

Prototype Real-Time Monitoring System Bus Trans Jateng Berbasis Android untuk Informasi Waktu Kedatangan Bus di Halte

Felia Citra Dwiyani Putri Rosyadi^{#1}, Yudha Saintika^{#2}, Yoso Adi Setyoko^{#3}

¹Program Studi S1 Teknik Informatika, ² Program Studi S1 Sistem Informasi, ³ Program Studi S1 Teknik Informatika

^{1,2,3} Institut Teknologi Telkom Purwokerto
Jl D.I Panjaitan No 128 Purwokerto, Jawa Tengah

¹ 17102128@ittelkom-pwt.ac.id

² yudha@ittelkom-pwt.ac.id

³ yoso@ittelkom-pwt.ac.id

accepted on 10-06-2021

Abstract

Pada tahun 2018 Gubernur Jawa Tengah yaitu Ganjar Pranowo me-*launching* moda transportasi modern berupa bus Trans Jateng yang beroperasi di rute Purwokerto-Purbalingga dan sebaliknya. Pihak pelayanan bus Trans Jateng telah mengeluarkan aplikasi Bus Trans Jateng yang bernama “Si Anteng”. Namun, berdasarkan hasil *launching* pada tanggal 7 Maret 2021, aplikasi tersebut belum dilengkapi notifikasi ketika bus telah tiba di halte, dan ketika pengguna memilih bus terdekat, pengguna tidak dapat memantau pergerakan bus. Faktor belum adanya jalur khusus Trans Jateng mengakibatkan sering terganggunya perjalanan bus Trans Jateng dengan kendaraan lain seperti kemacetan lalu lintas, dan faktor lainnya seperti kerusakan armada di jalan menjadi penyebab kedatangan bus Trans Jateng di halte seringkali tidak sesuai dengan jadwal yang sudah ditetapkan oleh pihak BRT. Sehubungan dengan masalah tersebut, peneliti membuat sebuah sistem dan alat untuk *monitoring* bus Trans Jateng dengan metode GPS tracking. Pada sistem ini menggunakan mikrokontroler Arduino uno dan Modul SIM808 untuk menerima data GPS dan dapat diakses melalui aplikasi android. Berdasarkan hasil pengembangan yang dilakukan dengan metode *prototyping* diperoleh hasil uji *black box* menyatakan sebesar 93,8% aplikasi berjalan sesuai fungsinya, dan berdasarkan hasil uji keakuratan diperoleh perbandingan waktu telat aplikasi *monitoring* bus Trans Jateng memunculkan notifikasi setelah bus Trans Jateng tiba di halte yaitu 16 detik, dan waktu tercepat 5 detik, serta rata-rata dari setiap rute diantaranya rute Purwokerto-Purbalingga sebesar 10 detik, dan pada rute Purbalingga-Purwokerto sebesar 9 detik. Dengan menggunakan sistem *monitoring* ini dapat mengefesiansikan waktu tunggu, serta dapat memotivasi masyarakat agar memilih untuk menggunakan transportasi umum.

Kata kunci: Bus Trans Jateng, Arduino uno, Modul SIM808, *GPS tracking*, *Smartphone* Android

I. PENDAHULUAN

Peningkatan populasi manusia terutama di daerah perkotaan menyebabkan kebutuhan akan alat transportasi dari waktu ke waktu terus meningkat. Pada tahun 2018 Gubernur Jawa Tengah yaitu Ganjar Pranowo me-*launching* moda transportasi modern berupa Bus Trans Jateng. Adapun *Bus Rapid Transit*

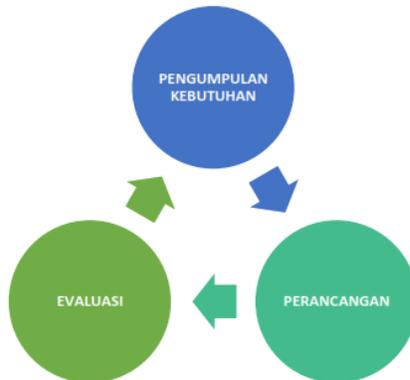
(BRT) ini akan beroperasi dengan rute Purwokerto-Bukateja (Purbalingga) dan sebaliknya [1]. Moda transportasi bus Trans Jateng diharapkan mampu memenuhi kebutuhan masyarakat untuk memperoleh angkutan yang cepat, efektif dan murah.

