

PERANCANGAN DAN IMPLEMENTASI SISTEM INFORMASI GEOGRAFIS : SISTEM PEMANTAUAN BANJIR BERBASIS ANDROID

DESIGN AND IMPLEMENTATION GEOGRAPHIC INFORMATION SYSTEM : FLOOD MONITORING SYSTEM BASED ON ANDROID

Perdana Erick Oktafianto¹, Asep Mulyana S.T., M.T.², Umar Ali Ahmad S.T., M.T.³

^{1,2,3}Prodi S1 Sistem Komputer, Fakultas Teknik, Universitas Telkom

¹erickperdana1@gmail.com, ²asepm267@gmail.com, ³umar@telkomuniversity.ac.id

Abstrak

Banyak sistem yang sudah digunakan untuk mengatasi masalah banjir. Sistem pendeteksi banjir berbasis *Android* menggunakan *Geographic Information System* yang dibuat adalah suatu sistem yang diterapkan pada masyarakat. Sistem pendeteksi ini terdiri dari *hardware*, *database*, dan *software*. Pada bagian *hardware*, akan digunakan *microcontroller Arduino ATMEGA328* yang terpasang *ultrasonic sensor* dan *GSM module Icomsat V1.1* untuk pembacaan data dan pengiriman data. Pada bagian *database*, akan digunakan *database MySQL*. Pada bagian *software*, akan dibuat sebuah aplikasi pemetaan dan pemantauan banjir berbasis *Android*.

Sistem ini memiliki lima klasifikasi jarak air sungai yang nantinya akan dipetakan pada *software*. Klasifikasi jarak untuk level 1 adalah 1-25 cm, untuk level 2 adalah 26-35 cm, untuk level 3 adalah 36 -70 cm, untuk level 4 adalah 71-150 cm, dan untuk level 5 adalah 151-200cm.

Dari hasil pengujian dapat disimpulkan bahwa sistem ini dapat bekerja secara *realtime* dengan respon pembacaan ketinggian air dan pembacaan peta kurang dari satu detik, tingkat akurasi pembacaan yang dilakukan oleh sensor ultrasonik adalah sebesar 83.33% sampai dengan 100% tergantung permukaan yang diukur, dan rata-rata *delay* dari sistem ini dari mulai pembacaan sensor hingga pembacaan peta adalah 6.93 detik karena terjadi waktu tunggu untuk pengiriman data dari modul GSM ke modem GSM sebesar 5,72 detik.

Kata kunci : *Geographic Information System, Arduino, Android, database*

Abstract

Many systems that have been used to overcome the flood problems. Flood Monitoring System based on *Android* using *Geographic Information System* is a system that is applied to the community. This System will be composed of *hardware*, *database*, and *software*. In the *hardware*, will be used *Arduino microcontroller ATMEGA328* installed *ultrasonic sensor* and *GSM module Icomsat V..1* for reading the data and data delivery. In the *database*, will be used *MySQL database*. In the *software*, will be made an application mapping and monitoring flood based *Android*.

This system has five types of distance water of the river which will be mapped in *software*. Types of distance to level 1 is 1-25 cm, to level 2 is 26-35 cm, to level 3 is 36 -70 cm, to level 4 is 71-150 cm, and to level 5 is 151-200cm. Types was used to display a map is in *software* in accordance with water levels.

From test result can be concluded that the system is able to work with the response *realtime* for reading the height of the water and reading the map of a second, reading accuracy by ultrasonic sensor was 83.33 percent to 100 percent depending on the surface that is measured, and the average delay from this system from reading the sensors to map the reading was 6.93 seconds because there was time waiting for sending data from *GSM module* to modem *GSM* of 5,72 seconds.

