RANCANG BANGUN PELACAK BUS UNTUK *SMART CITY*

Chintya Nermelita Mandalahi[1], Simon Siregar[2], Duddy Soegiarto[3]

[1]Universitas Telkom, [2]Universitas Telkom, [3]Universitas Telkom
[1]chintyanm@gmail.com, [2]simon.siregar@tass.telkomuniversity.ac.id,
[3]duddy@tass.telkomunivesity.ac.id

Abstrak

Abstrak – Sistem pelacak kendaraan berbasis GPS menggunakan komunikasi GPRS yang merupakan sebuah sistem dimana posisi suatu kendaraan dapat diketahui secara pasti. Sistem pelacak menggunakan GPS (*Global Positioning System*) untuk menentukan posisi tersebut serta GPRS (*General Packet Radio Service*) sebagai jembatan pengiriman data dari mikrokontroler ke server. Modul GPS dan GPRS menggunakan SIMCOM SIM900. Sistem GPS menggunakan *NMEA-0183 protocol* untuk berkomunikasi dengan mikrokontroler ATmega 328-P. Mikrokontroler digunakan sebagai penghubung sistem GPS (menggunakan *NMEA-0183 Protocol*) dan GPRS melalui serial TTL (*UART*). Mikrokontroler bertugas mengirimkan data GPS untuk kemudian dikirimkan kepada server melalui komunikasi GPRS menggunakan protokol *AT+Command* untuk memerintahkan GSM mengirimkan data kedalam server. Pengiriman data tersebut menggunakan protokol HTTP dengan metode *$GET*. Koordinat lokasi kendaraan yang tersimpan pada server akan ditampilkan pada halaman web yang telah dilengkapi denga peta (menggunakan *google maps*) dan didalamnya disertai *marker* posisi kendaraan. Halaman web tersebut akan ditampilkan oleh *mini computer* (RADXA *Board*) dan akan muncul pada layar monitor. Terdapat tabel informasi memungkinkan untuk pencarian lokasi dan jam.

Kata kunci: SIMCOM SIM900, GPS, GPRS, Icomsat, *Google Maps, Google APIs*

Abstract

(6) Abstract – ***GPS-based vehicle tracking system using GPRS communication which is a system where the position of a vehicle can be known with certainty. Tracking system using GPS (Global Positioning System) to determine the position of the vehicle as well as GPRS (General Packet Radio Service) as a bridge sending data from the microcontroller to the server. GPS module and GPRS using SIM900 Simcom of DFrobot. GPS systems using NMEA-0183 protocol to communicate with the microcontroller ATmega 328-P. Microcontroller is used as a connector GPS system (using the NMEA-0183 protocol) and GPRS via serial TTL (UART). Microcontroller in charge of sending the GPS data to then be sent to the server via GPRS communication protocol using the AT + Command to instruct the GSM transmit data to server. The data transmission using HTTP protocol with $ GET method. Vehicle location coordinates stored on server will be displayed on web pages that have been equipped premises map (using google maps) and under accompanied marker position of the vehicle. Page the web will be displayed by a mini computer (RADXA Board) and will appear on the monitor screen. There is a table of information makes it possible to search the location and hours.***

Keywords: SIMCOM SIM900, GPS, GPRS, Icomsat, *Google Maps, Google APIs*

1. Pendahuluan

Perkembangan sarana tranportasi di dunia mambawa pengaruh yang sangat besar pada kehidupan masyarakat Indonesia terutama banyaknya masyarakat lebih memilih tranportasi tersebut yaitu Bus. Bus merupakan sarana transportasi yang kini menjadi favorit masyarakat umum. Orang-orang lebih memilih sarana transportasi bus karena lebih praktis dan ekonomis. Karena hal tersebut, hatle bus atau sering disebut terminal bus kini menjadi ramai dan penuh dengan orang yang berdesak-desakan untuk menggunakan transportasi ini. Salah satu akibat antrian tersebut karena masyarakat tidak tahu secara pasti kapan kedatangan bus, sehingga orang-orang harus datang lebih dahulu untuk menunggu busway yang akan datang. Keadaan tersebut dirasa kurang efektif dan dari sistem transportasi bus belum memberikan sistem yang informatif kepada masyarakat.

Berdasarkan permasalahan tersebut maka harus diterapkan sistem yang lebih informatif yaitu sistem *Smart City*. Salah satu manfaat sistem ini akan memudahkan masyarakat untuk mengetahui keadaan-keadaan suatu tempat. Dalam kasus ini menggunakan *Bus Tracking* sebagai pengimplementasikan dari *Smart City* dalam bidang Transportasi. Dengan *Bus tracking* yang menggunakan GPS, calon penumpang bus akan dimudahkan dengan layar monitor yang ada didalam terminal atau halte bus yang akan menampilkan posisi semua armada bus dalam peta kota dan terdapat waktu kedatangan bus ke koridor tersebut.