Pihak pelayanan bus Trans Jateng juga telah berkolaborasi dengan Dinas Perhubungan Jawa Tengah agar aplikasi Bus Trans Jateng yang bernama “Si Anteng” diperluas dari yang awalnya terdiri dari koridor Bawen-Tawang, Bahurekso-Mangkang, Tirtonadi-Sumberlawang, dan Kutoarjo-Borobudur, sekarang ditambah dengan koridor baru yaitu Bulupitu-Bukateja (Purwokerto-Purbalingga) yang telah dirilis aplikasinya pada tanggal 7 Maret 2021. Namun, pada aplikasi tersebut belum dilengkapi notifikasi ketika bus telah datang di halte pengguna, dan ketika pengguna memilih bus terdekat, pengguna tidak dapat *monitoring* pergerakan bus. Faktor belum adanya jalur khusus Trans Jateng yang mengakibatkan sering terganggunya perjalanan bus Trans Jateng dengan kendaraan lain seperti kemacetan lalu lintas, dan faktor lainnya yang tidak menentu seperti kerusakan armada di jalan, maupun insiden tidak terduga menjadi penyebab kedatangan bus Trans Jateng di halte seringkali tidak sesuai dengan jadwal yang sudah ditetapkan oleh pihak BRT.

Sehubungan dengan masalah tersebut, peneliti membuat sebuah sistem dan alat untuk *monitoring* bus Trans Jateng dengan metode GPS tracking. Pada sistem ini menggunakan mikrokontroler Arduino uno dan Modul SIM808 untuk menerima data GPS dan mendukung konektivitas GPRS sehingga data dapat dikirim ke server dan dapat diakses melalui aplikasi android. Sistem kemudian akan dikembangkan menggunakan metode metode prototyping. Dengan menggunakan sistem *monitoring* ini diharapkan dapat mengefesienkan waktu tunggu, meningkatkan sarana prasarana transportasi publik, serta memotivasi masyarakat agar memilih untuk menggunakan transportasi umum.

II. TINJAUAN PUSTAKA

Dalam penelitian ini penulis menggunakan metode prototyping, dimana dalam pengembangan sistem menggunakan metode ini memiliki beberapa tahapan sebagai berikut:



Gambar 1. Model prototype

Berdasarkan gambar diatas, terdapat beberapa proses dalam *prototyping*, adapun pembahasannya sebagai berikut:

a) Pengumpulan kebutuhan

Pengembang software bersama pelanggan mendefinisikan format seluruh perangkat lunak, semua kebutuhan, dan garis besar sistem yang akan dibuat.

b) Perancangan

Perancangan ini menjadi dasar pembuatan prototype dimana perancangan dilakukan dengan cepat dan mewakili semua aspek perangkat lunak yang diketahui.

c) Evaluasi *prototype*

Pengembang dan pelanggan akan mengevaluasi prototype yang dibuat guna memperjelas kebutuhan perangkat lunak [2].

Kemudian dalam penelitian ini dilakukan pengumpulan referensi dan teori-teori yang relevan yang berkaitan diantaranya:

2.1. Arduino uno

Arduino uno merupakan salah satu dari jenis-jenis *Arduino* yang ada, dimana *Arduino uno* ini dibekali dengan mikrokontroler ATmega328 dan versi R3 adalah versi terakhir yang di buat. *Arduino uno* R3 mempunyai 14 pin digital input *output*, dimana 6 pin digunakan sebagai *output* PWM, 6 pin analog input, 2x3 pin ICSP untuk pemrograman *Arduino* dengan software lain, clockspeed 16 Mhz, dan kabel USB. Pin-pin tersebut berisi semua yang dibutuhkan sebagai pendukung mikrokontroler dengan menghubungkan kabel USB ke komputer atau menggunakan adaptor 5 VDC [3].

Bahasa pemrograman *Arduino* adalah menggunakan Bahasa C yang sudah disederhanakan sehingga mempermudah pengguna dalam mendalami dan mempelajarinya. Adapun deskripsi *Arduino uno* adalah seperti pada Tabel 1.

TABEL 1 Deskripsi Arduino

Mikrokontroler	ATMega328
Tegangan Pengoperasian	5V
Batas Tegangan input	6-20V
Pin Input/Output digital	14 pin digital (6 pin menyediakan keluaran PWM)
Jumlah pin input analog	6 pin
Arus DC tiap pin I/O	40 mA
Arus DC pin 3,3 V	50 mA
Memory Flash	32 KB
SRAM	2 KB
EPROM	1 KB
Clock Speed	16 MHZ

2.2. Global Positioning System (GPS)

Global Positioning System (GPS) adalah sebuah sistem atau alat yang digunakan untuk menginformasikan dimana pengguna berada secara global di permukaan bumi dengan berbasis satelit. Data yang dikirim dari satelit yaitu berupa data digital dari sinyal radio[4].

2.3. General Packet Radio Service (GPRS)

General Packet Radio Service (GPRS) merupakan layanan bukan suara (non-voice) dimana informasi dikirimkan dan diterima melalui jaringan telepon. Layanan ini melengkapi teknologi komunikasi yang sudah ada sebelumnya yaitu *short message service* (SMS) dan circuit switched data (CSD). GPRS salah satu standar komunikasi data pada jaringan GSM yang mempunyai kecepatan transfer dari 56 kbps sampai 115 kbps. Sistem GPRS dapat digunakan untuk mengirim data yang berhubungan dengan e-mail, *Wireless Application Protocol* (WAP), data gambar (MMS), dan *World Wide Web* (WWW) (Falih, and Ramadan, 2020).