Key words : *Geographic Information System, Arduino, Android, database*

1. Pendahuluan

Kebutuhan manusia akan hidup nyaman dan sehat mendorong eksploitasi alam secara besar-besaran, diikuti kemajuan rekayasa dalam pengolahan hasil alam yang bermuara pada proses industrialisasi. Kondisi ini semakin hari tidak semakin berkurang, namun semakin meningkat. Akibatnya timbul bencana alam banjir, tanah longsor, perubahan iklim, pencemaran lingkungan dan beragam penyakit baru.^[10]

Dilain pihak, perkembangan teknologi yang begitu pesat menyebabkan timbulnya pemikiran untuk memanfaatkan teknologi tersebut agar segala hal yang menjadi aktivitas kehidupan mudah dan tidak terlalu

menyita banyak waktu. Salah satu sebagai pemanfaatannya adalah dengan menciptakan suatu peralatan yang mampu bekerja secara mekanik untuk membantu segala aktivitas yang diinginkan.^[10]

Oleh karena itu pada tugas akhir kali ini dibuatlah suatu sistem pendeteksi banjir berbasis *android* menggunakan *Google Maps* untuk pemetaan daerah rawan banjir. Sistem ini bekerja dengan menggunakan *microcontroller arduino* yang terkoneksi dengan *ultrasonic sensor* dan *GSM module* yang akan mendeteksi kenaikan air dan mengirimnya ke server untuk disimpan dalam *database*. Saat *hardware* mendeteksi ketinggian air meningkat, maka sistem ini akan mengirimkan data ke *database* sehingga dalam program *android* yang dibuat juga akan memetakan sesuai data yang masuk di dalam *database*.

2. Dasar Teori^{[1][8]}

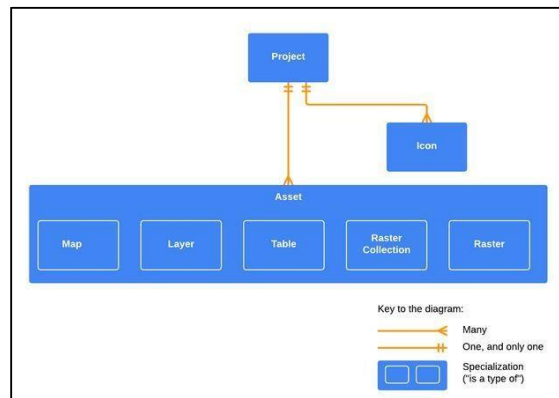
Sistem Informasi Geografis dalam bahasa inggris *Geographic Information System (GIS)* adalah sistem informasi khusus yang mengelola data yang memiliki informasi spasial (bereferensi keruangan) atau dalam arti yang lebih sempit, adalah sistem komputer yang memiliki kemampuan untuk membangun, menyimpan, mengelola dan menampilkan informasi bereferensi geografis, misalnya data yang diidentifikasi menurut lokasinya, dalam sebuah database. Para praktisi juga memasukkan orang yang membangun dan mengoperasikannya dan data sebagai bagian dari sistem ini.

Berikut ini adalah beberapa definisi Sistem Informasi Geografis menurut beberapa ahli :

1. Marbel et al (1983), SIG merupakan sistem penanganan data keruangan,
2. Burrough (1986), SIG adalah sistem berbasis komputer yang digunakan untuk memasukan, menyimpan, mengelola, menganalisis dan mengaktifkan kembali data yang mempunyai referensi keruangan untuk berbagai tujuan yang berkaitan dengan pemetaan dan perencanaan,
3. Chrisman (1997), SIG adalah sistem yang terdiri dari perangkat keras, perangkat lunak, data, manusia (brainware), organisasi dan lembaga yang digunakan untuk mengumpulkan, menyimpan, menganalisis, dan menyebarkan informasi-informasi mengenai daerah-daerah di permukaan bumi.

Salah satu Sistem Informasi Geografis (SIG) yang sering digunakan adalah *Google Maps Engine*. Hal ini dikarenakan *Google Maps Engine* bersifat *opensource* dan mudah digunakan. Salah satu keunggulan dari *Google Maps Engine* adalah pengguna tidak perlu melakukan instalasi untuk editor peta. Semua proses *editing* peta dilakukan secara *online*. Kemudian peta yang sudah diedit di simpan di server google.

