

PERANCANGAN APLIKASI PENDETEKSI DENYUT NADI KORBAN BENCANA ALAM MENGGUNAKAN PERANGKAT *ANDROID* BERBASIS *INTERNET OF THING*

PULSE DETECTION APPLICATION DESIGN ON NATURAL DISASTER VICTIMS USING ANDROID DEVICE BASED ON INTERNET OF THING

Nursyifa Pratiwi¹, Budhi Irawan², Randy Erfa Saputra³

Prodi SI Sistem Komputer
Fakultas Teknik Elektro, Universitas Telkom
JL. Telekomunikasi Terusan Buah Batu 40257, Indonesia

¹nursyifa@telkomuniversity.ac.id, ² budhiirawan@telkomuniversity.ac.id, ³ rasa@telkomuniversity.ac.id

Abstrak

Pada tugas akhir ini akan dibuat sistem perancangan aplikasi untuk membantu korban bencana alam agar mendapatkan penanganan yang cepat dan tepat saat di lokasi kejadian. Sistem akan menggunakan aplikasi *android* yang nantinya akan terhubung dengan alat pendeteksi denyut nadi (*sensor pulse*). Data korban yang sudah terdeteksi akan terkirim ke *cloud service* melalui *gsm* dengan memanfaatkan protokol *REST*. Setelah data masuk dan diolah di *cloud service* maka *user* aplikasi *android* dapat mengakses data melalui *internet* dan melihat data korban yang harus ditolong untuk mendapatkan penanganan yang lebih intensif dengan mencari lokasi keberadaan korban menggunakan *gps*.

Diharapkan sistem aplikasi *android* berbasis *internet of thing* ini dapat berjalan sesuai perencanaan dan membantu mengurangi korban jiwa untuk mendapatkan penanganan medis yang lebih cepat dan tepat.

Kata Kunci : *cloud service, REST protocol, gsm, gps, android*

Abstract

Final project will be made application design system to help the victims of natural disasters in order to get a quick and precise handling at the scene. The system will use android applications that will be connected to the pulse detector (pulse sensor). Victim data that has been detected will be sent to cloud service via gsm by using REST protocol. After the data entered and processed in the cloud service then the user android applications can access data through internet and see the victim data that must be helped to get more intensive treatment by finding the location of the existence of the victim using gps.

Expected that the internet-based android-based application system can run as planned and help reduce the loss of life to get medical treatment faster and more precise.

Keywords: cloud service, REST protocol, gsm, gps, android

I. Pendahuluan

Banyaknya korban jiwa yang sering terjadi pada bencana alam ini membuat tim SAR harus cepat dan tepat untuk mencari para korban di lokasi kejadian yang akan ditangani lebih lanjut oleh tim medis.

Lokasi kejadian bencana alam yang tidak memadai membuat penanganan yang tidak intensif. Para korban harus dibawa ke tempat yang aman untuk mendapatkan pemeriksaan lebih lanjut oleh tim medis. Adanya tenda darurat untuk penanganan sangat dibutuhkan untuk para korban. Dari masalah tersebut, maka dirancang suatu sistem yang membantu para korban untuk mendapatkan bantuan yang lebih cepat. Sistem tersebut merupakan aplikasi berbasis *mobile* yang akan terintegrasi dengan *cloud service* untuk memonitoring keadaan korban bencana alam.

2. Dasar Teori

2.1 REST

REST (*Representational State Transfer*) merupakan protokol yang menggunakan aturan *client-server* , *stateless*, *cache* dan *uniform interface* untuk cara berkomunikasi [1]. Dalam penggunaan *REST* menggunakan protokol *HTTP* [1]. *REST* memiliki *architectural style* dalam penggunaan sistem media distributif. Konsep *REST* dikemukakan oleh Roy Thomas Fielding pada tahun 2000 [2].

Pada arsitektur *REST*, *REST* server menyediakan *resources* (sumber daya/data) dan *REST* client mengakses dan menampilkan *resource* tersebut untuk penggunaan selanjutnya. Setiap *resource* diidentifikasi oleh URIs (*Universal Resource Identifiers*) atau global ID. *Resource* tersebut direpresentasikan dalam bentuk format teks, JSON atau XML. Pada umumnya formatnya menggunakan JSON dan XML.