Dari segi biaya, pembayaran bergantung pada banyaknya penggunaan yang mana penggunaan tersebut berupa banyaknya byte yang diterima atau dikirim. Sehingga dengan laju bit yang tinggi namun dengan tarif rendah membuat lebih banyak pelanggan yang memilihnya (Falih, and Ramadan, 2020).

2.4. Modul SIM808

Modul SIM808 digunakan untuk komunikasi secara jaringan seluler GPRS (Internet), selain itu modul SIM808 juga terdapat sensor lokasi A-GPS (indoor/outdoor) yang bisa berkomunikasi dengan satelit di dalam gedung ataupun di area terbuka sehingga pada sistem ini modul SIM808 juga digunakan untuk mendapatkan nilai koordinat latitude dan longitude. modul SIM808 dihubungkan dengan *Arduino* melalui pin Rx dan Tx, juga pada pin LI-ion [6].

2.5. Haversine Formula

Teorema *Haversine* Formula merupakan sebuah persamaan yang sangat berguna dalam bidang navigasi, yaitu untuk mencari jarak busur antara dua titik pada bola dari latitude dan longitude. Hal ini merupakan

bentuk persamaan khusus dari trigonometri bola, *law of haversine* mencari hubungan sisi dan sudut pada segitiga dalam bidang bola [7].

pada penelitian ini untuk mencari jarak terdekat dari dua titik adalah sebagai berikut [8]:

$$d = R * c \quad (1)$$

$$R = 6,371 \text{ km} \quad (2)$$

$$c = 2 * \text{atan2}(\sqrt{a}, \sqrt{(1-a)}) \quad (3)$$

$$a = \sin^2(\Delta\text{lat}/2) + \cos(\text{lat}1) * \cos^2(\text{lat}2) * \sin^2(\Delta\text{long}/2) \quad (4)$$

$$\Delta\text{lat} = \text{lat}2 - \text{lat}1 \quad (5)$$

$$\Delta\text{long} = \text{long}2 - \text{long}1 \quad (6)$$

Dari perhitungan diatas digunakan untuk menentukan jarak paling dekat harus dapat perbandingan. Jadi harus terdapat banyak titik lokasi dan ditemukan banyak jarak, kemudian dibandingkan dengan jarak terpendek. Untuk membatasi maka digunakan radius agar wilayah pada peta tidak melebar.

III. METODE PENELITIAN

Pada penelitian ini diperlukan alat dan bahan yang digunakan diantaranya:

a) Perangkat Keras

Perangkat keras yang digunakan dalam pengembangan penelitian ini adalah sebagai berikut:

- a. Laptop HP.
- b. Prosesor AMD A9.
- c. Memory 8 GB DDR4.
- d. Storage 1 TB SATA HDD 5400RPM.
- e. *Arduino uno*, sebagai pengolah data masukan dari modul SIM808.
- f. Modul SIM808, untuk GPS dan GPRS.
- g. Kabel USB.
- h. *Smartphone* Samsung Galaxy Grand Prime untuk uji coba aplikasi

b) Perangkat Lunak

Perangkat lunak yang dibutuhkan dalam pengembangan penelitian ini adalah sebagai berikut:

- a. Sistem Operasi Windows 10.
- b. *Browser*.
- c. Visual Studio Code, sebagai editor untuk membangun aplikasi android.
- d. *Arduino IDE*, untuk melakukan konfigurasi sebuah program untuk mikrokontroler *Arduino uno*.
- e. *Firebase*, sebagai penyimpanan data secara *cloud*.

3.1. Pengumpulan Data

Pada bagian ini, peneliti telah memperoleh data yang didapatkan dari hasil wawancara dengan pihak kantor pelayanan BRT Trans Jateng untuk dapat diimplementasikan kedalam sistem. Data tersebut terdiri dari data jadwal keberangkatan bus Trans Jateng, jumlah armada bus Trans Jateng koridor 1 Purwokerto-Purbalingga, dan koordinat titik halte yang dilalui bus Trans Jateng.

3.2. Analisis Kebutuhan Sistem

Pada analisis kebutuhan sistem dibagi menjadi dua yaitu kebutuhna fungsional dan kebutuhan non fungsional sebagai berikut:

a) Kebutuhan fungsional

Kebutuhan fungsional pada sistem *monitoring* bus Trans Jateng diantaranya sebagai berikut:

1. Aplikasi dapat menampilkan estimasi waktu yang diperlukan untuk mengetahui bus Trans Jateng akan tiba di halte pengguna.
2. Aplikasi dapat menampilkan rute yang dilalui bus Trans Jateng.
3. Aplikasi dapat memilih halte terdekat yang dilewati bus Trans Jateng dari lokasi pengguna.

b) Kebutuhan nonfungsional

Kebutuhan nonfungsional dari sistem *monitoring* bus Trans Jateng yaitu sebagai berikut:

