

SISTEM MONITORING KETINGGIAN AIR BERBASIS INTERNET OF THINGS MENGUNAKAN GOOGLE FIREBASE

WATER LEVEL MONITORING SYSTEM BASED ON INTERNET OF THINGS USING GOOGLE FIREBASE

Eljire Bagas Lewi¹, Unang Sunarya, S.T., M.T.², Dadan Nur Ramadan, S.Pd., M.T.³

¹²³Prodi D3 Teknik Telekomunikasi, Fakultas Ilmu Terapan, Universitas Telkom

¹eljirebagas@student.telkomuniversity.ac.id, ²unangsunarya@tass.telkomuniversity.ac.id,

³dadan.nr@tass.telkomuniversity.ac.id

Abstrak

Indonesia merupakan negara yang memiliki curah hujan cukup tinggi. pada musim penghujan hampir keseluruhan daerah diguyur hujan dengan intensitas yang tinggi. Dan beberapa masyarakat yang belum sadar membuang sampah pada aliran air, menjadi faktor yang memperburuk aliran air. Karena sedikitnya air hujan yang terserap ke tanah. dan terhambatnya aliran air menyebabkan terjadinya banjir, bahkan dapat menyebabkan banjir bandang seperti yang terjadi di kabupaten Garut, Jawa Barat, Selasa (20/09/2016) lalu. Proyek Akhir ini dibuat dengan konsep *Internet of Things* Prinsip kerjanya, sensor *Ultrasonic* akan membaca ketinggian air, dan akan mengirimkan data tersebut ke Web server melalui internet, lalu disimpan di Firebase yang sudah terintegrasi dengan Android. Berdasarkan hasil pengujian, terdapat delay saat pengiriman data dari database ke aplikasi dengan rata-rata delay pengiriman data adalah 0.514 detik. Rata-rata delay pada saat pengiriman data dari *hardware* ke aplikasi adalah 6.69 detik. Rata-rata penggunaan data aplikasi pada saat kondisi *idle* adalah 0.64 MB setiap jam. Aplikasi ini dapat menampilkan data koordinat yang dikirim alat sehingga pengguna mengetahui lokasi dari alat *monitoring*, terdapat tiga status level air yang terprogram yaitu Aman, Siaga dan Bahaya. Jika jarak ketinggian air lebih dari batas yang ditentukan, maka *android* akan memunculkan notifikasi "Bahaya, ketinggian air melebihi batas!".

Kata kunci: *Android, Firebase, Internet of Things, Banjir*

Abstrack

Indonesia is a country that has a high rainfall. during the rainy season almost the entire area of rain with high intensity. And some people are not aware of throwing garbage in waterways, are also factors that aggravate the flow of water. Because the amount of rainwater that is absorbed into the ground is minim and obstruction of water flow causing flooding, and even can cause flash flooding as occurred in Garut district, West Java, on Tuesday (20/09/2016). This Final Project is made with the concept of Internet of Things. Ultrasonic sensors will read the water level, and send the data to the Web server via the internet, then stored in Firebase that is already integrated with Android. Based on the test results, there is a delay when sending data from the database to the application with the average delay of data transmission is 0.514 seconds. The average delay when sending data from hardware to application is 6.69 seconds. The average use of application data at idle condition is 0.64 MB per hour. There are three programmed water level status that is Safe, Standby and Hazard. If the water level distance is more than the specified limit, then the smartphone will bring up the notification "Danger, water height exceeds the limit!".

Keyword : *Android, Firebase, Internet of Things, Flood*

1. Pendahuluan

1.1 Latar Belakang

Indonesia merupakan negara yang memiliki curah hujan cukup tinggi. pada musim penghujan hampir keseluruhan daerah diguyur hujan dengan intensitas yang tinggi. musim hujan biasanya berlangsung sampai dengan 6 – 7 bulan^[8].

Dibangunnya gedung perkantoran, pusat perbelanjaan, perumahan penduduk, dan lapangan parkir yang menggunakan beton sebagai bahan pembangun, membuat daya resapan air hujan ke tanah menjadi berkurang. Membuang sampah pada aliran air, juga menjadi faktor yang memperparah aliran air.