Sistem ini akan sangat memudahkan masyarakat yang ingin menggunakan sarana transportasi bus karena lebih informatif. Calon penumpang tidak perlu terburu-buru dan berdesak-desakan untuk masuk ke dalam bus karena takut bus berikutnya akan datang lama. Kedatangan bus berikutnya dapat diketahui secara langsung tanpa perlu menunggu.

2. Dasar Teori

2.1. *Single Board Computer* "Radxa *Board*"

Radxa adalah mini PC atau *single board computer* yang merek dagangnya terdaftar oleh Radxa Limited di Shenzhen, China. Hardware ini bentuknya kecil tetapi banyak sekali kegunaannya. Board ini memiliki spesifikasi yang setara dengan gadget masa kini. Terdapat 2 tipe Radxa yaitu Radxa Pro Lite dan Radxa Pro Rock. Radxa Pro Lite memiliki spesifikasi lebih kecil disbanding Radxa Pro Rock. Radxa Pro Rock memiliki spesifikasi yang terbilang cukup untuk kebutuhan riset. [1]

Gambar 1: Radxa *Board*

Adapun spesifikasi Radxa *Board* adalah sebagai berikut:

1. CPU = ARM Cortex-A9 quad core @ 1.6Ghz
2. 2GB DDR3 @800Mhz
3. 8GB Nand Flash, Micro-SD SDXC up to 128GB
4. HDMI 1.4 up to 1080p@60hz

5. GPU = Mali400-mp4@533Mhz, OpenGL ES 2.0
6. 10/100M Ethernet port
7. 150Mbps 802.11b/g/n with antenna
8. Bluetooth 4.0
9. 2 USB 2.0 standard A type
10. USB OTG micro-USB connector
11. Serial consol for debugging
12. 1 power key, 1 recovery key, 1 reset key
13. RTC = Support power off alarm
14. Extend header = 80 pins including GPIO, I2C, SPI, Line in, USB 2.0, PWM, ADC, LCD, etc

2.2. U-Blox Neo 6M

GPS (*Global Positioning System*) adalah sistem navigasi satelit untuk menentukan letak posisi di bumi. Sistem GPS memanfaatkan satelit yang berada di orbit bumi, satelit yang memancarkan sinyal ke bumi akan ditangkap dan diolah oleh modul GPS. Salah satu contoh modul GPS adalah U-Blox GPS NEO-6M. [2]

Gambar 2: Arduino Pro Mini

U-Blox GPS NEO-6M memiliki spesifikasi sebagai berikut:

1. Tipe penerima: 50 kanal, GPS L1 frequency, C/A Code. SBAS: WAAS, EGNOS, MSAS
2. Sensitivitas penjejak & navigasi: -161 dBm (reakuisisi dari blank-spot: -160 dBm)
3. Sensitivitas start: -147 dBm pada cold-start, -156 dBm pada hot start.
4. Kecepatan pembaharuan data / navigation update rate: 5 Hz
5. Akurasi penetapan lokasi GPS secara horisontal: 2,5 meter (SBAS = 2m)
6. Rentang frekuensi pulsa waktu dapat diseting antara 0,25 Hz hingga 1 kHz.
7. Akurasi sinyal pulsa waktu: RMS 30 ns (99% dalam kurang dari 60 ns) dengan granularitas 21 ns atau 15 ns saat terkompensasi.
8. Akurasi kecepatan: 0,1 meter / detik.
9. Akurasi arah (heading accuracy): 0,5°.

Batasan operasi: daya tarik maksimum 4x gravitasi, ketinggian maksimum 50 Km, kecepatan maksimum 500 meter / detik (1800 km/jam).

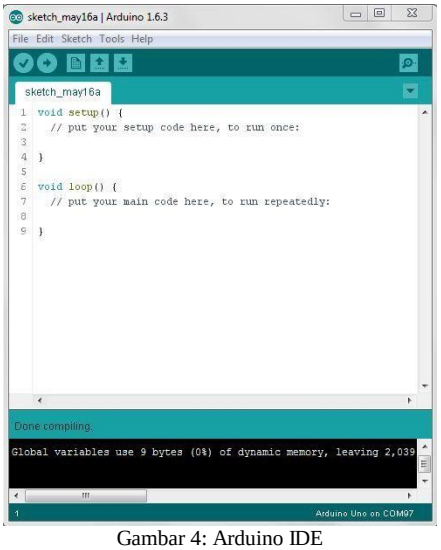
2.3. Arduino

Arduino Uno adalah board mikrokontroler berbasis ATmega328 (*datasheet*). Memiliki 14 pin *input* dari *output* digital dimana 6 pin *input* tersebut dapat digunakan sebagai *output* PWM dan 6 pin *input analog*, 16 MHz osilator kristal, koneksi USB, jack power, ICSP header, dan tombol reset. Untuk mendukung mikrokontroler agar dapat digunakan, cukup hanya menghubungkan b*oard* Arduino Uno ke komputer dengan menggunakan kabel USB atau listrik dengan AC yang-ke adaptor-DC atau baterai untuk menjalankannya.