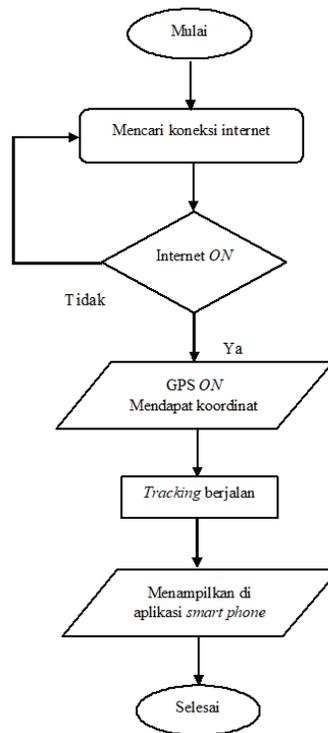
1. Aplikasi memiliki antarmuka yang mudah dipahami,
2. Aplikasi memiliki kemudahan dalam memenuhi kebutuhan pengguna,

3.3. Perancangan Sistem

Perancangan sistem pada penelitian ini terdiri dari beberapa tahap diantaranya yaitu:

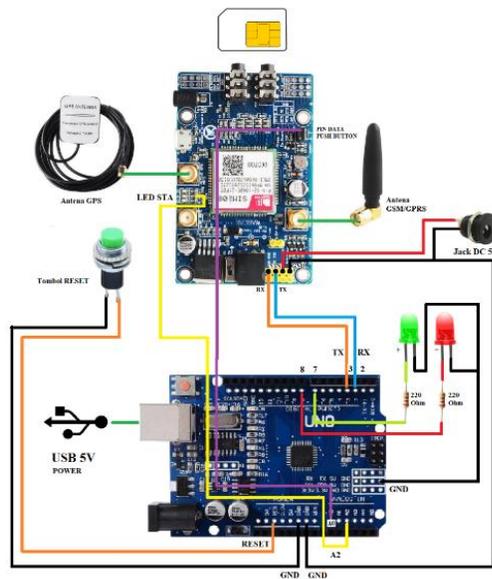
a) Perancangan GPS *Tracking*

Pada gambar 2 merupakan flowchart dari alur kerja dari GPS *tracking*:



Gambar 2. Flowchart sistem GPS *tracking*

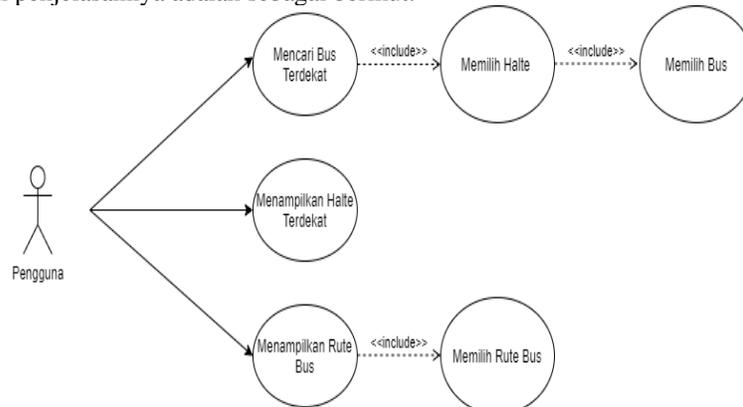
Berdasarkan flowchart alur sistem GPS *tracking* diatas adalah diawali dari bus Trans Jateng yang telah dipasang sistem *monitoring* kemudian akan mencari koneksi internet, apabila koneksi internet telah tersambung maka GPS akan aktif dan mendapatkan serta mengirimkan koordinat selama *tracking* berjalan, selanjutnya akan ditampilkan pada aplikasi *smartphone* android. Pada gambar 3 di bawah ini merupakan gambar dari rangkaian GPS *tracking* yang dibangun menggunakan *Arduino uno* dan Modul SIM808:



Gambar 3. Rangkaian GPS tracking

b) Perancangan aplikasi android

Adapun perancangan sistem aplikasi android *monitoring* bus Trans Jateng dirancang dengan menggunakan UML yang terdiri dari analisis *use case*, dan analisis kebutuhan *database* berdasarkan format dari *Firebase*. Adapun penjelasannya adalah sebagai berikut:

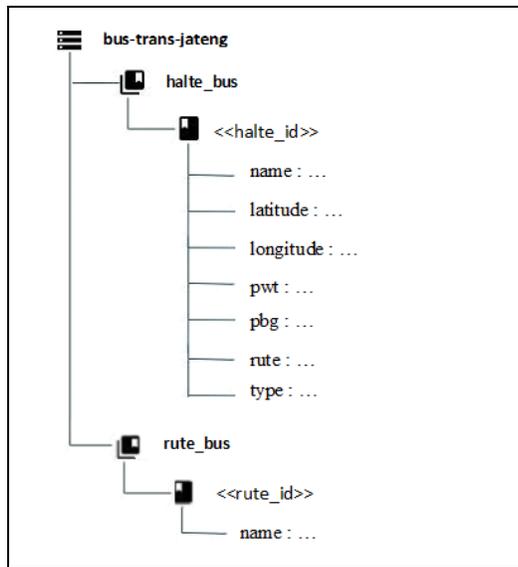


Gambar 4. Use case diagram

Pada Gambar 4. menunjukkan *use case* dari aplikasi *monitoring* bus Trans Jateng. *Use case* ini memiliki satu aktor yaitu pengguna layanan bus trans jateng. Untuk *use case* mencari bus terdekat memiliki relasi dengan *use case* memilih halte terdekat dengan tipe include, yang berarti seluruh pengguna harus memilih halte dimana pengguna berada terlebih dahulu agar dapat melihat bus terdekat. Setelah itu aktor harus memilih bus terdekat agar aktor selanjutnya dapat melihat pergerakan bus menuju halte dimana aktor berada tersebut.