Google Maps Engine API menawarkan kontrol program atas data *Google Maps Engine*. *API* google mendukung *CRUD* (*create, read, update, delete*) pada semua *Google Maps Engine* aset: peta, layer, vector tables, raster image dan koleksi raster. Menggunakan *API* google, dapat secara otomatis manajemen data, mengintegrasikan dengan alur kerja yang ada, membangun konektor antara *Google Maps Engine* dengan tools lainnya. Berikut diagram yang menampilkan konsep *Google Maps Engine* :

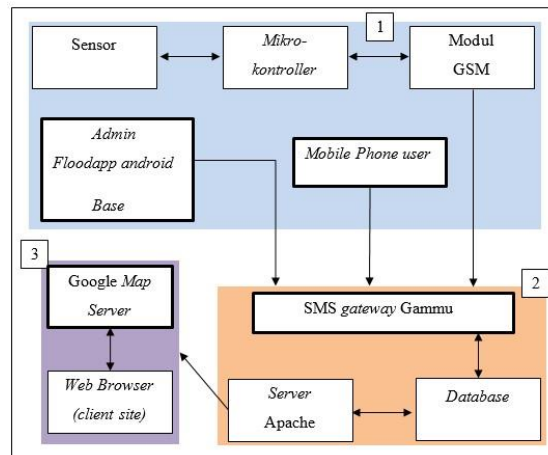


Gambar 2.1 Diagram *Project* dan *Asset*

Google Maps Engine API adalah *RESTful API*. Semua *requests* untuk *API* adalah *HTTP requests*. Oleh karena itu dapat menggunakan bahasa pemrograman dengan library *HTTP* untuk query atau memodifikasi data melalui *API*. *Resources* data direpresentasikan sebagai *JavaScript Object Notation (JSON)*. *JSON* adalah standar *resources* yang di dukung oleh banyak bahasa pemrograman.

3. Perancangan Sistem

Dalam sistem ini terdapat tiga proses utama yaitu *ranging*, *updating*, dan *mapping*. Secara umum rancangan sistem digambarkan sebagai berikut :



Gambar 3.1 Rancangan Sistem

Langkah proses secara singkat berdasarkan gambar dilakukan mengikuti tahapan yang dijelaskan di tabel berikut :

Tabel 3.1 Langkah Proses

Tahap ke-	Proses	Deskripsi
1	<i>Data Colecting</i>	Proses ini dapat dimulai dari sensor ultrasonik melakukan pengukuran ketinggian air. Setelah tinggi air didapat, maka akan diolah untuk diperoleh data berupa klasifikasi level ketinggian air. Selanjut-nya level ketinggian air ini akan dikirim ke server menggunakan modul GSM untuk disimpan dalam <i>database</i> . Dari level ketinggian air inilah yang akan dijadikan input dari sistem.
2	<i>Updating</i>	Proses ini dimulai dari penerimaan data level ketinggian air oleh server yang dikirim oleh <i>hardware</i> (modul GSM). Level ketinggian air tersebut lalu akan disimpan dalam <i>database</i> yang telah dibuat.
3	<i>Mapping</i>	Proses ini menggunakan input level ketinggian air. Program akan memanggil peta sesuai dengan level ketinggian air yang berada pada <i>database</i> . Peta akan selalu berubah apabila sensor ultrasonik mendapatkan level ketinggian yang baru dan dikirim ke server untuk disimpan dalam <i>database</i> .

3.1 Identifikasi Klasifikasi Ketinggian Air

Berdasarkan penelitian dari badan BMPT dan survey ke lokasi, maka di dapatkan data untuk klasifikasi ketinggian air sebagai berikut :

Tabel 3.2 Klasifikasi ketinggian air

No	Klasifikasi Level Air	Ketinggian Air (cm)
1	Level 1	1 – 25
2	Level 2	26 – 35
3	Level 3	36 – 70
4	Level 4	71 – 150
5	Level 5	151 – 200

3.2 Perancangan Sensor

Pada pengukuran jarak atau ketinggian banyak digunakan sensor ultrasonik karena keuntungan dari arah dan kekuatan pemancar ultrasonik yang mudah dikontrol dan sensor ultrasonik tidak berhubungan dengan objek yang sedang diukur. Oleh sebab itu pada sistem yang akan dibuat menggunakan sensor ultrasonik karena sensor ini memenuhi kebutuhan untuk pengukuran ketinggian air. ^[2]

Berikut ini adalah koneksi antara sensor ultrasonik dengan Arduino :

- Kaki *Vcc* sensor ultrasonik tersambung dengan *pin 5V* pada mikrokontroler Arduino,
- Kaki *Ground* sensor ultrasonik tersambung dengan *pin Ground* pada mikrokontroler Arduino,
- Kaki *Trigger* sensor ultrasonik tersambung dengan *pin 12* pada mikrokontroler Arduino,
- Kaki *Echo* sensor ultrasonik tersambung dengan *pin 11* pada mikrokontroler Arduino.

