

Merancang Aplikasi Pemantau Kendaraan Pribadi Untuk *Intelligent Transportation* Di Jakarta

1st Muhammad Ardra Ivandika

Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

ardraivan@student.telkomuniversity.ac.id

2nd Dhoni Putra Setiawan

Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

setiawandhoni@telkomuniversity.ac.id

3rd Muhamad Reza

Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

muhamad.reza@gmail.com

Abstrak — Pemeliharaan kendaraan sering menjadi tantangan bagi pemilik, terutama dengan meningkatnya jumlah kendaraan yang memerlukan perawatan tepat waktu untuk menghindari kerusakan dan biaya perbaikan yang tinggi. Proyek ini memanfaatkan teknologi GPS untuk memantau lokasi dan status pemeliharaan kendaraan secara real-time, melalui aplikasi berbasis web yang dikembangkan dengan framework Laravel dan basis data Firebase. Data dari perangkat pelacak GPS GT06 dikumpulkan dan dianalisis untuk menentukan waktu optimal pemeliharaan kendaraan. Aplikasi ini di-deploy pada Google Cloud Platform untuk memastikan ketersediaan dan skalabilitas tinggi, memungkinkan pemilik kendaraan memantau kondisi dan menerima notifikasi perawatan kapan saja. Solusi ini menawarkan pendekatan efektif dalam meningkatkan efisiensi pengelolaan pemeliharaan kendaraan, baik untuk keperluan pribadi maupun komersial.

Kata kunci— Internet of Things, Tracking, Laravel, MySQL, Google Cloud Platform, GPS.

I. PENDAHULUAN

Kendaraan adalah aset penting dalam mendukung aktivitas masyarakat, dan peningkatan jumlahnya di Indonesia dalam 10 tahun terakhir membuat pemantauan kendaraan semakin mendesak. Tantangan dalam mengelola kendaraan secara efisien dapat diatasi melalui konsep *Intelligent Transportation System* (ITS), yang mengintegrasikan teknologi informasi, komunikasi, dan infrastruktur transportasi untuk meningkatkan efisiensi dan keamanan transportasi. Dalam konteks ini, ITS dapat mengintegrasikan modul GPS dan aplikasi untuk memantau kondisi serta kebutuhan pemeliharaan kendaraan secara real-time [1].

Penelitian ini mengembangkan aplikasi dashboard untuk melacak posisi kendaraan secara real-time sebagai bagian dari penerapan ITS. Data lokasi dikumpulkan menggunakan modul GPS dan NodeMCU, lalu disimpan di Firebase, yang dipilih karena kemampuannya menangani data real-time dan kemudahan integrasi. Aplikasi ini di-deploy menggunakan VM instance di Google Cloud Platform (GCP), dengan

Docker sebagai containerization tool untuk memastikan portabilitas dan konsistensi lingkungan.

Docker menawarkan kelebihan seperti kemudahan skala, manajemen aplikasi, dan efisiensi sumber daya. GCP dipilih karena keandalannya dalam menyediakan layanan seperti Compute Engine untuk VM instance dan Cloud Monitoring untuk pemantauan kinerja, yang memungkinkan sistem untuk diskalakan sesuai kebutuhan. Dengan kombinasi ini, penelitian menghadirkan solusi ITS yang efisien dan mudah dikelola, dengan fokus pada keandalan, skalabilitas, dan integrasi teknologi yang lebih baik..

II. TINJAUAN PUSTAKA

A. Internet of Things

Internet of Things merupakan konsep dimana dunia virtual teknologi informasi menyatu dengan benda riil di dunia nyata. Hal ini dimungkinkan dengan cara memberikan sensor tertentu pada suatu benda agar benda dapat menangkap *event* yang terjadi pada dunia nyata sebagai data dan kemudian dikirim ke sistem server. Hal ini memungkinkan pekerjaan dapat dilakukan tanpa campur tangan manusia [2].

B. Global Positioning System (GPS)

Global Positioning System (GPS) merupakan teknologi penentu lokasi yang banyak digunakan saat ini. GPS adalah suatu sistem radio navigasi penentuan lokasi menggunakan satelit [3].

C. Google Maps API

Google Maps adalah sebuah jasa peta virtual gratis dan online yang disediakan oleh Google. Google Maps dapat ditemukan di <https://google.co.id/maps/>. Google Maps API merupakan *interface* yang dapat diakses menggunakan Java agar Google Maps dapat ditampilkan di web atau aplikasi yang sedang kita bangun. Untuk dapat mengakses Google Maps, kita harus melakukan pendaftaran API Key terlebih dahulu dengan data pendaftaran berupa nama domain web yang kita bangun [4].

D. NodeMCU

NodeMCU ESP32 adalah platform IoT berbasis microcontroller ESP32 yang menawarkan performa lebih tinggi dibandingkan pendahulunya, ESP8266. ESP32 dilengkapi dengan dual-core processor, Wi-Fi, dan Bluetooth bawaan, memberikan fleksibilitas lebih dalam mengembangkan proyek-proyek IoT yang kompleks [5].

E. Arduino IDE

Arduino IDE adalah *software* yang digunakan untuk membuat *sketch* pemrograman atau dengan kata lain arduino IDE sebagai media untuk pemrograman pada *board* yang ingin diprogram. Arduino IDE ini berguna untuk mengedit, membuat, meng-*upload* ke *board* yang ditentukan, dan meng-*coding* program tertentu. Arduino IDE dibuat dari bahasa pemrograman JAVA, yang dilengkapi dengan *library* C/C++(*wiring*), yang membuat operasi *input/output* lebih mudah [6].