Dengan sedikitnya air hujan yang terserap ke tanah. dan terhambatnya aliran air menyebabkan terjadinya banjir. Banjir menjadi hal

yang lumrah bagi warga Indonesia, namun jika terjadi banjir bandang, seperti yang terjadi di kabupaten Garut, Jawa Barat, Selasa (20/09/2016) lalu sangat memperhatikan. Setiap terjadi hujan lebat mengguyur wilayah Garut ini, banjir mulai melanda di titik-titik rawan banjir. Akibat dari banjir tersebut menimbulkan kerugian yang cukup besar seperti rusaknya rumah-rumah warga, korban jiwa, perubahan ekosistem yang bersifat permanen atau sementara, serta kegiatan warga yang terpaksa harus berhenti sementara. Dari permasalahan tersebut, maka dibutuhkan suatu sistem yang dapat melakukan pemantauan ketinggian air dan peringatan dini akan terjadinya banjir. Beberapa Proyek Akhir sebelumnya telah membuat sistem monitoring ketinggian air dengan judul Sistem Monitoring Ketinggian Air Menggunakan Sensor Ultrasonik Berbasis Android^[5], dengan memanfaatkan teknologi bluetooth untuk pengiriman datanya. Namun terdapat keterbatasan jarak dalam sistem monitoringnya dan tidak dapat memberikan peringatan dini kepada pengguna.

Berdasarkan latar belakang tersebut pada Proyek Akhir ini telah dibuat suatu sistem yang dapat melakukan monitoring ketinggian air bernama *Water Level Monitoring*. Perbedaan *Water Level Monitoring* dengan Proyek Akhir sebelumnya adalah aplikasi dapat melakukan monitoring tanpa ada keterbatasan jarak, dan dapat memberikan peringatan dini akan terjadinya banjir. Sistem pada proyek akhir ini adalah menggunakan sensor *ultrasonik* untuk mengukur batas air, data ketinggian air tersebut akan dikirim ke Firebase melalui server dan dapat dimonitori menggunakan *Smartphone Android*. Sistem monitoring ketinggian air ini juga akan memberikan informasi lokasi alat monitoring sehingga pengguna dapat mengetahui letak alat monitoring.

1.2 Tujuan

Tujuan dari dibuatnya Proyek Akhir ini adalah:

1. Membuat aplikasi android yang dapat melakukan monitoring ketinggian air dari jarak jauh secara *real time*.
2. Membuat aplikasi android yang dapat memberikan informasi kepada pengguna berupa status, ketinggian air, dan lokasi dengan tingkat akurasi yang tinggi.
3. Membuat aplikasi yang dapat memberikan peringatan dini akan terjadinya banjir.

1.3 Rumusan Masalah

Berdasarkan latar belakang yang telah dikemukakan, maka dapat dirumuskan beberapa permasalahan sebagai berikut:

1. Bagaimana cara mengintegrasikan *smartphone* Android dengan Firebase?
2. Bagaimana membuat aplikasi dapat melakukan *monitoring* ketinggian air?
3. Bagaimana menampilkan lokasi alat *monitoring* kedalam peta?
4. Bagaimana membuat aplikasi dapat memunculkan notifikasi ke aplikasi apabila tinggi air sudah berada pada kondisi yang ditentukan?

1.4 Batasan Masalah

Batasan – batasan masalah dalam pengerjaan proyek akhir adalah sebagai berikut:

1. Pengerjaan Proyek Akhir dibatasi hanya mengerjakan bagian monitoring pada *smartphone*.
2. Menggunakan firebase sebagai database.
3. Data yang dimonitori adalah data status, data tinggi air, dan lokasi.
4. Minimal OS pada *smartphone* Android adalah Lollipop.