Gambar 3: Arduino UNO

Uno berbeda dengan semua board sebelumnya dalam hal koneksi USB-to-serial yaitu menggunakan fitur Atmega8U2 yang diprogram sebagai konverter USB-to-serial berbeda dengan board sebelumnya yang menggunakan chip FTDI driver USB-to-serial.

Selain *hardware*, Arduino juga memiliki software yaitu Arduino IDE (*Integrated Development Environment*) yang merupakan *software* pemrograman mikrokontroler Arduino. Dalam aplikasi Arduino IDE telah terdapat *compiler* berbasis GCC. Arduino IDE menggunakan bahasa pemrograman C dan memiliki *library* yang bersifat *open source* sehingga setiap orang dapat membuat *library* sendiri sesuai dengan keperluan.



Gambar 4: Arduino IDE

2.4. IComsat SIM900

IComsat adalah GSM/GPRS shield untuk Arduino dan berbasis SIM900 *quad-band* module. Dalam hal ini dikontrol via *AT Commands* (GSM 07.07, 07.05 dan *SIMCOM AT Commands*), dan sangat *compatible* dengan arduino baik Arduino UNO, arduino MEGA dan lainnya. [4]

Gambar 5: IComsat SIM900

Fitur dari IComsat module ini yaitu:

1. *Quad-Band* 850/900/1800/1900MHz
2. GPRS *multi-slot calss* 10/8
3. GPRS *mobile station class* B
4. *Compliant to* GSM *phase* 2/2+
5. *Class* 4 (2W@850/900MHz)
6. *Class* 1 (1W@1800/1900MHz)
7. *Control via commands* (GSM 07.07, 07.05 and *SIMCOM enhanced AT Commands*)
8. *Short message service*
9. *Free serial port selection*
10. *All* SIM900 *pins breakout*
11. RTC *supported with Super Cap*
12. *Power on/off and reset function supported by* Arduino *interface*

2.5. NMEA *Protocol*

NMEA (*National Marine Electronics Association*) *protocol* adalah standar protokol data dalam sistem navigasi. NMEA *Protocol* digunakan pada sistem GPS dalam pengiriman data dari satelit. NMEA memiliki beberapa *header* paket data dengan informasi yang berbeda.

Tabel 1: *Header* Paket Data NMEA

Header	Deskripsi
GGA	*Time, position and fix type data.*
GLL	*Latitude, longitude, UTC time of position fix and status.*
GSA	*GPS receiver operating mode,* satelits *used in the position solution, and DOP values.*
GSV	*The number of GPS* satelits *in view* satelit *ID numbers, elevation, azimuth, and SNR values.*
MSS	*Signal-to-noise ratio, signal strength, frequency, and bit rate from a radio-beacon receiver.*
RMC	*Time, date, position, course and speed data.*
VTG	*Course and speed information relative to the ground.*
ZDA	*PPS timing message (synchronized to PPS).*
150	*OK to send message.*[5]

2.6. Google Maps

Google Maps adalah dasar pemetaan web dan teknologi aplikasi layanan yang disediakan oleh Google, gratis (untuk non-komersial). Di dalam *Google Maps* menawarkan peta jalan, sebuah rute rencana untuk bepergian dengan berjalan kaki, mobil, atau angkutan umum dan pemantau bisnis di perkotaan untuk beberapa negara di sekitar dunia. Menurut salah satu pencipta (Las Rasmussen), *Google Maps* adalah suatu cara untuk mengorganisasikan informasi di dunia secara geografis. Seperti banyak aplikasi web Google lainya, *Google Maps* menggunakan *JavaScript* secara ekstensif. *Google maps* menyediakan "*API key*" sebagai sarana untuk dapat menampilkan peta Google maps pada halaman web yang telah dibuat. *API key* tersebut disisipkan pada program halaman utama. [6]

2.7. Parsing Data

Parsing data atau dikenal dengan penguraian data adalah suatu metode yang digunakan untuk membaca paket data dari suatu protokol. Dalam menguraikan suatu paket data terdapat tiga komponen penting dalam susunan paket data tersebut yaitu:

1. *Header*
   Sebagai perintah atau indikator alamat dari data yang dibawakan.
2. *Data*
   Merupakan nilai nilai yang memiliki informasi yang nanntinya dapat diolah dan menghasilkan informasi yang mudah dimengerti.
3. *Checksum*
   Merupakan penutup dari suatu paket data yang akan mengindikasikan kelengkapan data.