F. Modul GPS

Modul GPS adalah komponen penting dalam banyak aplikasi IoT yang membutuhkan pelacakan lokasi. Modul GPS bekerja dengan menerima sinyal dari beberapa satelit GPS yang berada di luar angkasa. Setiap satelit mengirim informasi waktu dan posisi satelit tersebut. Dengan menerima sinyal setidaknya empat satelit, modul GPS dapat menghitung posisi tiga dimensi (lintang, bujur, dan ketinggian). Hasil perhitungan ini adalah koordinat geografis yang dapat digunakan untuk menentukan lokasi perangkat.

G. Visual Studio Code

Visual Studio Code (VS Code) adalah aplikasi code editor untuk membantu proses pengembangan sebuah aplikasi. *Software* ini dikembangkan oleh perusahaan teknologi raksasa ternama, Microsoft. Meskipun begitu, VSCODE mendukung untuk dioperasikan pada perangkat selain Windows, seperti Linux dan Mac OS [7].

H. Laravel

Laravel merupakan framework PHP yang open-source dan berisi banyak modul dasar untuk mengoptimalkan kinerja PHP dalam pengembangan aplikasi web, apalagi PHP adalah bahasa pemrograman yang dinamis dan Laravel bisa membuat web development lebih cepat, lebih aman, dan lebih simpel [8].

I. PHP

PHP adalah singkatan dari "*Hypertext Preprocessor*" adalah bahasa pemrograman *server-side* yang sudah lama digunakan untuk pengembangan aplikasi web.

J. Javascript

Javascript adalah bahasa pemrograman yang awalnya dikembangkan untuk membuat halaman web lebih interaktif. Dengan Javascript, pengembang dapat menulis kode yang berjalan di sisi klien maupun di sisi server.

K. HTML

Hypertext Markup Language (HTML) adalah bahasa dasar yang digunakan untuk membuat struktur halaman web. HTML digunakan untuk membuat halaman web yang

menjadi antar muka bagi pengguna untuk berinteraksi dengan perangkat IoT.

L. CSS

Cascading Style Sheets (CSS) adalah bahasa yang digunakan untuk mendesain dan memformat tampilan halaman web yang ditulis dalam HTML.

M. XAMPP

XAMPP adalah perangkat lunak yang menyediakan lingkungan server local untuk pengembang web.

N. Firebase Realtime Database

Firebase Realtime Database adalah database yang dihosting di cloud. Data disimpan sebagai JSON dan disinkronkan secara realtime dengan setiap klien yang terhubung. Ketika Anda mem-build aplikasi lintas platform dengan SDK platform Apple, Android, dan JavaScript kami, semua klien akan menggunakan satu instance Realtime Database yang sama dan menerima perubahan data terbaru secara otomatis. [9]

O. Google Cloud Platform

Google Cloud Platform (GCP) adalah layanan Google yang menyediakan berbagai alat dan layanan untuk pengembangan, penyimpanan, dan analisis data.

P. Model-View-Controller (MVC)

MVC adalah pola desain perangkat lunak yang memisahkan aplikasi menjadi tiga komponen utama, yaitu *model* yang bertanggung jawab untuk mengelola database, *view* yang bertugas menampilkan data dan antarmuka pengguna, *controller* yang menjadi penghubung antara *model* dan *view*.

Q. Infrastructure as a Service (IaaS)

Infrastructure as a Service (IaaS) adalah salah satu model layanan komputasi awan yang menyediakan infrastruktur IT sebagai layanan yang dapat diakses secara online. IaaS memungkinkan pengguna untuk menyewa sumber daya komputasi seperti server, penyimpanan, dan jaringan melalui internet tanpa perlu memiliki atau mengelola infrastruktur fisik.

R. Group Inclusive Tour (GIT)

Git adalah sistem kontrol versi terdistribusi yang digunakan untuk melacak perubahan dalam kode sumber selama pengembangan perangkat lunak.

S. Containerisation

Containerization adalah teknologi yang memungkinkan pengembang untuk mengemas aplikasi dan semua dependensinya ke dalam satu paket yang disebut container.

T. Docker

Docker adalah platform yang memfasilitasi pengembangan, pengiriman, dan menjalankan aplikasi dalam container. Dengan Docker, pengembang dapat mengemas aplikasi beserta semua dependensinya ke dalam container, yang kemudian dapat dijalankan di mana saja tanpa perlu khawatir tentang perbedaan lingkungan.

U. Virtual Machine Instances

Virtual Machine (VM) Instance adalah versi digital dari komputer fisik. Perangkat lunak mesin virtual dapat menjalankan program dan sistem operasi, menyimpan data, terhubung ke jaringan, dan melakukan fungsi komputasi lainnya, dan memerlukan pemeliharaan seperti pembaruan dan pemantauan sistem. [10]

V. Compute Engine API

Compute Engine API adalah layanan yang disediakan oleh Google Cloud Platform (GCP) untuk mengelola sumber daya di Compute Engine secara otomatis melalui kode program.