1.5 Metodologi Penelitian

Metodologi penelitian yang dilakukan adalah eksperimental dengan tahapan sebagai berikut:

1. Studi *literature* dan studi lapangan
Mencari referensi buku dan mencari berbagai komponen pendukung yang akan dipakai dalam perancangan proyek akhir ini serta mempelajari bahasa pemrograman yang akan digunakan.
2. Perancangan
Setelah melakukan studi *literature* dan studi lapangan, selanjutnya dilakukan perancangan, yaitu dengan cara pembuatan kode dan perancangan antar muka.
3. Analisis dan pengujian
Pengujian dilakukan dengan cara mengambil data dari sistem yang dibuat dan menganalisis data tersebut, apakah telah sesuai dengan yang diinginkan atau belum, apabila data yang diperoleh tidak sesuai dengan yang diinginkan maka akan dilakukan pengecekan kembali dan memperbaikinya sehingga diperoleh hasil pengujian yang sesuai.
4. Penarikan kesimpulan
Dari keseluruhan tahapan yang telah dilakukan diatas ditambah dengan masukan dari dosen pembimbing maka dapat diambil kesimpulan dari hasil yang telah dilakukan.

2. Dasar Teori

2.1 Internet of Things

Internet of Things (IoT) dapat diartikan sebagai segala benda yang dapat berkomunikasi dengan benda lainnya, seperti komunikasi *machine to machine* dan komunikasi orang dengan komputer serta akan meluas sampai komunikasi kesegalanya^[8]. IoT juga dapat digambarkan sebagai hubungan benda seperti ponsel pintar, internet TV, sensor, dan aktuator ke internet dimana perangkat yang cerdas memungkinkan untuk dihubungkan bersama – sama membentuk komunikasi antara sesuatu dengan orang dan antara sesuatu dengan dirinya sendiri^[7]. Yang diperlukan dalam terbentuknya IoT adalah database yang besar, jaringan yang menghubungkan antar benda, kemampuan untuk mendeteksi perubahan yang terjadi, dan kepintaran yang tertanam sehingga meningkatkan performansi dari kemampuan memproses data.

2.2 Android

Android merupakan sebuah sistem operasi yang berbasis Linux untuk telepon seluler seperti telepon pintar dan komputer tablet. Android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak^[4].

Android adalah sistem operasi yang bersifat *open source* (sumber terbuka). Disebut *open source* karena *source code* (kode sumber) dari system operasi Android dapat dilihat, di-download, dan dimodifikasi secara bebas. Paradigma *open source* ini memudahkan pengembangan teknologi Android karena semua pihak yang tertarik dapat memberikan kontribusi, baik pada pengembangan system operasi maupun aplikasi^[6]

2.3 Android Studio

Android Studio adalah IDE resmi untuk membangun aplikasi Android berdasarkan IntelliJ IDEA. IntelliJ IDEA sendiri adalah *Java Integrated Development Environment* (IDE) yang dikembangkan oleh JetBrains, untuk mengembangkan perangkat lunak komputer. IntelliJ IDE berfungsi untuk membantu pengguna dalam dunia pemrograman baik dari segi navigasi, penyokong produktivitas, hingga *code editor* yang cerdas.^[6]

2.4 Java

Java adalah bahasa yang dapat dijalankan diberbagai platform, diberagam lingkungan internet, *consumer electronic products*, dan aplikasi komputer. Bahasa *Java* merupakan karya *Sun Micorosystem Inc*^[8]. Pada pengembangan *enterprise applications*, kita menggunakan sejumlah besar paket. Pada *consumer electronic product*, hanya sejumlah kecil bagian bahasa yang digunakan. Masing-masing edisi berisi *Java 2 SDK* untuk mengembangkan aplikasi dan *Java 2 Runtime Envirotment* (JRE) untuk menjalankan aplikasi^[11].

2.5 Google Firebase

Firebase adalah penyedia layanan *real time database* dan *backend* yang saat ini dimiliki oleh *Google*. *Firebase* ini merupakan solusi yang ditawarkan oleh *Google* untuk mempermudah pekerjaan *Mobile Apps Developer*. Dengan adanya *Firebase*, pengembang aplikasi bisa fokus mengembangkan aplikasi tanpa harus memberikan *effort* yang besar untuk urusan *backend*^[11]. Fitur yang dimiliki oleh *Firebase* adalah sebagai berikut:

2.5.1 Firebase Real Time Database

Firebase Real Time Database adalah fitur yang memberikan sebuah *NoSQL database* yang dapat diakses secara *Real Time* oleh pengguna aplikasi. Kelebihan dari *Firebase Real Time Database* adalah aplikasi dapat menyimpan data secara lokal ketika tidak ada akses internet, kemudian melakukan *sync* data segera setelah mendapatkan akses *internet*^[3].