3.3 Perancangan Mikrokontroler

Pada perancangan mikrokontoller sistem ini menggunakan Arduino Uno karena memenuhi kebutuhan untuk berinteraksi dengan perangkat lain yaitu sensor dan modul GSM. Mikrokontroller Arduino Uno ini dipilih juga karena harganya yang relatif murah dan sudah sangat kompleks untuk memenuhi kebutuhan sistem. ^[3]

Berikut ini adalah data yang akan dikirim setelah diproses oleh mikrokontroler :

Tabel 3.3 Klasifikasi pengiriman SMS

No	Klasifikasi Level Air	Ketinggian Air (cm)	Jarak Pada Sensor (cm)	SMS yang dikirim
1	Level 1	1 – 25	13 – 15	INPUT#1
2	Level 2	26 – 35	10 – 12	INPUT#2
3	Level 3	36 – 70	8 – 9	INPUT#3
4	Level 4	71 – 150	5 – 7	INPUT#4
5	Level 5	151 – 200	3 - 4	INPUT#5

3.4 Perancangan Modul GSM

Sistem yang akan dibuat membutuhkan perangkat untuk mengirim data berupa level air berdasarkan pengolahan data dari sensor dan mikrokontroler. Data yang dikirimkan cukup menggunakan SMS (*Short Message Service*). Oleh karena itu dibutuhkan sebuah modul *GSM* untuk memenuhi kebutuhan tersebut. Pada sistem yang dirancang digunakan modul *GSM IComSat v1.1* karena dirasa sudah cukup untuk memenuhi kebutuhan karena dapat mengirimkan data berupa teks. ^[7]

Berikut ini adalah koneksi antara sensor ultrasonik dengan Arduino :

- *Pin 77 Reset Mode modul GSM* terhubung dengan *pin 9* pada mikrokontroler Arduino,
- *Pin Tx* pada modul *GSM* terhubung dengan *pin 2* pada mikrokontroler Arduino,
- *Pin Rx* pada modul *GSM* terhubung dengan *pin 3* pada mikrokontroler Arduino,
- *Vcc* di dapat dari adaptor sebesar 9V,
- *Ground* pada modul *GSM* terhubung dengan *pin Ground* pada mikrokontroler Arduino.

3.5 Perancangan Server

Server diperlukan sebagai tempat penerima *message* dari *hardware* dan akan menyimpannya dalam *database*. Berikut ini adalah spesifikasi *server* dan perangkat yang digunakan :

- Sistem Operasi Windows 8.1 64bit
- RAM 4GB
- HDD 250GB
- *Gammu 1.33.0*
- Modem *GSM*
- *Apache/2.4.10(Win32)OpenSSL*
- 5.6.20 - *MySQL Community Server (GPL)*
- *XAMPP-win32-1.8.3-5-VC11*
- Ethernet dengan internet layanan speedy
- *IP adress PC 192.168.1.119*

3.6 Perancangan Database

Database digunakan untuk penyimpanan data baik data baru ataupun *update* data. *Database* yang digunakan adalah *MySQL* karena mudah dalam penggunaannya. ^[4] Berikut ini tabel yang akan dibuat dalam *database* :

1. Tabel Admin/User (*users*)

Tabel ini berfungsi untuk menyimpan data dari *admin*. Jika *admin* melakukan *login* dan data tersebut sudah tersimpan dalam *database*, maka *user admin* tersebut mampu mempunyai hak ases sebagai *admin*.

2. Tabel Level Banjir (*banjir*)

Tabel ini digunakan untuk menyimpan data *leveling* kenaikan air yang terbaca oleh sensor ultrasonik dan dikirimkan oleh modul GSM ke server.