W. OS Linux

Linux adalah sistem operasi komputer yang bebas dan sumber terbuka, didistribusikan di bawah lisensi GNU General Public License (GPL) yang memungkinkan pengguna untuk mengunduh, menginstal, dan mengubah kode sumber sistem operasi secara gratis. [11]

X. Ubuntu

Ubuntu terkenal sebagai OS yang memiliki sistem keamanan lebih baik dalam mencegah *malware* dibandingkan dengan salah satu pesaingnya, yakni Windows. Padahal, Ubuntu menjadi OS gratis tanpa lisensi. Tak heran, OS ini pun banyak diminati oleh banyak orang.

Y. NginX

Nginx adalah web server open-source yang sangat populer dan banyak digunakan untuk berbagai kebutuhan, mulai dari hosting situs web statis hingga menangani aplikasi web yang kompleks. Dirancang untuk kinerja tinggi dan efisiensi, Nginx dikenal karena kemampuannya menangani ribuan koneksi secara bersamaan dengan penggunaan sumber daya yang relatif rendah.

Z. Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah bahasa pemodelan secara visual yang digunakan untuk merancang, menganalisis, dan mendokumentasikan sistem perangkat lunak serta proses bisnis. UML menyediakan serangkaian diagram yang menggambarkan berbagai aspek dari sistem, mulai dari struktur dan hubungan antar komponen hingga interaksi pengguna dan alur proses. [12]

1. Activity Diagram

Activity diagram adalah alat visual dalam Unified Modeling Language (UML) yang digunakan untuk menggambarkan alur kerja atau proses dalam sistem perangkat lunak. Diagram ini menunjukkan urutan aktivitas yang dilakukan, serta keputusan dan interaksi antara aktivitas-aktivitas tersebut.

2. Visual Paradigm Online

Visual Paradigm Online adalah perangkat lunak untuk menggambar online gratis dengan dukungan untuk Activity Diagram, bagan organisasi, dan diagram UML lainnya

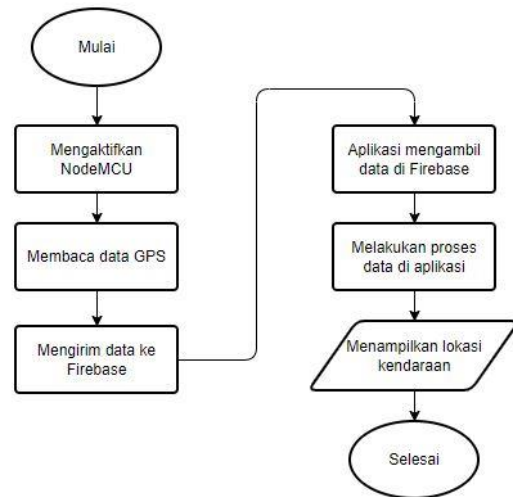
3. Wireshark

Wireshark adalah alat analisis jaringan yang digunakan untuk menangkap dan memeriksa lalu lintas data di jaringan secara mendalam.

III. Perancangan Sistem

A. Flowchart

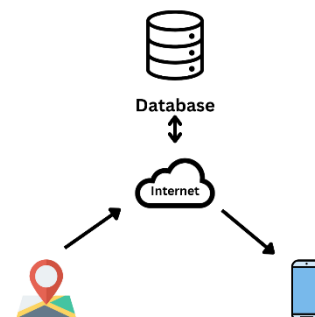
Flowchart dibuat sebagai acuan dalam melakukan langkah-langkah untuk implementasi sistem yang akan dibuat. Gambar 1 menunjukkan bentuk *flowchart*.



GAMBAR 1
(Flowchart)

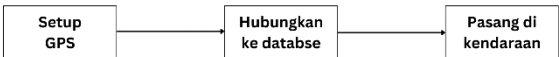
B. Desain Sistem

Desain sistem pemantauan posisi kendaraan menggunakan modul Internet of Things (IoT) yang terintegrasi ke aplikasi. Pertama, kendaraan akan dilengkapi dengan modul IoT yang terdiri dari GPS dan berbagai sensor untuk mengumpulkan data posisi dan kondisi kendaraan secara real-time. Data ini kemudian dikirimkan melalui jaringan seluler ke server pusat yang aman. Server pusat ini bertanggung jawab untuk menyimpan, memproses, dan menganalisis data yang diterima. Selanjutnya, informasi yang telah diproses disajikan melalui antarmuka aplikasi yang intuitif. Gambar 2 adalah gambaran untuk desain sistem.



GAMBAR 2
(Desain Sistem)

C. Skenario Perancangan GPS



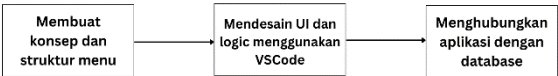
GAMBAR 3
(Perancangan GPS)

Gambar 3 menunjukkan skenario perancangan GPS dari mulai setup hingga pasang di kendaraan.

Perancangan GPS menggunakan modul GPS untuk mengambil data latitude dan longitude, serta menggunakan NodeMCU sebagai microcontroller untuk mengirimkan data yang didapatkan ke database.

Modul GPS dihubungkan dengan NodeMCU sebagai microcontroller yang akan mendapat perintah untuk mengirim data ke database. NodeMCU mendapat perintah dengan cara menambahkan code menggunakan perangkat lunak Arduino IDE.