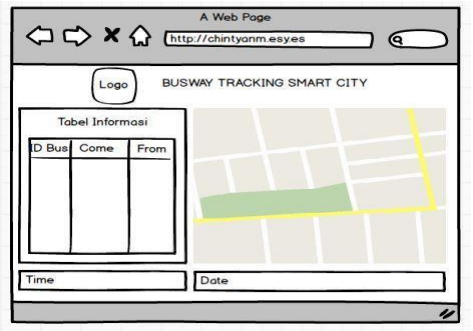
3. Analisis dan Perancangan

3.1. Gambaran Sistem Saat Ini

Bus merupakan sarana transportasi yang menjadi favorit masyarakat umum. Masyarakat lebih memilih sarana transportasi bus karena lebih praktis dan ekonomis. Karena hal tersebut kini hatle bus menjadi ramai dan penuh dengan orang yang berdesak-desakan untuk menggunakan transportasi ini. Salah satu akibat antrian tersebut karena masyarakat tidak tahu secera pasti kapan kedatangan bus sehingga mereka harus datang lebih dahulu untuk menunggu bus yang akan datang. Keadaan tersebut dirasa kurang efektif dan dari sistem transportasi bus belum memberikan sistem yang informatif kepada masyarakat. Saat ini perusaahan tranportasi sudah membuat sistem yang informastif agar masyarakat tau pasti kedatangan bus akan datang. Namun, hal tersebut kurang efisien, karena informasi yang didapat kurang dipahami olah kalangan awam. Maka dari itu, *bus tracking for smart city* bisa menjawab semua keluhan masyarakat. Sistem tersebut sangat membantu masyarakat dalam mengetahui kedatangan serta keberangkatan bus. Dilengkapi *maps*, calon penumpang bisa secara langsung melihat pergerakan bus.

3.2. Perancangan Desain Aplikasi

Perancangan desai ini terdapat pada pembuatan halaman web. Pada halamna web tersebut berisi peta, tabel informasi, jam digital. Hal ini dapat dilihat melalui monitor atau melalui *smartphone*. Berikut desain perancangan halaman web:

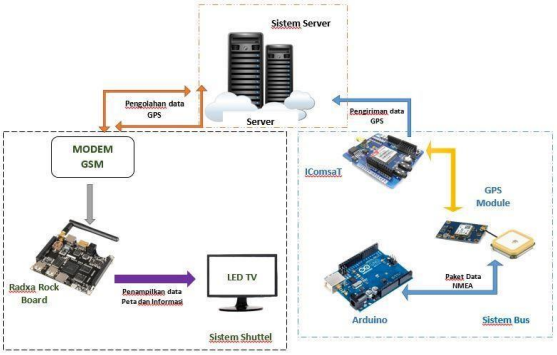


Gambar 6: Desain tampilan web pada *desktop*

Gambar 7: Desain tampilan web pada *smartphone*

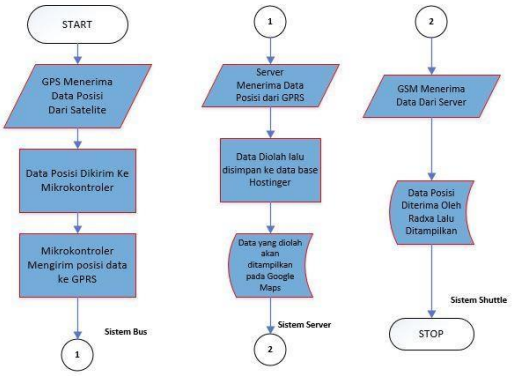
3.3. Perancangan Sistem

Dalam melakukan pengiriman informasi GPS dari GPRS kepada *Server* dan ditampilkan pada *shuttle* maka diperlukan adanya perancangan sistem yang dapat dilihat pada Gambar 8.

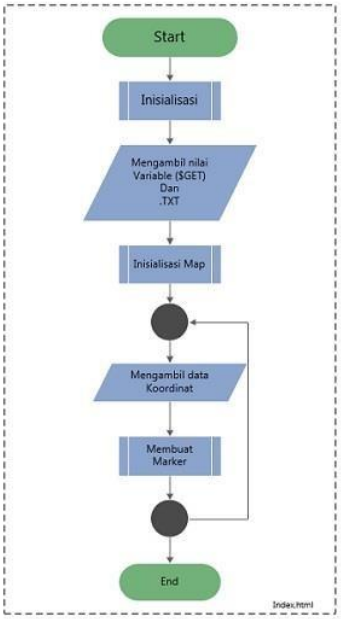


Gambar 8: Blok Diagram Sistem

Selain perancangan yang ditunjukkan pada gambar 8 juga terdapat alur algoritma pemrograman pada sistem pelacak bus serta alur pemograman pada HTML yang dapat dilihat pada Gambar 9 dan Gambar 10.