Kemudian terdapat *use case* menampilkan halte terdekat dimana didalam *use case* ini actor dapat melihat seluruh halte dan dapat memilih halte yang terdekat. Pada *use case* menampilkan rute bus didalamnya terdapat *use case* memilih rute bus, dimana aktor di berikan pilihan untuk melihat rute bus mana yang akan dipilih.

Adapun Struktur *Firebase* yang peneliti bangun adalah sebagai berikut seperti pada gambar 5:

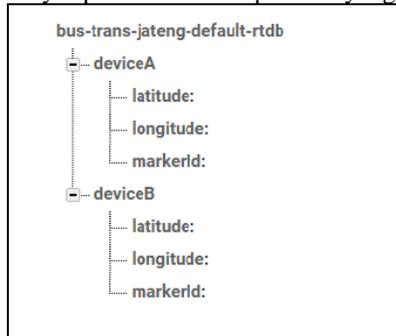


Gambar 5. Struktur *Firestore* menggunakan *Cloud Firestore*

Pada koleksi `halte_bus` akan berisi dokumen `halte_id` yang menggunakan format “halte_<AutoIncrementNumber>”, contohnya “halte_01”, dengan isi berbagai key dan isinya, antara lain:

- 1) name : menyimpan nama halte bus
- 2) latitude : menyimpan koordinat latitude halte bus
- 3) longitude : menyimpan koordinat longitude halte bus
- 4) type : menyimpan tipe bus, contohnya “Purwokerto-Purbalingga”
- 5) pbg : menyimpan halte urutan keberapa di rute pwt
- 6) pwt : menyimpan halte urutan keberapa di rute pbg
- 7) rute : menyimpan kode rute, contohnya “pbg” dan “pwt”

Pada koleksi `rute_bus` akan berisi dokumen `rute_id` yang menggunakan nama dari rute contohnya “pwt” dan “pbg”, dengan isi key name yang menyimpan nama dari tipe halte yang dilalui bus tersebut.



Gambar 6. Struktur *Firestore* menggunakan *Cloud Realtime Database*

Pada *cloud realtime database Firestore* data yang disimpan berupa JSON kemudian disinkronkan secara *realtime* dan disimpan kedalam turunan *database* secara bertingkat. Dikarenakan peneliti hanya menggunakan 2 alat GPS dan masing-masing GPS akan disimpan ke *child* deviceA dan deviceB. Setiap *child* didalamnya terdapat data field yang disimpan kedalam field latitude, longitude, dan markerId. Dimana markerId akan diisi dengan plat nomor dari delegasi bus trans jateng.

3.4. Pengembangan Sistem dengan Metode *Prototyping*

Tahap pembangunan sistem menggunakan metode *prototyping* ini memiliki beberapa tahap sebagai berikut:

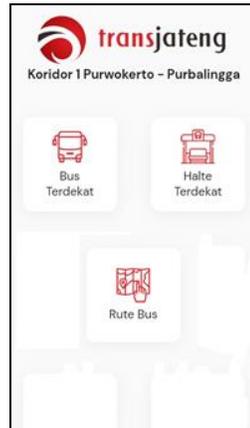
1) Mendengarkan *feedback* pengguna

Pada tahap mendengarkan *feedback* pengguna, dilakukan interaksi antar pengguna bersama dengan peneliti dengan menggunakan metode wawancara untuk mengembangkan *prototype* aplikasi. Untuk setiap iterasi dilakukan berdasarkan perancangan aplikasi yang telah disusun.

2) Pengembangan *software*

Dibawah ini merupakan hasil dari aplikasi *monitoring* bus trans jateng setelah dilakukan tahapan metode prototyping berdasarkan hasil iterasi satu sampai iterasi tiga:

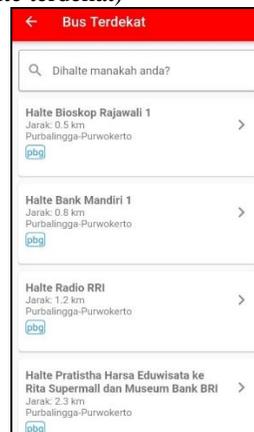
a) Tampilan menu utama



Gambar 7. Tampilan menu utama

Pada menu utama (Gambar 7), ditampilkan seluruh menu yang terdapat di aplikasi *monitoring* bus trans jateng. Dimana didalamnya terdiri dari menu Bus Terdekat, dan Halte Tedekat.