2.6 Web Server

Server atau Web server adalah sebuah software yang memberikan layanan berbasis data dan berfungsi menerima permintaan dari HTTP atau HTTPS pada klien yang dikenal dan biasanya kita kenal dengan nama web browser (Mozilla Firefox, Google Chrome) dan untuk mengirimkan kembali yang hasilnya dalam bentuk beberapa halaman web dan pada umumnya akan berbentuk dokumen HTML^[12].

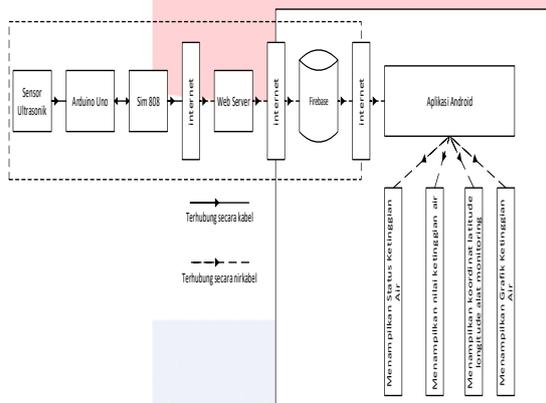
Fungsi utama Server atau Web server adalah untuk melakukan atau akan mentransfer berkas permintaan pengguna melalui protokol komunikasi yang telah ditentukan sedemikian rupa. halaman web yang diminta terdiri dari berkas teks, video, gambar, file dan banyak lagi. pemanfaatan web server berfungsi untuk mentransfer seluruh aspek

pemberkasas dalam sebuah halaman web termasuk yang di dalam berupa teks, video, gambar dan banyak lagi.

3. Perancangan Sistem Dan Implementasi

3.1 Diagram Blok Sistem

Perancangan sistem *Water Level Monitoring* secara keseluruhan dibagi menjadi dua bagian, yaitu bagian *hardware* dan *software*. Diagram blok sistem *Water Level Monitoring* dapat dilihat pada Gambar 3.1.



Gambar 3.1 Diagram Blok Sistem *Water Level Monitoring*

Pada alat *monitoring* ketinggian air, terdapat sensor *ultrasonic* yang berfungsi untuk mengukur ketinggian air, selain *sensor ultrasonic*, terdapat juga modul *sim808* yang berfungsi untuk mengirimkan latitude longitude melalui jaringan internet ke *web server*, lalu akan diteruskan dan disimpan ke *database (Firebase)*.

Setelah *Firebase* menerima data tersebut maka akan diteruskan ke *smartphone android* dan diproses berdasarkan ketinggian air, ketika ketinggian air telah mencapai dan atau melebihi kondisi tertentu, maka akan muncul notifikasi pada *smartphone*.

3.1.1 Web Server

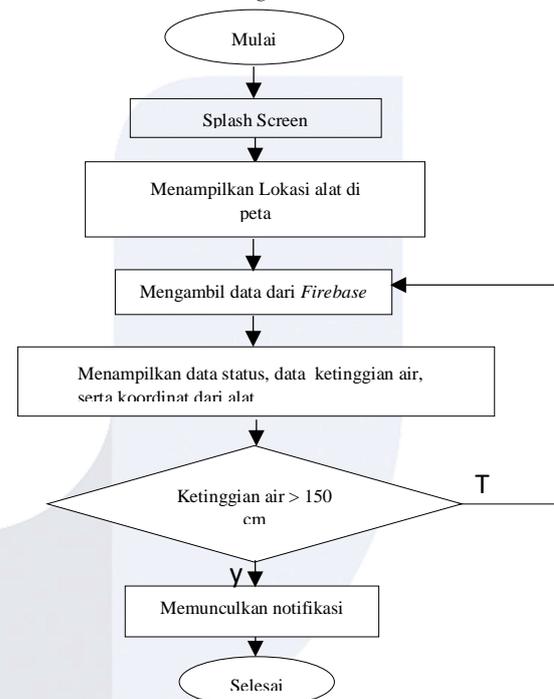
Web server yang akan digunakan pada pembuatan Proyek Akhir ini adalah penyedia layanan hosting gratis *hostinger.id*. *Web server* ini berfungsi sebagai jembatan penghubung antara *Arduino Uno* dengan *Firebase*. *Arduino Uno* tidak dapat mengirimkan data secara langsung ke *Firebase*, hal ini disebabkan oleh alasan keamanan oleh sistem *firebase* itu sendiri, karena *arduino* tidak mempunyai kemampuan enkripsi data.