D. Skenario Perancangan Aplikasi

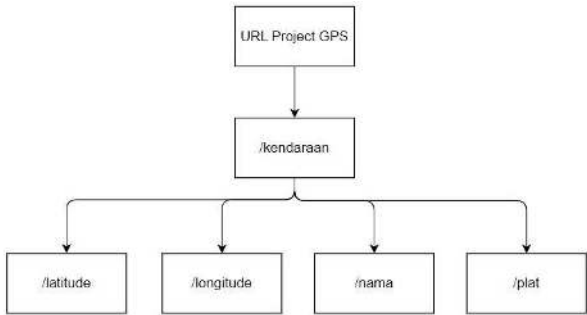


GAMBAR 4
(Perancangan Aplikasi)

Dalam merancang aplikasi menggunakan Laravel sebagai frame work, proses dimulai dari membuat konsep struktur menu. Langkah berikutnya adalah mendesain antarmuka pengguna (UI) menggunakan Visual Studio Code. Setelah desain UI selesai, tahap selanjutnya adalah menghubungkan aplikasi Laravel dengan database Firebase.

E. Desain Database

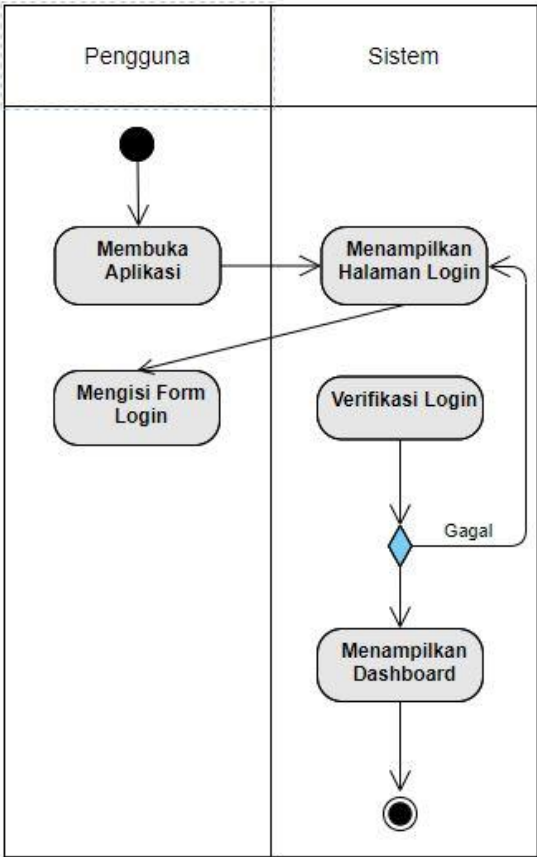
Database yang digunakan pada sistem yang dirancang adalah Firebase. Fitur Firebase yang digunakan untuk menampung data dari sensor GPS adalah Realtime Database. Gambar 3.4 menunjukkan struktur direktori penyimpanan di database.



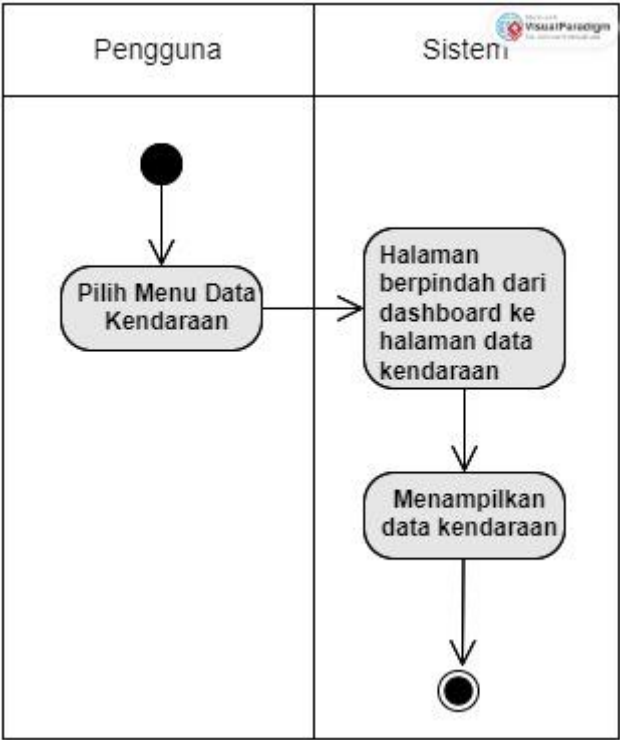
GAMBAR 5
(Struktur Direktori Database)