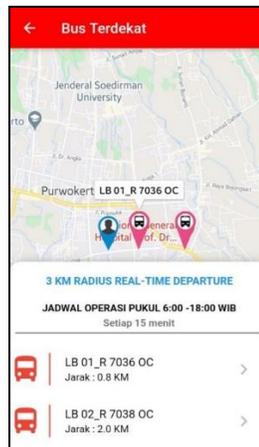
b) Tampilan menu bus terdekat (daftar halte terdekat)



Gambar 8. Tampilan bus terdekat (daftar halte terdekat)

Gambar 8 merupakan tampilan awal dari menu bus terdekat, dihalaman tersebut disediakan daftar halte terdekat dengan radius 3km dari lokasi pengguna dengan menggunakan rumus *haversine*. Ketika salah satu halte ditekan maka akan diteruskan ke halaman selanjutnya dibawah ini:

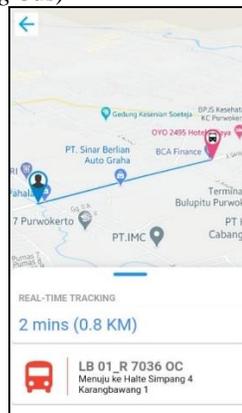
c) Tampilan menu bus terdekat (daftar bus terdekat)



Gambar 9. Tampilan bus terdekat (daftar bus terdekat)

Gambar 9 merupakan halaman tersebut ditampilkan marker halte dan marker bus di peta. Selanjutnya terdapat daftar bus yang jaraknya radius 3km dari halte dengan menggunakan rumus *haversine*. Ketika salah satu bus di pilih maka akan diteruskan kehalaman dibawah ini:

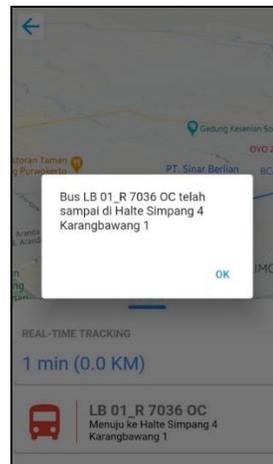
d) Tampilan menu bus terdekat (*monitoring bus*)



Gambar 10. Tampilan bus terdekat (*monitoring bus*)

Gambar 10 merupakan tampilan menu bus terdekat di halaman *monitoring*, dimana pada tampilan tersebut berisi peta yang terdiri dari dua marker yaitu pengguna di halte dan marker bus. Kemudian terdapat informasi mengenai jarak dan waktu yang diperlukan untuk menempuh antara bus dengan halte, dan informasi bus akan tiba di halte mana dengan menggunakan distance matrix API dari *google map*.

e) Tampilan menu bus terdekat (notifikasi bus telah sampai)



Gambar 11. Tampilan bus terdekat (notifikasi bus telah sampai)

Gambar 11 merupakan tampilan ketika bus sudah tiba di halte dimana pengguna berada, maka akan ditampilkan dialog atau popup bahwa bus yang dipilih telah sampai di halte tersebut. Ketika menekan OK maka akan kembali ke halaman sebelumnya.

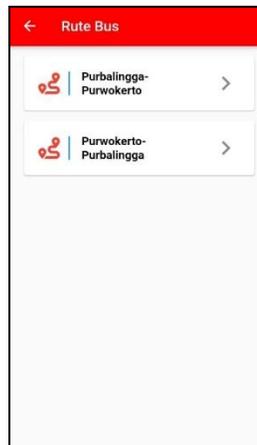
f) Tampilan menu halte terdekat



Gambar 12. Tampilan halte terdekat

Gambar 12 merupakan tampilan dari halaman halte terdekat dimana aplikasi akan menampilkan marker seluruh halte dan marker pengguna. Setelah itu pengguna memilih salah satu halte dan aplikasi akan menampilkan informasi nama, tipe dan rute dari halte yang dipilih tersebut.

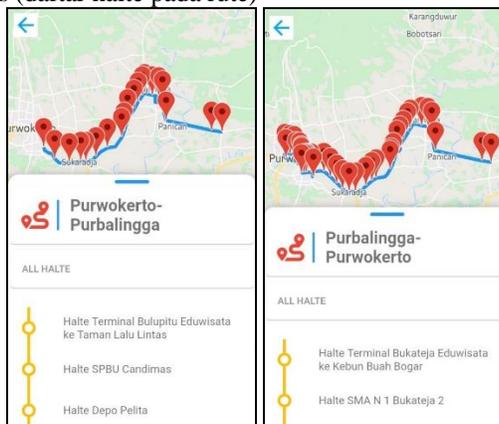
g) Tampilan menu rute bus (daftar rute)



Gambar 13. Tampilan rute bus (daftar rute bus)

Gambar 13 merupakan tampilan dari menu rute bus terdiri dari dua rute untuk koridor 1 Purwokerto-Purbalingga, yang apabila dipilih salah satu rute, maka akan diteruskan ke halaman selanjutnya dibawah ini:

h) Tampilan menu rute bus (daftar halte pada rute)



Gambar 14. Tampilan rute bus (daftar halte pada rute)

Gambar 14 merupakan ditampilkan daftar rute halte yang dilalui bus Trans Jateng. Didalamnya berupa tampilan peta beserta marker dan polyline, serta list dari nama-nama halte di rute tersebut.