3. Tabel SMS Pengaduan (*sms_penga-duan*)

Tabel ini digunakan untuk *user* biasa yang ingin mengirimkan pesan berupa kritik, saran, ataupun komplain apabila pada pelaksanaan program ini nanti terjadi banjir dan belum ada bantuan dari pihak pemerintah.

4. Tabel Evakuasi Warga (*evakuasi*)

Tabel ini digunakan untuk data evakuasi banji yang diinputkan admin yang ada dilapangan atau lokasi banjir. Sehingga nantinya dapat dilihat warga mana saja yang di evakuasi dan apasaja keperluannya.

3.7 Perancangan Aplikasi Android^[5]

Aplikasi yang dirancang ini adalah berbasis Android untuk menampilkan pemetaan daerah banjir menurut klasifikasi level yang direkam oleh sensor ultrasonik. Aplikasi ini dibuat menggunakan *PHP*, *Java Android*, dan *MySQL* sebagai database. Hal ini dikarenakan, antara lain :

1. Aplikasi ini digunakan lebih dari satu orang (*multiuser*),
2. Sistem yang akan dibuat membutuhkan mobilitas tinggi,
3. *MySQL* bersifat *opensource*.

4. Implementasi dan Pengujian Sistem

Pada bagian ini merupakan tahap penerapan dan realisasi dari perancangan yang telah dibuat.

4.1 Implementasi Sistem

4.1.1 Implementasi Hardware

Pada sistem yang dibuat, menggunakan tiga komponen yaitu :

- *Mikrokontroller Arduino Uno* yang digunakan sebagai otak dari sistem *hardware* ini,
- *Sensor Ultrasonik HC-SR04* yang digunakan untuk menghitung ketinggian air dalam penentuan level air untuk pemetaan pada *software* nantinya,
- *Modul GSM IcomSat V1.1* yang digunakan untuk mengirimkan data berupa teks yang berisi level air sesuai pengukuran jarak yang dilakukan.

4.1.2 Implementasi Database

Aplikasi *FloodApp* berbasis *Android* ini mempunyai data yang harus disimpan, maka dari itu dibutuhkan *database* untuk menyimpan data tersebut. *Database* beserta tabel yang dibuat akan dipaparkan sebagai berikut :

- Tabel *users*

#	Nama	Jenis	Penyortiran	Atribut Kosong	Bawaan	Ekstra
1	user_id	int(11)		Tidak	Tidak ada	AUTO_INCREMENT
2	username	varchar(32)	latin1_swedish_ci	Tidak	Tidak ada	
3	password	varchar(32)	latin1_swedish_ci	Tidak	Tidak ada	
4	first_name	varchar(32)	latin1_swedish_ci	Tidak	Tidak ada	
5	last_name	varchar(32)	latin1_swedish_ci	Tidak	Tidak ada	
6	email	varchar(1024)	latin1_swedish_ci	Tidak	Tidak ada	
7	active	int(11)		Tidak	0	

Gambar 4.1 Tabel Users

- Tabel *banjir*

#	Nama	Jenis	Penyortiran	Atribut	Kosong	Bawaan	Ekstra
1	<u>banjir_id</u>	int(11)			Tidak	Tidak ada	AUTO_INCREMENT
2	<u>level_air</u>	int(11)			Tidak	Tidak ada	
3	<u>status</u>	int(10)			Tidak	0	
4	<u>change_time</u>	timestamp			Tidak	CURRENT_TIMESTAMP	

Gambar 4.2 Tabel Banjir

- Tabel *sms_pengaduan*

#	Nama	Jenis	Penyortiran	Atribut	Kosong	Bawaan	Ekstra
1	<u>id_pengaduan</u>	int(11)			Tidak	Tidak ada	AUTO_INCREMENT
2	<u>nama</u>	varchar(16)	latin1_swedish_ci		Tidak	Tidak ada	
3	<u>no_hp</u>	varchar(20)	latin1_swedish_ci		Tidak	Tidak ada	
4	<u>isi_pengaduan</u>	text	latin1_swedish_ci		Tidak	Tidak ada	
5	<u>waktu_sms</u>	timestamp	on update CURRENT_TIMESTAMP		Tidak	CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP	