F. Pemodelan Activity Diagram

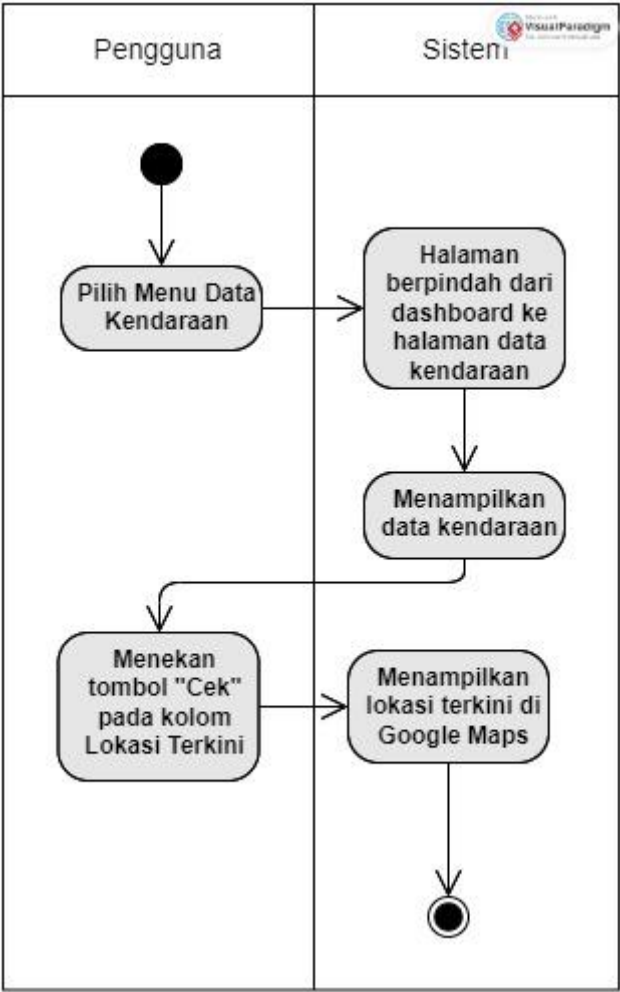
Activity Diagram dibuat berdasarkan fitur dari aplikasi yang diusulkan untuk membantu memahami proses yang terjadi antara pengguna aplikasi dan sistem. Activity Diagram dibuat menggunakan Visual Paradigm Online.



GAMBAR 6
(Activity Diagram Login)



GAMBAR 7
(Activity Diagram Cek Data Kendaraan)



GAMBAR 8
(Activity Diagram Cek Lokasi Kendaraan)

G. Desain Aplikasi

Aplikasi ini didesain menggunakan Visual Studio Code sebagai text editor, Laravel sebagai frame work, dan Bootstrap untuk styling.

H. Perancangan Deployment

Deployment pada aplikasi ini menggunakan Virtual Machine Instance dan Docker.



GAMBAR 9
(Perancangan Deployment)

I. Perancangan Virtual Machine

Perancangan Virtual Machine menggunakan infrastruktur Google Cloud Platform (GCP).

Name *
vm-tugas-akhir

MANAGE TAGS AND LABELS

Region *
asia-southeast2 (Jakarta)

Zone *
asia-southeast2-a

GAMBAR 10
(Memberi Nama VM)

Machine configuration

General purpose | Compute optimized | Memory optimized | Storage optimized | GPUs

Machine types for common workloads, optimized for cost and flexibility

Series	Description	vCPUs	Memory	Platform
C4	Consistently high performance	2 - 192	4 - 1,488 GB	Intel Emerald
N4	Flexible & cost-optimized	2 - 80	4 - 640 GB	Intel Emerald
C3	Consistently high performance	4 - 192	8 - 1,536 GB	Intel Sapphire
C3D	Consistently high performance	4 - 360	8 - 2,880 GB	AMD Genoa
E2	Low cost, day-to-day computing	0.25 - 32	1 - 128 GB	Based on av
N2	Balanced price & performance	2 - 128	2 - 864 GB	Intel Cascadi
N2D	Balanced price & performance	2 - 224	2 - 896 GB	AMD EPYC
T2A	Scale-out workloads	1 - 48	4 - 192 GB	Ampere Altra
T2D	Scale-out workloads	1 - 60	4 - 240 GB	AMD EPYC
N1	Balanced price & performance	0.25 - 96	0.6 - 624 GB	Intel Skylake

GAMBAR 11
(Konfigurasi VM)

Machine type

Choose a machine type with preset amounts of vCPUs and memory that suit most workloads. Or, you can create a custom machine for your workload's particular needs. [Learn more](#)

PRESET | CUSTOM

e2-medium (2 vCPU, 1 core, 4 GB memory)

vCPU: 1-2 vCPU (1 shared core)

Memory: 4 GB

ADVANCED CONFIGURATIONS

GAMBAR 12
(Memilih Tipe VM)

Boot disk

Name	vm-tugas-akhir
Type	New balanced persistent disk
Size	60 GB
Snapshot schedule	No schedule selected
License type	Free
Image	Ubuntu 20.04 LTS

GAMBAR 13
(Boot Disk)

Monthly estimate

\$40.69

That's about \$0.06 hourly

Pay for what you use: no upfront costs and per second billing

Item	Monthly estimate
2 vCPU + 4 GB memory	\$32.89
60 GB balanced persistent disk	\$7.80
Total	\$40.69

[Compute Engine pricing](#)

[LESS](#)

GAMBAR 14
(Harga Virtual Machine)

J. Skenario Pengujian Alat

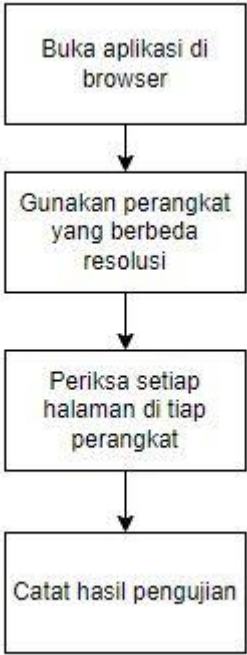
Subbab ini membahas skenario pengujian keakuratan sensor GPS berdasarkan data yang diambil dan dikirim ke database dengan lokasi di lapangan dimana data tersebut diambil.

