IN LARGE TRUCK COLLISION ANALYSIS*, PC Brake Short Paper PCB 1-2013
- [4] DigiCore Australia. *IVMS Components and Functions*. Retrieved at 20 June 2019. <https://digicore-australia.com.au/resources/ivms-components-and-functions/>
- [5] Zhao, Jianfeng; Liang, Bodong; Chen, Qiuxia (2018-01-02). *The key technology toward the self-driving car*. International Journal of Intelligent Unmanned Systems. 6 (1): 2–20. doi:10.1108/IJUS-08-2017-0008. ISSN 2049-6427.
- [6] *OpenStreetMap powers map data on thousands of web sites, mobile apps, and hardware devices* Retrieved at 24 June 2019. <https://www.openstreetmap.org/about>
- [7] Coast, Steve. *Changes #1 on OpenStreetMap*. Retrieved 19 September 2018. <https://wiki.openstreetmap.org>
- [8] Michael, Tandy. *Contraction Hierarchies path finding algorithm, illustrated using three.js* retrieved 21 June 2019. <https://www.mjt.me.uk/posts/contraction-hierarchies/>
- [9] Robert, Geisberger. Peter, Sandler. Dominik, Schultes. 2008. *Daniel Deling. Contraction Hierarchies: Faster and Simpler Hierarchical Routing in Road Networks*. C.C. McGeoch (Ed.): WEA 2008, LNCS 5038, pp. 319–333, 2008.
- [10] *Web Services Architecture*. World Wide Web Consortium. 11 February 2004. 3.1.3 Relationship to the World Wide Web and REST Architectures. Retrieved 29 September 2016. <https://www.w3.org/TR/ws-arch/>
- [11] Engineering Task Force (IETF). RFC 7231. Retrieved 2018-02-14. <https://tools.ietf.org/html/rfc7231>
- [12] Duan, Yucong; Fu, Guohua; Zhou, Nianjun; Sun, Xiaobing; Narendra, Nanjangud; Hu, Bo (2015). "Everything as a Service (XaaS) on the Cloud: Origins, Current and Future Trends". 2015 IEEE 8th International Conference on Cloud Computing. IEEE. pp. 621–628. doi:10.1109/CLOUD.2015.88. ISBN 978-1-4673-7287-9.
- [13] Peter Mell; Timothy Grance (September 2011). *The NIST Definition of Cloud Computing* (Technical report). National Institute of Standards and Technology: U.S. Department of Commerce. doi:10.6028/NIST.SP.800-145. Special publication 800-145
- [14] Alex Chaffe. 2012. *What is a web application (or "webapp")?* <http://www.jguru.com/faq/view.jsp?EID=129328>