3.1.2 Database

Pada pembuatan Proyek Akhir ini, penyimpanan data dari perangkat yang ada akan disimpan dengan menggunakan fitur pada *Firebase*. Fitur layanan pada *firebase* ini memudahkan untuk mengembangkan aplikasi yang dibuat sehingga tidak perlu membuat databasenya dari nol. *Firebase database* sudah dilengkapi dengan kemampuan *event handler*, yang mana setiap perubahan yang terjadi di *database* bisa dengan segera dideteksi dan ditangani, sehingga sangat cocok untuk penyimpanan data yang kolaboratif. Pada *database* ini data yang akan disimpan berupa data status, data ketinggian air, dan data lokasi yang berupa latitude dan longitude.

3.2 Diagram Alir Sistem

Gambar 3.2 merupakan diagram alir sistem *Water Level Monitoring*.



Gambar 3.2 Diagram Alir Sistem pada *Water Level Monitoring*

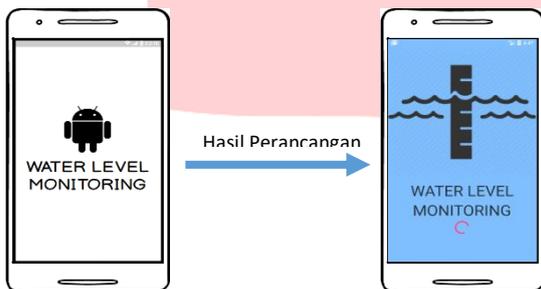
Proses kerja diagram alir sistem pada *Smartphone Android* Gambar 3.2 yaitu, saat memasuki aplikasi sistem aplikasi akan menampilkan *Splash screen*, dan masuk ke tampilan peta yang menunjukkan lokasi (*marker*) alat monitoring. Saat berada pada tampilan peta ini pengguna dapat memilih *marker* untuk melihat data monitoring, lalu *Smartphone Android* akan membaca data yang dikirim dari *Firebase*, dan menampilkannya, setelah itu sistem

mengkalkulasikan berdasarkan ketinggian air, pada saat ketinggian air melebihi 150 cm, maka aplikasi akan segera memunculkan notifikasi kepada pengguna.

3.3 Perancangan Antar Muka Dan Diagram Aktifitas Sistem

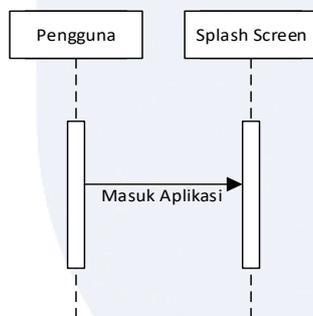
3.3.1 Perancangan Antar Muka Halaman *Splash Screen*

Pada antar muka halaman splash screen menampilkan logo dari aplikasi monitoring ketinggian air dapat dilihat pada Gambar 3.6.



Gambar 3.6 Antar Muka Halaman *Splash Screen*

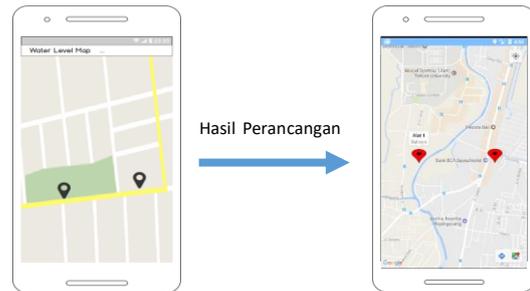
Gambar 3.7 dibawah ini merupakan diagram aktifitas halaman *splash screen*, ketika pengguna memasuki aplikasi *Water Level Monitoring* maka sistem akan memunculkan halaman *splash screen*.