Gambar 9: *Flowchart* Pemrograman sistem keseluruhan



Gambar 10: *Flowvhart* index.html

Pada *flowchart* tersebut, Inisialisasi dilakukan pada awal program dengan tujuan mendeklarasikan fungsi-fungsi *javascript* serta variable-variabel yang nantinya dijadikan tempat penyimpanan koordinat *device*. Kemudian program akan menyimpan data GPS pada "*$filename= .txt*". Program akan menginisialisasi *maps* dengan memanggil halaman yang telah disediakan oleh *Google* kemudian ditampilkan. Setelah data koordinat diterima, maka program akan menampilkan *marker* sesuai koordinat yang diterima oleh "*$filename= .txt*".

3.4. Kebutuhan Perangkat Keras dan Perangkat Lunak

Adapun kebutuhan perangkat keras dan perangkat lunak yang digunakan dalam pembuatan sistem pelacak bus adalah sebagai berikut:

1. Kebutuhan perangkat keras:

Tabel 2: Kebutuhan Perangkat Keras

Hardware	Kegunaan
Arduino UNO	Sebagai mikrokontroler yang berfungsi mengolah data dari GPS
IComasat	Sebagai komunikasi pengiriman data GPS melalui GPRS.
U-Blox NEO 6-M GPS *Module*	Sebagai modul GPS yang berfungsi sebagai sistem penerima sinyal dari satelit yang mengelilingi bumi dan memperhitungkan posisi perangkat sesuai data yang diterima dari satelit.
HDMI Cable	Sebagai penghubung single board computer dengan layar monitor.
***Single Board Computer* (RADXA)**	Mini computer yang berfungsi menampilkan data posisi suatu kendaraan.
Monitor	Dalam hal ini, berfungsi hanya menampilkan web serta posisi bus berada sesuai dengan jalur yang ditentukan
Adaptor	Hal ini berfungsi sebagai penambah daya pada mikrokontroler dan *single board computer*

2. Kebutuhan perangkat lunak:

Tabel 3: Kebutuhan Perangkat Lunak

Software	Kegunaan
Arduino IDE	Sebagai aplikasi yang digunakan untuk penulisan kode program untuk mikrokontroler.
Notepad	Sebagai aplikasi yang digunakan dalam memprogram HTTP, PHP serta CSS.

4. Implementasi dan Pengujian

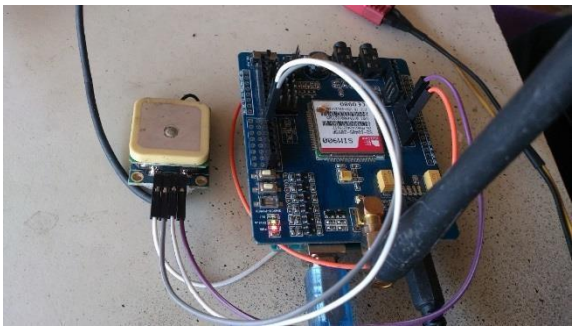
4.1. Implementasi

Implementasi pelacak bus tersebut terintegrasi pada tiga sistem yang dapat terhubung dan saling mengirim informasi sesuai dengan alur yang telah dilakukan. Dalam sub Bab ini akan dibahas langkah-langkah pembangunan sistem pelacak bus.

A. Pemasangan Komponen dan Modul

Pada pemasangan komponen maupun modul ditetapkan 2 tempat yaitu pada bagian Bus dan bagian terminal.

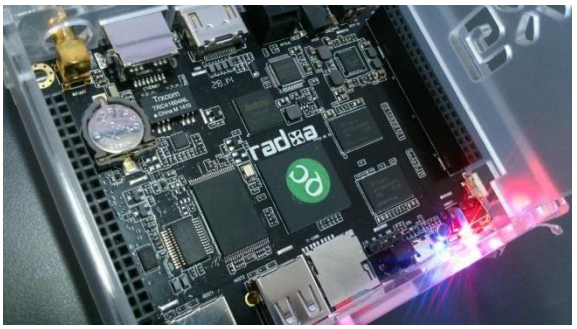
Pada bagian bus terdapat 3 module yaitu Arduino, Icomsat dan GPS (Ublox). Ketiga module tersebut terpasang secara bertingkat dengan menancapkan IComsat pada Arduino serta pada bagian IComsat dipasang GPS.