K. Skenario Pengujian Aplikasi

Subbab ini membahas skenario pengujian aplikasi berdasarkan aspek functional suitability dan compatibility. Pengujian ini dilakukan untuk memastikan aplikasi yang dirancang berjalan dengan baik.

Tabel 1
(Tabel *Functional Suitaility*)

No.	Fungsi	Hasil yang diharapkan
1	Login	Aktivitas Login pada aplikasi berjalan dengan benar
2	Menu Data Kendaraan	Aktivitas di menu Data Kendaraan berjalan dengan benar.
3	Cek Lokasi	Aktivitas cek lokasi di Maps sesuai dengan data dari database.
3	Logout	Aktivitas Logout pada aplikasi berjalan dengan benar.



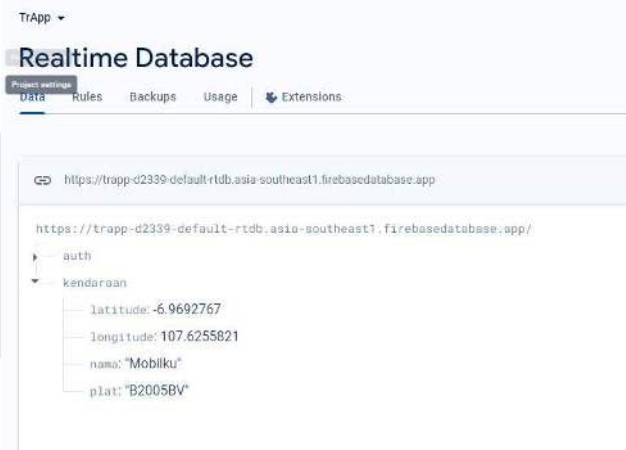
GAMBAR 15
(Skenario Pengujian *Responsive Layout*)

IV. HASIL DAN PEMBAHASAN

A. Implementasi

Subbab ini membahas implementasi sistem pemantauan posisi kendaraan menggunakan modul IoT yang terintegrasi ke aplikasi dashboard dengan menggunakan Firebase sebagai database. Database menerima data yang berupa latitude dan longitude yang didapat dari sensor GPS NEO 6N.

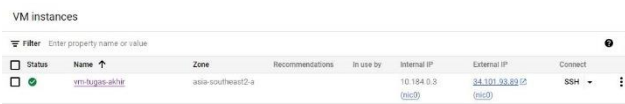
Penyimpanan data Latitude dan Longitude di Firebase menggunakan fitur realtime database.



GAMBAR 16
(Data Latitude dan Longitude di Firebase)

Pada aplikasi yang sudah dideploy menggunakan Docker, JSON file yang berisi private key digunakan oleh Laravel untuk melakukan otentikasi terhadap Firebase, sehingga Laravel dapat berinteraksi dengan database Firebase.

Deployment aplikasi dilakukan di Virtual Machine dengan menggunakan Google Cloud Platform (GCP).



GAMBAR 17
(Virtual Machine)

B. Pengujian Aplikasi

Pengujian aplikasi ini aplikasi berdasarkan aspek functional suitability dan compatibility.

Tabel 2
(Tabel Hasil Pengujian *Functional Suitaility*)

No.	Fungsi	Hasil
1	Login	Aktivitas Login pada aplikasi sudah berjalan dengan benar
2	Menu Kendaraan	Aktivitas di menu Data Kendaraan sudah berjalan dengan benar.
3	Cek Lokasi	Aktivitas cek lokasi di Maps sudah sesuai dengan data dari database.
3	Logout	Aktivitas Logout pada aplikasi sudah berjalan dengan benar.

Pengujian Responsive Layout di perangkat berbeda mendapatkan hasil yang baik dan tidak ada masalah.

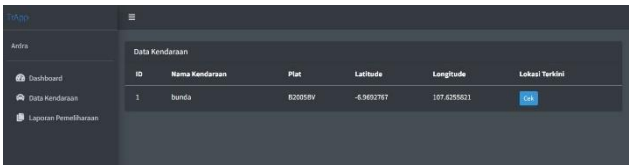
TABEL 3
(Tabel Hasil Pengujian *Responsive Layout*)

No	Perangkat	Ukuran Layar	Hasil
1	Laptop Lenovo X270	12.5 inci	Berhasil
2	Hand Phone Redmi Note 11	6.43 inci	Berhasil

Gambar 17 dan Gambar 18 menampilkan *Layout* aplikasi pada saat dibuka di laptop.



GAMBAR 17
(Layout Dashboard di Desktop)

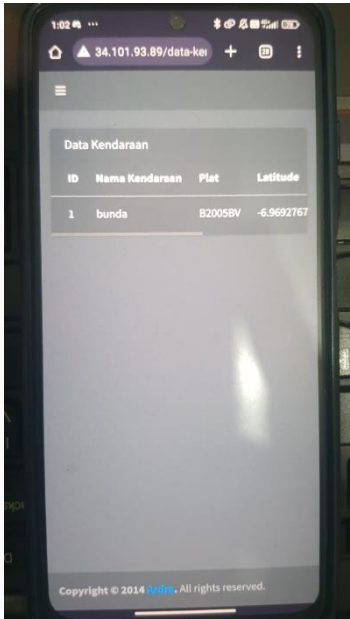


GAMBAR 18
(Layout Data Kendaraan di Desktop)

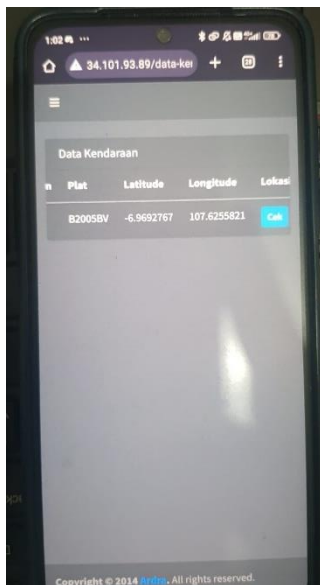
Adapun Gambar 19 sampai Gambar 21 menampilkan *Layout* aplikasi pada saat dibuka di ponsel.