Gambar 3.7 Diagram Aktifitas Halaman *Splash Screen*

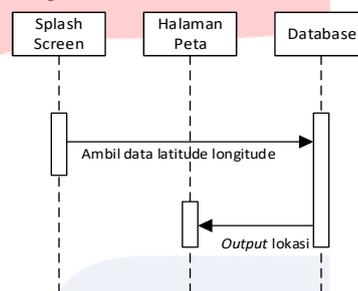
3.3.2 Perancangan Antar Muka Halaman Peta

Gambar 3.9 merupakan antar muka halaman peta, aplikasi akan menampilkan lokasi alat monitoring dengan menggunakan marker sebagai simbolnya, ketika *marker* disentuh maka pengguna akan menuju ke halaman monitoring.



Gambar 3.9 Antar Muka Halaman Peta

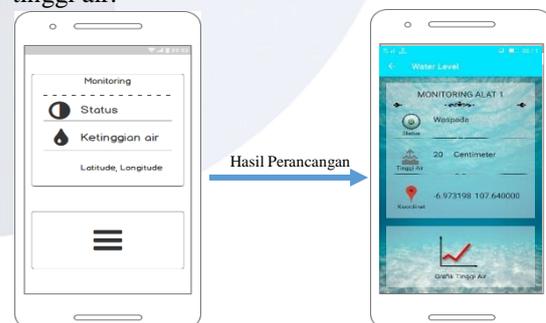
Gambar 3.10 merupakan diagram aktifitas halaman peta, halaman peta merupakan halaman utama dari aplikasi *Water Level Monitoring*.



Gambar 3.10 Diagram Aktifitas Halaman Peta

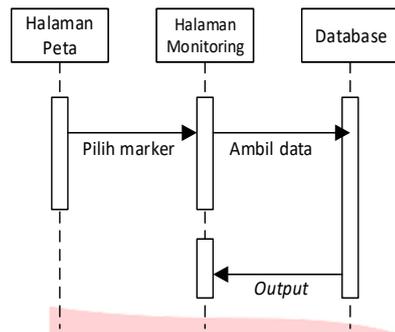
3.3.3 Perancangan Antar Muka Halaman *Monitoring*

Pada halaman *monitoring* ini Gambar 3.12, akan menampilkan data Status, Tinggi air, dan Lokasi dari pada alat *monitoring*. Pada halaman *monitoring* ini juga terdapat tombol Grafik Tinggi Air yang berfungsi untuk menampilkan grafik tinggi air.



Gambar 3.12 Antar Muka Halaman *Monitoring*

Gambar 3.13, merupakan diagram aktifitas halaman *monitoring*, aplikasi mengambil data status, data ketinggian air, dan data koordinat lokasi alat dari database dan menampilkan data tersebut pada halaman monitoring ini.



Gambar 3.13 Diagram Aktifitas Halaman Monitoring

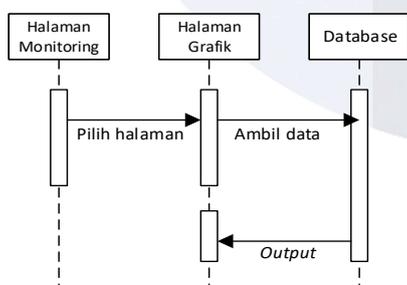
3.3.4 Perancangan Antar Muka Halaman Grafik Ketinggian Air

Pada halaman grafik Gambar 3.15, aplikasi akan menampilkan tanggal dan waktu, serta tampilan grafik Ketinggian Air. Grafik ini menampilkan ketinggian air pada hari tertentu dari Pukul 07:00-21:00.



Gambar 3.15 Antar Muka Halaman Grafik

Gambar 3.16 merupakan diagram aktifitas halaman grafik ketinggian air, pada saat berada pada halaman grafik, aplikasi akan mengambil data ketinggian air dari database dan menampilkannya di halaman grafik.