Gambar 11: Susunan modul sistem bus

Pada bagian terminal terdapat 3 alat yaitu *single board computer* (RADXA board), Layar monitor serta GSM modem. Pada bagian ini *single board computer* menampilkan peta serta

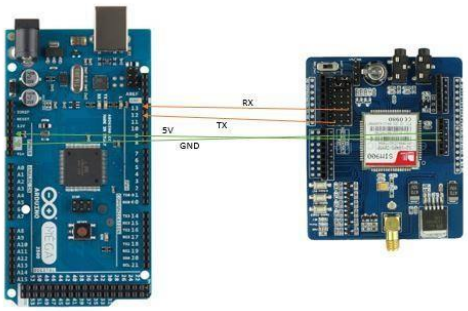
tabel informasi yang secara *default* terdiri dari id_halteu, keberangkatan serta kedatangan bus. GSM modem menerima data dari server lalu akan diterima oleh *single board computer* dan ditampilkan pada layar monitor.



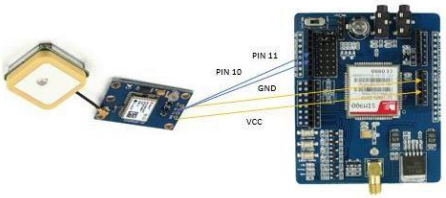
Gambar 12: Pengujian Radxa *Board*

B. Wiring Diagram

Untuk menghubungkan komponen-komponen dan modul-modul yang digunakan maka harus mengetahui pin mana saja yang akan digunakan. Berikut adalah *wiring diagram* sistem pada GPS:



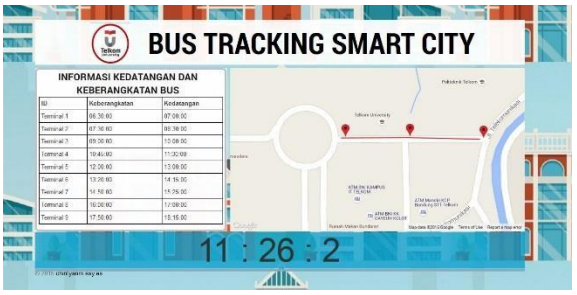
Gambar 13: Gabungan modul Arduino dan IComsat



Gambar 14: Gabungan modul IComsat dan GPS

C. Pemograman Halaman Web

Pada Gambar 15, dalam pembuatan halaman web terdapat fasilitas berupa tabel informasi serta peta penunjuk bus yang ditandai dengan pergerakan *marker*. Adapun desain web yang telah dibuat adalah sebagai berikut:



Gambar 15: Desain Halaman Web Pada *Desktop*

4.2. Pengujian

A. Pengujian Pembacaan Data GPS Pada *Serial Monitor*

Pengujian pembacaan data GPS melalui PC dengan cara menghubungkan modul GPS dengan USB TTL yang telah terhubung dengan PC. Setelah itu dilakukan pembacaan data GPS dengan *Serial Monitor* yang terdapat pada aplikasi *Arduino IDE* Berikut Tabel 4 berupa tingkat keberhasilan kesesuaian data dalam pengambilan data *NMEA* pada *Serial Monitor*.

Tabel 4: Hasil Pengujian Pembacaan Data GPS

Data NMEA	Kesesuaian Data
$GPGGA	Data sesuai
$GPGLL	Data sesuai
$GPGSA	Data sesuai
$GPGSV	Data sesuai
$GPMSS	Data sesuai
$GPRMC	Data sesuai
$GPVTG	Data sesuai
$GPZDA	Data sesuai

Pada Tabel 4, pengujian pembacaan data GPS berhasil dilakukan dengan menggunakan *Serial Monitor*. Pengujian ini dilakukan untuk memastikan bahwa modul GPS dapat mengirim paket data dan terkoneksi dengan satelit.

B. Pengujian Akurasi dan Sensitifitas Sistem GPS

Pengujian GPS ini dilakukan dengan memberikan sumber tegangan pada sistem GPS dan mencatat hasil pengujian. Berikut Tabel 5 yang menunjukan sistem pengujian pada GPS:

Tabel 5: Pengujian Akurasi & Sensitifitas GPS

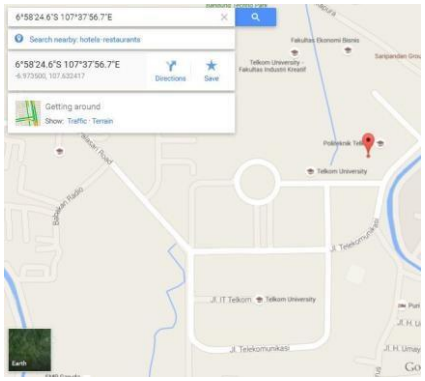
No	Latitude	Longitude	Jumlah Satelite	Valid = A Tidak Valid= V	Speed
1	-6. 9780373	107. 6377105	10	A	0.400
2	-6. 9780049	107. 6376495	10	A	0.060
3	-6. 9780049	107. 6376495	10	A	0.010
4	-6. 9780049	107. 6376495	10	A	0.010
5	-6. 9780049	107. 6376495	10	A	0.090
6	-6. 9780049	107. 6376495	10	A	0.090
7	-6. 9780049	107. 6376495	10	A	0.006
8	-6. 9780049	107. 6376495	10	A	0.010
9	-6. 9780049	107. 6376495	10	A	0.030
10	-6. 9780049	107. 6376495	10	A	0.450