GAMBAR 19
(Layout Dashboard di Handphone)



GAMBAR 20
(Layout Data Kendaraan di Handphone)



GAMBAR 21

(Layout Data Kendaraan di Handphone)

Pengujian Response Time Pingdom Website Speed Test adalah untuk mengukur seberapa cepat aplikasi merespons permintaan dari pengguna. Pengujian ini memastikan bahwa aplikasi dapat memberikan pengalaman yang baik untuk pengguna dengan waktu tunggu yang minimal.

Pengukuran response time halaman Dashboard saat dilakukan menggunakan tools Pingdom Website Speed Test mendapatkan hasil seperti di gambar 22 dan gambar 23.



GAMBAR 22

(Response Test Dashboard)



GAMBAR 23

(Response Test Halaman Data Kendaraan)

C. Throughput

Dari pengecekan di Wireshark, total paket yang ter-captured ada 27096 paket, dan time span, s adalah 1657,438. Maka dapat dihitung dengan rumus :

$$\text{Throughput} = \frac{27096 \text{ packets}}{1657.438 \text{ seconds}}$$

Maka Throughput-nya adalah sekitar 16.34 paket perdetik.

D. Hasil Deployment

Proses Deployment ke Virtual Machine yang berada di Cloud berhasil dengan baik

Basic information

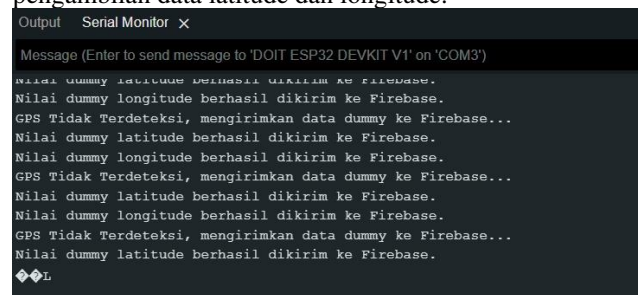
Name	vm-tugas-akhir
Instance Id	4987489608312221244
Description	None
Type	Instance
Status	Running
Creation time	Aug 25, 2024, 10:42:12 PM UTC+07:00
Zone	asia-southeast2-a

GAMBAR 24

(Informasi Virtual Machine)

E. Hasil Pengambilan Data Lokasi

Alat berhasil dirancang sesuai dengan rencana. NodeMCU sudah berhasil terkoneksi dan sudah bisa mengirim data ke database. Namun, modul GPS tidak berfungsi saat beberapa kali melakukan percobaan pengambilan data latitude dan longitude.



GAMBAR 25

(Output Arduino IDE)

F. Pengujian Sensor Ultrasonik HC-SR04

Hasil dari pengujian fungsionalitas alat dapat bekerja sesuai dengan yang diharapkan. Pada pengukuran sensor ultrasonik HC-SR04 diset pada 5cm adalah batas tempat sampah penuh, sedangkan sensor gas MQ2 dan sensor api dikalibrasi menggunakan potensiometer pada sensor.

Tabel 1

(Tabel hasil pengujian sensor ultrasonik HC-SR04)

V. KESIMPULAN

Pada Bab 5 ini, dapat disimpulkan bahwa perancangan sistem pemantau kendaraan menggunakan aplikasi dashboard berbasis Laravel sebagai frame work dan Firebase sebagai database berhasil diimplementasikan. Aplikasi mampu menjalankan fungsionalitas yang diinginkan. Fitur responsifnya memastikan bahwa aplikasi dapat berfungsi dengan baik di berbagai perangkat yang berbeda-beda ukuran layar. Namun GPS tidak bisa mengambil data dikarenakan modul GPS yang tidak berfungsi, sehingga data yang dikirim oleh microcontroller adalah data dummy.

Proses deployment aplikasi di Virtual Machine Instance menggunakan fitur Google Cloud Platform (GCP) juga menunjukkan hasil yang baik. Penggunaan VM-Instance memberikan efisiensi dan bisa memberikan kemudahan

dalam mengelola server, serta memberikan harga sesuai dengan penggunaan yang kita inginkan.

Selain itu, aplikasi juga menunjukkan waktu respons yang cukup baik, memastikan pengguna dapat berinteraksi dengan sistem secara real-time tanpa harus ada waktu menunggu yang lama.

Throughput yang didapatkan adalah 16.34 paket perdetik. Hasil ini sudah cukup memadai karena pemakaian data yang rendah pada aplikasi IoT ini. Aplikasi ini juga tidak banyak proses data antara database dan aplikasi. Pada saat penggunaan aplikasi juga tidak ada kendala dan aplikasi juga memiliki respon yang cukup baik.