Gambar 4.3 Tabel Sms_pengaduan

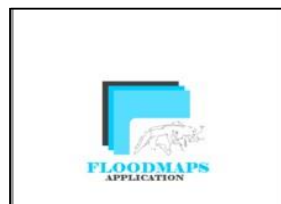
- Tabel *evakuasi*

#	Nama	Jenis	Penyortiran	Atribut	Kosong	Bawaan	Ekstra
1	<u>id_evakuasi</u>	int(16)			Tidak	Tidak ada	AUTO_INCREMENT
2	<u>nama</u>	varchar(64)	latin1_swedish_ci		Tidak	Tidak ada	
3	<u>lokasi</u>	varchar(64)	latin1_swedish_ci		Tidak	Tidak ada	
4	<u>status_evakuasi</u>	varchar(16)	latin1_swedish_ci		Tidak	Tidak ada	
5	<u>Keterangan</u>	text	latin1_swedish_ci		Tidak	Tidak ada	
6	<u>waktu</u>	date			Tidak	Tidak ada	
7	<u>update_time</u>	timestamp	on update CURRENT_TIMESTAMP		Tidak	CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP	

Gambar 4.4 Tabel Evakuasi

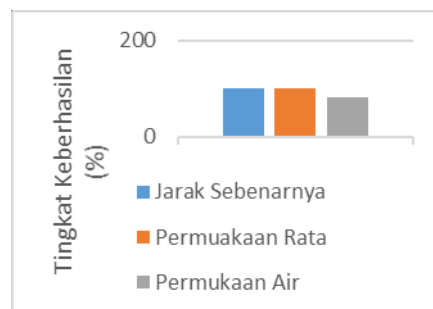
4.1.3 Implementasi Aplikasi

Implementasi antarmuka aplikasi *FloodApp* ini disesuaikan dengan perancangan yang dibuat sebelumnya. Untuk implementasi antarmuka aplikasi adalah sebagai berikut :



Gambar 4.5 Antarmuka Aplikasi

4.2 Pengujian Sistem



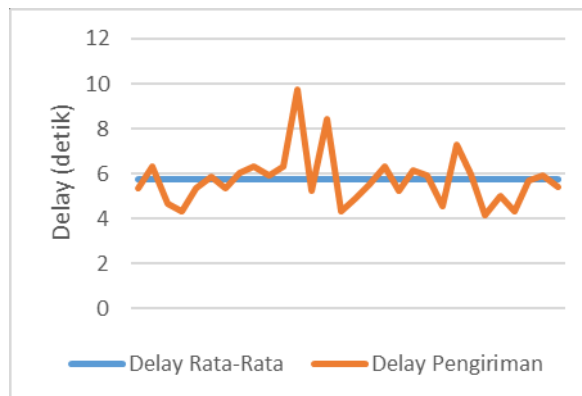
Gambar 4.6 Grafik Pengujian Sensor Ultrasonik

Berdasarkan grafik diatas, dapat diamati linieritas data ultrasonik. Pada permukaan dinding, data yang dihasilkan linier dimana perubahan datanya sebanding dengan jarak sebenarnya. Sedangkan pada permukaan air terdapat beberapa pergeseran nilai yang menunjukkan adanya *error*, yaitu pada jarak 15cm, 25cm, 35cm, 40cm, dan 45cm. Hal ini disebabkan oleh pantulan gelombang suara yang tidak sempurna. Gelombang yang dipancarkan memantul ke air yang memiliki permukaan tidak padat, sehingga pantulan yang diterima oleh *receiver* ultrasonik terdeteksi secara lebih lambat.

Tingkat akurasi pengujian di atas adalah :

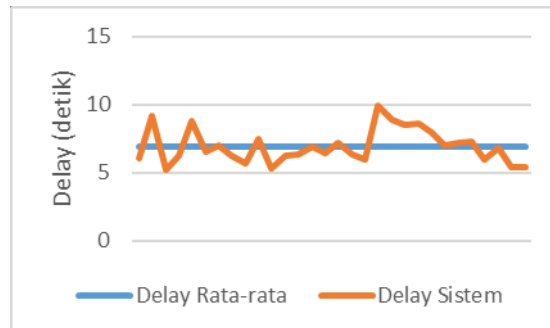
$$\text{Tingkat Akurasi \%} = \frac{\text{pembacaan jarak yg benar}}{\text{total pembacaan}} \times 100\%.$$

- Tingkat akurasi untuk permukaan rata $\frac{30}{30} \times 100\% = 100\%$
- Tingkat akurasi untuk permukaan air $\frac{25}{30} \times 100\% = 83,33\%$



Gambar 4.8 Grafik Delay Pengiriman SMS Modul GSM IcomSat V1.1

Berdasarkan grafik di atas menunjukkan bahwa delay dari pengiriman SMS tidaklah terlalu besar karena hanya membutuhkan waktu rata-rata sekitar 5,72 detik.