Berdasarkan pengujian diatas dapat disimpulkan bahwa GPS sangat besar pengaruhnya. Kondisi lingkungan dan kondisi alam mempengaruhi dalam pengambilan data GPS sehingga pada pengujian terdapat ketidaksesuaian data sekitar ±0.010 meter. Pengujian di dalam ruangan dan di luar ruangan memberikan hasil yang berbeda-beda. Penggunaan GPS di luar ruangan (*outdoor*) akan mendapat sinyal lebih banyak dibandingkan dengan di dalam ruangan. Penggunaan GPS pada cuaca yang cerah akan mendapatkan sinyal yang maksimal dibandingkan dengan cuaca saat berawan atau mendung. Pada tabel 5 di atas terdapat *satellite* dengan angka yang tetap, dikarenakan adanya kondisi tertentu yang memungkinkan kedudukan *satellite* sama.

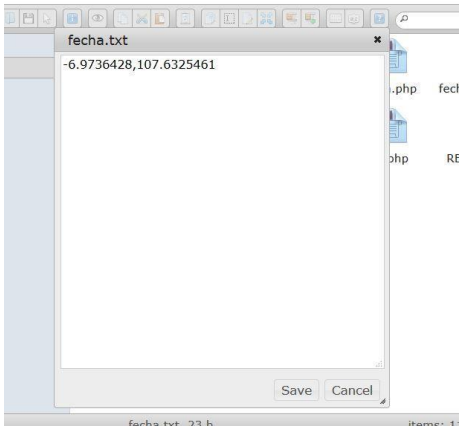
C. Pengujian Data Posisi GPS

Pada pengujian ini akan dicari posisi koordinat tersebut pada peta dengan memasukkan angka-angka koordinat dari data GPS yang telah dikirim pada file *.txt* menggunakan *google maps*. Hal ini

untuk memastikan apakah posisi yang telah di kirim pada file *.txt* tersebut apakah benar atau tidak dapat diketahui pada gambar 16.



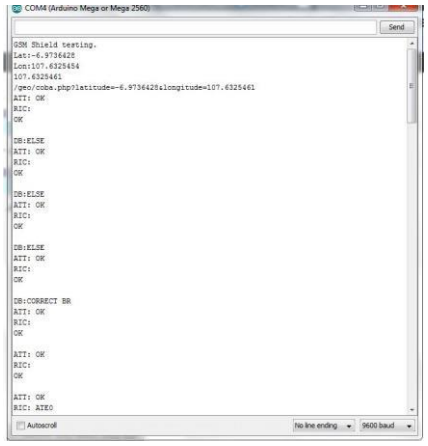
Gambar 16: Percobaan posisi GPS pada *Google Maps*



Gambar 17: Titik koordinat GPS yang di simpan pada file.txt

D. Pengujian Pengiriman Data Melalui GPRS

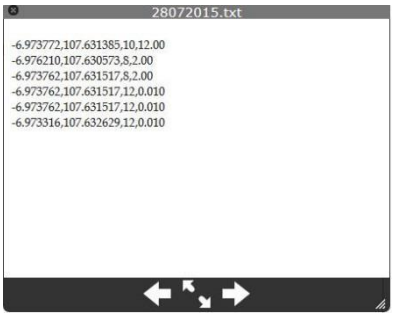
Pada Gambar 18, data yang dihasilkan dari GPS akan masuk kedalam "*filename= .txt*" dan ditampilakan pada web. Pengiriman melalui GPRS dilakukan dengan pemograman Arduino IDE dimana terdapat IComsat *shield* yang langsung terintegrasi dengan Arduino.



Gambar 18: Proses pengiriman data melalui komunikasi GPRS

Data yang telah dikirim oleh GPRS akan otomatis masuk ke dalam sebuah file berupa *.txt* sesuai dengan tanggal, bulan serta tahun. Hal ini memudahkan user dalam mencari data jika ada kemungkinan hal terjadi pada bus sebelumnya, maka user hanya membuka data sesuai dengan tanggal kejadian.

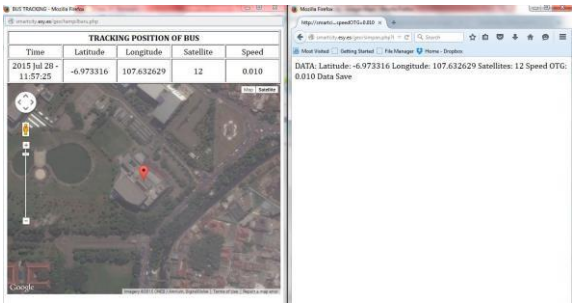
Gambar 19: Penyimpanan data GPS pada *.txt*



Gambar 20: Isi penyimpana data di *.txt*

Setelah data masuk, maka data tersebut akan kuncul pada web dan terisi secara otomatis, sehingga akan muncul suatu tanda yang menunjukan letak atau posisi bus pada peta. Untuk membuktikan data masuk atau tidak dengan memasukan data secara manual dengan menyipan data di dalam PHP sesuai dengan Gambar 19 dan Gambar 20.