Gambar 3.16 Diagram Aktifitas Halaman Grafik

4. Pengujian dan Analisa

4.1 Pengujian

Pengujian aplikasi monitoring ketinggian air ini akan dilakukan dengan melakukan black box testing. Pengujian yang dilakukan dengan mengamati hasil eksekusi melalui data uji dan memeriksa fungsionalitas dari *interface* perangkat lunak. Pada pengujian ini akan dilakukan oleh developer selaku pembuat aplikasi.

4.1.1 Pengujian Konsumsi Data

Pengujian ini dilakukan untuk mengetahui berapa banyak data yang dikonsumsi aplikasi. Pengujian dilakukan dalam tiga kondisi, yaitu saat aplikasi dalam kondisi *idle* selama lima jam dan saat aplikasi dalam kondisi aktif memunculkan notifikasi, pada jaringan 4G LTE.

Tabel 4.7 Pengujian Konsumsi Data Saat Kondisi *Idle*

Jam Ke	Konsumsi Data
1	0.2 Mb
2	0.5 Mb
3	0.7 Mb
4	0.8 Mb
5	1 Mb
Rata-rata	0.64 Mb

Berdasarkan tabel 4.7 aplikasi mengkonsumsi data selama lima jam rata-rata sebesar 0.64 Mb atau 640 Kb.

Tabel 4.8 Pengujian Konsumsi Data Saat Aplikasi Memunculkan Notifikasi

Notifikasi Ke	Konsumsi Data
1	404 B/s
2	292 B/s
3	196 B/s
4	404 B/s
5	196 B/s
6	196 B/s
7	192 B/s
8	196 B/s
9	196 B/s
10	196 B/s
Rata-rata	246.8 B/s

Berdasarkan pengujian pada tabel 4.8 rata-rata penggunaan data saat aplikasi memunculkan notifikasi adalah 246.8 B/s.

4.1.2 Pengujian *Delay* Pengiriman Data Hardware Ke Database

Pada pengujian ini akan diuji berapa lama delay yang dihasilkan modul GSM/GPRS saat pengiriman data ke database.

Tabel 4.10 Pengujian Delay Pengiriman Data

No.	Pengiriman data	Lama pengiriman
1.	Data 1	4.32 detik
2.	Data 2	3.86 detik
3.	Data 3	3.43 detik
4.	Data 4	3.43 detik
5.	Data 5	3.43 detik
6.	Data 6	3.43 detik
7.	Data 7	3.43 detik
8.	Data 8	3.43 detik

4.1.3 Pengujian *Delay* Pengiriman Data Database Ke Aplikasi

Pada pengujian ini akan diuji berapa lama delay pengiriman data dari database ke aplikasi. Pengujian ini dilakukan dengan cara mengubah data pada database, lalu menghitung berapa lama waktu yang diperlukan aplikasi untuk melakukan pembaruan data tersebut.

Tabel 4.11 Delay Pengiriman Data Database ke Aplikasi

No	Data	Lama Pengiriman
1	Data 1	0.57 detik
2	Data 2	0.51 detik
3	Data 3	0.47 detik
4	Data 4	0.50 detik
5	Data 5	0.48 detik
6	Data 6	0.48 detik
7	Data 7	0.57 detik
8	Data 8	0.54 detik
9	Data 9	0.51 detik
10	Data 10	0.51 detik

Berdasarkan tabel 4.11 dapat disimpulkan bahwa pengiriman data dari database ke aplikasi dapat dikatakan cepat karena dari setiap data yang dikirim didapat rata-rata lama waktu pengiriman data adalah 0.514 detik.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Setelah melakukan implementasi dan pengujian dari Aplikasi Sistem *Monitoring* Ketinggian air ini, dapat diambil kesimpulan sebagai berikut :