Gambar 4.9 Grafik Delay Keseluruhan Sistem

Berdasarkan grafik di atas menunjukkan bahwa delay dari pengiriman SMS tidaklah terlalu besar karena hanya membutuhkan waktu rata-rata sekitar 6,93 detik.

5. Kesimpulan

Berdasarkan hasil pengujian dan analisis yang dilakukan, maka didapat beberapa kesimpulan, yaitu sebagai berikut :

1. Sistem mampu mendeteksi banjir melalui kenaikan air yang terdeteksi oleh sensor ultrasonik,
2. Aplikasi Android *FloodApp* mampu menampilkan peta sesuai dengan klasifikasi ketinggian air secara *realtime*,
3. Sistem mampu mengirim data berupa sms berisi level ketinggian air ke server guna menampilkan data peta dalam aplikasi *FloodApp*,
4. Aplikasi ini mampu menampilkan jarak terdekat untuk mencapai posko evakuasi.

6. Daftar Pustaka

- [1] Prahasta, Eddy. 2009. Sistem Informasi Geografis : Konsep-konsep Dasar (Perspektif Geodesi & Geomatika). Informatika, Bandung.

- [2] Sigit, Riyanto. 2007. Robotika, Sensor, & Akuator. Graha Ilmu, Yogyakarta.
- [3] Khadir, Abdul. 2013. Panduan Praktis Mempelajari Aplikasi Mikrokontroler dan Pemrogramannya Menggunakan Arduino. Andi Publisher, Yogyakarta.
- [4] Wahono, S, Romi.2009. Devinisi PHP dan MySQL. PT. Brainmatics, Jakarta.
- [5] Ardiansyah, F. 2011. Pengenalan Android Programming. Biraynara, Depok.
- [6] *Geographic Information System*. <http://www.westminster.edu/staff/athrock-/GIS/GIS.pdf> (diakses pada 30 Oktober 2014)
- [7] GPS/GPRS/GSM Module V2.0 (SKU:TEL0051) [http://www.dfrobot.com/wiki/index.php?title=-GPS/GPRS/GSM_Module_V2.0_\(SKU:TEL0051\)](http://www.dfrobot.com/wiki/index.php?title=-GPS/GPRS/GSM_Module_V2.0_(SKU:TEL0051)) (diakses pada 31 Oktober 2014)
- [8] *Google Maps Engine API Concepts*. <https://developers.google.com/maps-engine/documentation/definitions> (Diakses pada 10 November 2014)
- [9] *Gammu*. <http://wammu.eu/gammu/> (Diakses pada 10 September 2014)
- [10] Putranto, H. 2011. Sistem Deteksi dan Peringatan Dini Bencana Alam Banjir Berbasis Mikrokontroler ATmega 8535 dan Sms Gateway Di Aliran Sungai Code. *Skripsi*. Jurusan Teknik Informatika. Sekolah Tinggi Manajemen Informatika dan Komputer. Yogyakarta
- [11] Deviana, A. 2011. Kajian Pemodelan Spasial Banjir Untuk Mendukung Kebijakan Sempadan Sungai dan Tata Ruang Wilayah (Studi Kasus Wilayah Pengembangan Baleendah). *Tesis*. Mahasiswa Pascasarjana Pengelolaan Sumber Daya Air. Institut Teknologi Bandung. Bandung
- [12] Arief L.N.a, Purnama B.S.b, Trias Aditya b a. Pemetaan Risiko Bencana Banjir Rob Kota Semarang. 2011. Staf Pengajar Jurusan Teknik Geodesi. Teknik Geodesi UNDIP. Semarang