IV. HASIL DAN PEMBAHASAN

4.1. Uji Keandalan Sistem

Pada tahap pengujian sistem kepada pengguna iterasi satu dilakukan wawancara dengan pihak kantor BRT Trans Jateng dan masyarakat pengguna bus trans jateng. Dimana pada iterasi pertama prosesnya adalah wawancara dari mockup yang telah dibuat pada proses perancangan sistem. Kemudian peneliti melakukan pembangunan sistem dari mockup kedalam tahap pengkodean. Tahap pengujian selanjutnya yaitu dilakukan iterasi kedua dengan menampilkan demo aplikasi yang fungsionalitasnya sudah menjawab tujuan dari batasan masalah, serta menjawab permintaan dari iterasi ke satu. Pada tahap ini yaitu mendengarkan *feedback* untuk literasi dua yang kemudian dilakukan pembangunan sistem lanjutan agar sesuai dengan masukan dari iterasi ke dua, setelah itu dilakukan pengujian untuk iterasi selanjutnya. Tahap pengujian terakhir yaitu tahap pengujian iterasi tiga. Tahap pengujian ini merupakan tahap pengujian final iterasi tiga yaitu ketika dinyatakan fungsional sistem sudah menjawab kebutuhan pengguna. Pada tahap pengujian ini, diperoleh *feedback* dari pengguna yang menyatakan bahwa aplikasi sudah dapat berjalan sesuai fungsionalitas, maka tahap iterasi tiga dihentikan dan sudah menjawab fungsional pengguna.

4.2 Uji Keakuratan Sistem

Pengujian keakuratan dilakukan dengan membandingkan antara waktu kedatangan bus dalam kondisi nyata dengan estimasi waktu kedatangan bus dalam aplikasi yang peneliti bangun yaitu aplikasi *monitoring* bus Trans Jateng. Pengujian dilakukan pada sampel halte yang dilakukan sebanyak 10 kali.

Untuk waktu nyata yang ditempuh oleh bus untuk menuju halte tujuan diperoleh dengan cara peneliti menghitung waktu dari halte asal ke halte tujuan dengan menggunakan stopwatch. Kemudian secara bersamaan ketika bus telah sampai di halte, peneliti memperhatikan aplikasi dan menghitung waktu maju untuk aplikasi *monitoring* bus Trans Jateng memunculkan notifikasi bus telah sampai di halte tujuan menggunakan stopwatch. Dengan demikian akan diperoleh waktu nyata bus dari halte asal ke halte tujuan, dan waktu yang diperlukan aplikasi *monitoring* bus Trans Jateng untuk memunculkan notifikasi bus telah sampai di halte. Adapun hasil pengujian disajikan dalam tabel dibawah ini:

TABEL 2 Hasil Uji Keakuratan Sistem

No	Halte	Waktu Nyata (mm:ss)	Aplikasi <i>Monitoring</i> bus trans jateng (mm:ss)	Perbandingan (mm:ss)
Rute Purwokerto-Purbalingga				
1	Terminal Bulupito – SPBU Candimas	04:10	04:26	00:16
2	Depo Pelita – SMPN Sokaraja	02:53	03:00	00:07
3	Simpang 4 Klentheng-SMAN 1 Sokaraja	02:50	03:00	00:10
4	Terminal Jompo-Fak Teknik Unsoed	00:51	01:01	00:10
5	SMK YPT 1-Terminal Purbalingga	03:03	03:08	00:05
Rata-Rata				00:10
Rute Purbalingga - Purwokerto				
6	Terminal Bukateja – SMAN 2 Bukateja 2	03:31	03:38	00:07
7	SMAN Bukateja 3 – Simpang 3 Bojong	06:20	06:35	00:15
8	Simpang 3 Bojong – MAN Purbalingga	04:49	04:60	00:11
9	Taman Gringsing – Simpang 3 Yuro	03:00	03:06	00:06
10	Terminal Purbalingga – SMK YPT 1	03:50	03:55	00:05
Rata-Rata				00:09

Berdasarkan pada tabel 2 didapatkan perbandingan waktu terlama aplikasi *monitoring* bus Trans Jateng memunculkan notifikasi setelah bus Trans Jateng tiba di halte yaitu 16 detik, dan didapatkan perbandingan waktu tercepat aplikasi *monitoring* bus Trans Jateng memunculkan notifikasi setelah bus Trans Jateng tiba di halte yaitu 5 detik, serta rata-rata dari setiap rute diantaranya rute Purwokerto-Purbalingga diperoleh sebesar 10 detik, dan rata-rata pada rute Purbalingga-Purwokerto sebesar 9 detik.