Gambar 21: Posisi Marker



Gambar 22: Pengiriman data secara manual

Pada Gambar 21, terdapat 5 bagian di dalamnya yaitu *Time* sebagai estimasi waktu keberangkatan bus, *Latitude* dan *Longitude* sebagai koordinat bus, *Satellite* dan *Speed* sebagai penanda kecepatan bus pada peta. Pada gambar 4.15, dilakukan pengujian secara berturut agar mendapatkan hasil sesuai dengan posisi tiap terminal. Pada Gambar 22, dilakukan penginputan data manual dengan memasukan ke lima bagia tersebut dalam perintah *"smartcity.esy.es/geo/simpan.php?latitude=12345&longitude=54321&satellites=12&speed=0.001"*, data tersebut akan tersimpan langsung ke *"$filename=.txt"*.

5. Kesimpulan

Dari data pengujian tersebut dapat diambil kesimpulan sebagai berikut:

1. Sistem pelacak bus dapat di simulasikan melalui integrasi dua sistem yaitu sistem pada software dan sistem pada hardware.

2. Dengan adanya penentuan posisi, etimasi waktu, serta kecepatan membuat sistem pelacak bus semakin lebih informatif.

3. Sistem GPS sangat rentan terpengaruh oleh faktor cuaca. Saat cuaca cerah, GPS akan cepat mendapatkan data *latitude* serta *longitude* dengan estimasi waktu kurang dari 1 menit sedangkan saat cuaca berawan atau mendung maka sistem GPS akan sulit terkoneksi untuk mendapatkan data serta sulit juga untuk mendapatkan data *satellite*. Jika berada di luar ruangan, *satellite* dapat terkoneksi mencapai batas 12 *satellite* sedangkan di dalam ruangan GPS terkoneksi dengan *satellite* mencapai jangka waktu yang lama kurang dari 10 satelit. Selain itu, keadaan lingkungan juga berpengaruh.

4. Penggunaan teknologi GPRS sebagai media komunikasi dalam sistem ini memiliki keterbatasan yaitu tergantung kepada operator yang dipakai. Jika jaringan operator sedang tidak sibuk maka pengiriman data akan cepat, tetapi jika operator sedang sibuk maka pengiriman SMS menjadi lebih lambat, bahkan bisa gagal sama sekali.

Daftar Pustaka

[1] RADXA. RADXA PRO ROCK: Overview, Summary [Online]. HYPERLINK "http://radxa.com/Rock".

[2] U-BLOX*: NEO 6 U-blox 6 GPS Modules Datasheet*. [ONLINE]. HYPERLINK "http://www.u-blox.de/images/downloads/Product_Docs/NEO-6_DataSheet (GPS.G6-HW-09005).pdf".

[3] ARDUINO. ARDUINO PRO MINI: Overview, Summary [Online]. HYPERLINK "http://arduino.cc/en/Main/ArduinoUno".

[4] WIKI. IComsat. [Online]. HYPERLINK "http://wiki.iteadstudio.com/IComSatJohn"

[5] Pujiono Nanda, *Sistem Operasi, Mengenal Perintah Dasar Linux Ubuntu*. STMIK AMIKOM, 2015.

[6] Kurniawan Aris Pujud. *Pengiriman Informasi GPS (GLOBAL POSITIONING SYSTEM) Berupa Teks Melalui Wirelles Pada AR DRONE 2.0* PROYEK AKHIR, 2015.

[7] Reza Jordan Syah. *Pengertian PHP dan My SQL*. [Online]. HYPERLINK "http://jordansyahreza.blogspot.com/p/pengertian-php-dan-my-sql.html"

[8] Supriadi. *Sejarah HTML*. [Online]. HYPERLINK "http://supriadi.blogs.unhas.ac.id/sejarah-html/"

[9] Azmi Nurul. *Pemanfaatan Google Api (Google Maps) Pada Website Pariwisata Menggunakan Framework Codeignter*, AMIKOM Yogyakarta, 2013.

[11] Pranjoto Hartanto. *Sistem Pelacak Dan Pengaman Kendaraan Berbasis GPS Dengan Menggunakan Komunikasi GPRS*. Universitas Widya, 2015.

[12] Google Developers. Google Maps APIs. [Online]. HYPERLINK "https://developers.**google**.com"