REFERENSI

- [A. K. Sugito, "Intelligent Transportation System (ITS) ; 1 [Solusi Permasalahan Transportasi Perkotaan]," 13 Agustus 2017. [Online]. Available: <http://awankurniawansugito.blogspot.com/2017/08/intelligent-transportation-system-its.html>.
- [A. Ansori, "Studi Pemanfaatan Internet of Things dan 2 Data Mining untuk Pengawasan Bahan Bakar Minyak (Studi Kasus: Perusahaan Pelayaran Penumpang Nasional)," *Wave*, vol. XII, pp. 31-42, 2018.
- [A. Perdananto, "Sistem Pelacak Menggunakan GPS 3 Tracker Untuk Ponsel Android," *J. ICT Akad*, vol. 8, no. 15, p. 59, 2017.
- [F. Mahdia and F. Noviyato, "Pemanfaatan Google Maps 4 API untuk Pembangunan Sistem Informasi Manajemen Bantuan Logistik Pasca Bencana Alam Berbasis Mobile Web," *Jurnal Sarja Teknik Informatika*, vol. 1, no. 1, pp. 164-165, 2013.
- [K. Y. Maulana, "Apa itu ESP32, Salah Satu Modul Wifi 5 Populer," *Anak Teknik Indonesia*, 30 Desember 2022. [Online]. Available: <https://www.anakteknik.co.id/krysnayudhamaulana/articless/apa-itu-esp32-salah-satu-modul-wi-fi-poppuler?srsId=AfmBOoKhRpy-4Zmx8zNOIXDtTYqIcArx0SLRtTO0tRhF2bdtZ-d-m>. [Accessed 29 Agustus 2024].
- [Erintafifah, "Mengenal Perangkat Lunak Arduino IDE," 6 KMTek, 8 Oktober 2021. [Online]. Available: <https://www.kmtech.id/post/mengenal-perangkat-lunak-arduino-ide>.
- [E. Santi, "VSCODE Adalah," IDWebhost, 20 April 2024. 7 [Online]. Available: <https://idwebhost.com/blog/vscode-adalah/>.
- ["Apa Itu Laravel," BiznetGio, 17 Agustus 2022. [Online]. 8 Available: <https://www.biznetgio.com/news/apa-itu-laravel>.
- ["Firebase Realtime Database," Firebase, [Online]. 9 Available: <https://firebase.google.com/docs/database?hl=id#:~:text=Firebase%20Realtime%20Database%20adalah%20database,dengan%20setiap%20klien%20yang%20terhubung..> [Accessed 29 Agustus 2024].
- [A. Panatagama, "Apa itu Virtual machine dan fungsinya 1 yang perlu Anda ketahui," TERRALOGIC, 28 Oktober 0 2021. [Online].
- [H. R. P. Sailallah, "Pengertian OS Linux," Telkom 1 University, 28 April 2023. [Online]. Available: 1 <https://it.telkomuniversity.ac.id/pengertian-os-linux/>.
- [D. Intern, "Apa itu UML? Beserta Pengertian dan 1 Contohnya," Dicoding, 12 Mei 2021. [Online]. Available: 2 <https://www.dicoding.com/blog/apa-itu-uml/>.
- [P. Kardono, "Integrated Solid Waste Management in 1 Indonesia," *Proceedings of International symposium on 3 EcoTopia Science 2007. ISET07*, pp. 629-633, 2007.
- [M. Burgess, "What is the Internet of Things?," WIRED, 1 16 February 2018. [Online]. Available: 4 <https://www.wired.co.uk/article/internet-of-things-what-is-explained-iot>. [Accessed 9 October 2020].
- [Ecolink, Tentang sampah adalah suatu bahan yang 1 terbuang atau dibuang dari sumber hasil aktivitas manusia 5 maupun proses alam yang belum memiliki nilai] ekonomis., Jakarta, Indonesia, 1996.
- [M. F. R. S. Z. Nurul Hidayati Lusita Dewi, 1 "PROTOTYPE SMART HOME DENGAN MODUL 6 NODEMCU ESP8266 BERBASIS INTERNET OF] THINGS (IOT)," 2019.
- [H. Santoso, "Cara Kerja Sensor Ultrasonik, Rangkaian, & 1 Aplikasinya," ElangSakti, 2015. [Online]. Available: 7 [http://www.elangsakti.com/2015/05/sensorultrasonik.ht](http://www.elangsakti.com/2015/05/sensorultrasonik.html)] ml. [Accessed 2020].
- [K. A. Smith, Introduction to Gas Sensors, CRC Press, 1 2012. 8]
- [B. J. S. G. P. H. A. M. A.-M. G. A. Akpakwu, A Survey 1 on 5G Networks for the Internet of Things: 9 Communication Technology and Challenges, IEEE] Access, 2017.
- [S. T. W. I. E. N. M. W. a. C. N. S. Khedkar, Real Time 2 Database for Applications, 2017. 0]
- ["Memilih Database: Cloud Firestore atau Realtime 2 Database | Firebase," [Online]. Available: 1 [https://firebase.google.com/docs/firestore/rtdb-vs-](https://firebase.google.com/docs/firestore/rtdb-vs-firestore?hl=id)] [firestore?hl=id](https://firebase.google.com/docs/firestore/rtdb-vs-firestore?hl=id). [Accessed 02 aug 2021].