Berdasarkan hasil rekap perbandingan waktu pada rute diperoleh perbandingan waktu yang berbeda-beda tergantung pada banyak faktor, antara lain dikarenakan estimasi waktu yang ditampilkan pada aplikasi *monitoring* bus Trans Jateng merupakan pembulatan waktu dalam menit. Kemudian, faktor pengiriman data dari GPS yang terbuat dari *Arduino uno* dan Modul SIM808 dikirim setiap 12 detik sekali membuat akurasi ketepatan waktu tidak sebanding dengan kecepatan jalannya bus untuk tiba di setiap halte.

V. Kesimpulan

Berdasarkan hasil pengujian pada pengembangan aplikasi *monitoring* bus Trans Jateng, dapat disimpulkan sebagai berikut:

1. GPS *tracking* yang dibangun dengan mikrokontroler *Arduino uno* dan Modul SIM808 berdasarkan rangkaian dan alur sistem GPS *tracking* dapat memperoleh data koordinat bus Trans Jateng secara *realtime*.
2. Aplikasi *monitoring* bus Trans Jateng dapat menghitung jarak dan waktu kedatangan bus Trans Jateng berdasarkan data koordinat GPS yang didapat menggunakan *Google map* API untuk menghitung jarak dan estimasi waktu yang diperlukan bus untuk tiba di halte pengguna.
3. GPS *tracking* yang dibangun mampu mendapat data koordinat latitude dan longitude bus secara *realtime* dengan memanfaatkan teknologi GPRS sehingga data koordinat bus dapat dikirim ke *server* yang kemudian diteruskan dan disimpan di *cloud database realtime Firebase*.
4. Aplikasi *monitoring* bus Trans Jateng mampu memberikan informasi daftar halte yang terdekat dengan radius 3km menggunakan rumus *Haversine* untuk memperoleh jarak halte terdekat berdasarkan lokasi pengguna saat ini.
5. Aplikasi *monitoring* bus Trans Jateng dapat diakses melalui *smartphone* android dengan metode prototyping dan dapat menjawab fungsionalitas pengguna berdasarkan uji *black box* yang telah dilakukan diperoleh 93,8% menyatakan aplikasi sudah berjalan sesuai dengan fungsinya. Kemudian aplikasi *monitoring* bus Trans Jateng juga dapat memunculkan notifikasi setelah bus sampai dihalte dengan waktu terlama yaitu 16 detik dan waktu tercepat yaitu 5 detik, serta rata-rata dari setiap rute diantaranya rute Purwokerto-Purbalingga diperoleh sebesar 10 detik, dan rata-rata pada rute Purbalingga-Purwokerto sebesar 9 detik.

Aplikasi *monitoring* bus Trans Jateng perlu dikembangkan lagi agar dapat digunakan di platform iOS. Serta akan lebih baik jika dapat menghitung waktu kedatangan dengan pembulatan waktu dalam menit dan detik.

DAFTAR PUSTAKA

- [1] H. Jateng, "Groundbreaking, Bandara Jendral Soedirman Diharapkan Dongkrak Investasi Jateng.," 2018. .
- [2] W. Nugraha and M. Syarif, "Penerapan Metode Prototype Dalam Perancangan Sistem Informasi Penghitungan Volume Dan Cost Penjualan Minuman Berbasis Website.," *JUSIM (Jurnal Sist. Inf. Musirawas)*, vol. 3, no. 2, pp. 94–101, 2018, doi: 10.32767/jusim.v3i2.331.
- [3] Y. D. P. Junaidi, *Project Sistem Kendali Elektronik Berbasis Arduino*. Bandar Lampung: AURA, 2018.
- [4] S. Alfeno, R. Eka, C. Devi, D. S. Raharja, and A. S. Raharja, "Implementasi Global Positioning System (GPS) dan Location Based Service (LSB) pada Sistem Informasi Kereta Api untuk Wilayah Jabodetabek.," vol. 7, no. 2, 2017.
- [5] M. D. Falih, D. N. Ramadan, and U. Telkom, "Implementasi Layanan Gprs Dengan Osmocom Bts di Fakultas Ilmu Terapan.," vol. 6, no. 1, pp. 513–521, 2020.
- [6] R. R. Irawan, "Prototipe Pemberitahuan Lokasi Koordinat Darurat Menggunakan GPS dan Pulse Sensor Berbasis Arduino dan SMS.," 2018.
- [7] B. Maftukhah, "Analisis Dan Implementasi Rumus Haversine Dalam Penentuan Arah Kiblat Di Indonesia Menggunakan Trigonometri Sphiral.," Universitas Islam Negeri Sunan Ampel Surabaya, 2018.
- [8] F. Aming, "Aplikasi Toursharing Menggunakan Algoritma Haversine Di Wilayah Toraja.," Universitas Islam Negeri Alauddin Makassar, 2019.