- 1 Aplikasi ini dapat memonitoring ketinggian air tanpa keterbatasan jarak jika mempunyai koneksi internet.
- 2 Berdasarkan pengujian pembacaan lokasi, Aplikasi dapat menampilkan lokasi alat pada peta sesuai data koordinat yang dikirimkan oleh alat *monitoring*, namun ada sedikit perbedaan jarak jika dibandingkan dengan peta *Google Map*, dengan rata-rata perbedaan jarak antara dua titik koordinat yang dikirim oleh alat dan *google Map* adalah 1.230 meter.
- 3 Berdasarkan pengujian pembacaan data, aplikasi dapat membaca dan menampilkan data status, data ketinggian air dan data koordinat dari database secara akurat sehingga mempermudah pengguna dalam melakukan *monitoring*.
- 4 Berdasarkan pengujian notifikasi, aplikasi dapat memberi pemberitahuan ketika ketinggian air telah melebihi batas yang telah ditentukan, sebagai peringatan dini akan terjadi banjir.
- 5 Berdasarkan pengujian penggunaan data, diperoleh rata-rata penggunaan data selama 5 jam saat aplikasi *idle* adalah 0.64 MB, sedangkan rata-rata penggunaan data saat aplikasi aktif memunculkan notifikasi adalah 246.8 Bps.
- 6 Berdasarkan pengujian delay, dapat disimpulkan bahwa pengiriman data dari modul GSM/GPRS ke database cukup lama dengan rata-rata dari data keseluruhan yaitu 6.69 detik. Sedangkan pengiriman data dari database ke aplikasi diperoleh rata-rata *delay* pengiriman data adalah 0.514 detik.

5.2 Saran

Penulis menyadari masih banyak kekurangan yang kedepannya dapat diperbaiki serta dilengkapi. Maka penulis mengharapkan kepada pembaca agar dapat mengembangkan lebih baik lagi aplikasi ini seperti:

1. Membuat *interface* yang lebih menarik dan inovatif, seperti memberi animasi yang menarik saat perpindahan halaman.
2. Pada halaman peta sebaiknya dapat menggunakan peta satelit supaya pengguna dapat mengetahui letak alat secara *detail*.

3. Halaman grafik pada aplikasi hanya dapat melihat tinggi air dalam hari terakhir, disarankan grafik juga dapat melihat tinggi air pada hari-hari sebelumnya.

- [12] Web Server: "Web Server". 17 Juni 2017. <https://idcloudhost.com/pengertian-web-server-dan-fungsinya/>. Diakses pada tanggal 16 Juni 2017.

6. Daftar Pustaka

- [1] Firebase Features. 2016. <https://firebase.google.com/features/>. Diakses pada tanggal 8 Desember 2016.
- [2]. Gambar Anroid Studio: [online]. http://www.angon.co.id/file/2016/04/Tutorial_1_menggunakan-android-studio-1-2.jpg. Diakses pada tanggal 10 November 2016.
- [3]. Oktavianus, Boni. 2016. *[Android] Apa Itu Firebase dan apa saja fiturnya*. Diakses pada tanggal 8 Desember 2016.
- [4] P.K Dixit. 2014. *Android*. Ghaziabad: Vikas Publishing House.
- [5] Ritonga, Esa Nur Leolita. 2014. *SISTEM MONITORING KETINGGIAN AIR MENGGUNAKAN SENSOR ULTRASONIK BERBASIS ANDROID*. Palembang: Politeknik Negeri Sriwijaya.
- [6] Satyaputra, Alfa., Eva Maulina A. 2016. *Let's Build Your Android Apps with Android Studio*. Jakarta: Elex Media Komputindo.
- [7] S. Pandikumar and R. S. Vetrivel, "Internet of Things Based Architecture of Web and Smart Home Interface Using GSM," *Int. J. Innov. Res. Sci. Eng. Technol.*, vol. 3, no. 3, pp. 1721–1727, 2014.
- [8]. S. D. T. Kelly, N. K. Suryadevara, and S. C. Mukhopadhyay, "Towards the implementation of IoT for environmental condition monitoring in homes," *IEEE Sens. J.*, vol. 13, no. 10, pp. 3846–3853, 2013.
- [9] Tenggono, Alfred. 2015. *Sistem Monitoring Dan Peringatan Ketinggian Air Berbasis Web dan SMS Gateway*. STMIK PalComTech, Palembang.
- [10] URL: <http://www.sby.dnet.net.id/dnews/november-2012/article-android-4-2apasaja-fiturbarunya--211.html#UZ44f0o1ya0>, April 2013. Diakses pada tanggal 1 Desember 2016.
- [11] URL: digilib.mercubuana.ac.id/manager/n!@file_s_kripsi/Isi2692441649809.pdf. Diakses pada tanggal 17 Juni 2017