2.2 Cloud Server

Cloud Server secara umum, definisi cloud computing (komputasi awan) merupakan gabungan pemanfaatan teknologi komputer (komputasi) dalam suatu jaringan dengan pengembangan berbasis internet (awan) yang mempunyai fungsi untuk menjalankan program atau aplikasi melalui komputer – komputer yang terkoneksi pada waktu yang sama, tetapi tak semua yang terkoneksi melalui internet menggunakan cloud computing [3]. Teknologi komputer berbasis sistem Cloud ini merupakan sebuah teknologi yang menjadikan internet sebagai pusat server untuk mengelola data dan juga aplikasi pengguna. Teknologi ini mengizinkan para pengguna untuk menjalankan program tanpa instalasi dan mengizinkan pengguna untuk mengakses data pribadi mereka melalui komputer dengan akses internet [4].

2.3 Internet Of Things (IOT)

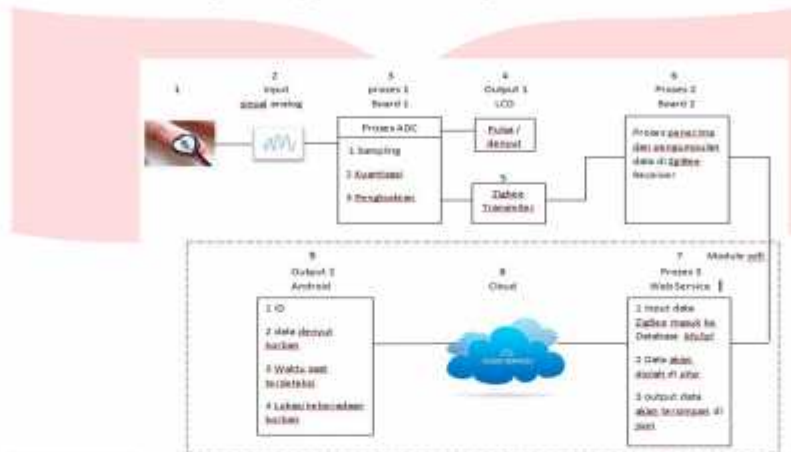
Internet of things (IOT) merupakan sebuah sistem yang berkerja menghubungkan suatu alat (device) dengan sensor agar terhubung ke internet melalui jaringan lokal maupun global. Sensor tersebut akan tertanam di device tersebut dan akan berkerja secara otomatis untuk mengirimkan data. Koneksi iot dibagi menjadi dua jenis area , local area connection dan wide area connection. Local area connection menghubungkan suatu benda dengan benda yang lainnya menggunakan jaringan nirkabel dalam jarak yang terbatas. Koneksi local ini bisanya menggunakan WI-FI, Bluetooth, Zigbee, RFID, dan NFC . Dan untuk penggunaan wide area connection memungkinkan suatu benda terhubung dengan jaringan internet global melalui jaringan GSM, GPRS, 3G, LTE [5].

3. Perancangan

3.1 Perancangan sistem

3.1.1 Dial alir sistem

Dalam skema umum sistem yang akan dijelaskan merupakan hubungan pengerjaan aplikasi sampai *cloud server*. Hubungan komunikasi dilakukan dengan menggunakan aturan protocol *REST API*. Skema umum akan dibuat seperti beri



Gambar 1 Skema umum aplikasi

Berikut merupakan pengerjaan skema umum aplikasi pada gambar :

1. User melakukan login untuk melihat denyut nadi, waktu, id, serta jarak dari korban dan untuk melihat riwayat korban yang sudah terdeteksi.
2. Aplikasi akan melakukan permintaan data melalui *HTTP request* ke alamat yang merupakan *REST API*.
3. *REST API* akan melakukan proses permintaan di server.
4. Server akan mengelola data ke databse berdasarkan proses yang diterima dari *REST API* dan memberikan hasil sesuai permintaan *REST API*.
5. *REST API* akan memberikan hasil dari server berupa data yang diminta dan akan ditampilkan ke user.

4. Analisis

4.1 Parameter Pengujian

Pada pengujian sistem dilakukan beberapa tahap proses pengujian seperti pengujian pada cloud server, pengujian pada sistem aplikasi, dan pengujian performansi perangkat. Pengujian yang dilakukan pada cloud service berupa *blackbox*, *stress server*, dan juga *time response*. Pengujian aplikasi berupa pengujian *blackbox(alpha)* dan juga *whitebox(beta)*. Pengujian performansi perangkat berupa pengujian *operating system (os)* yang digunakan.

4.2 Pengujian Cloud Server

4.2.1 Pengujian Blackbox

Pada pengujian *blackbox* dilakukan sesuai dengan perencanaan pengujian dimana data yang dapat dihasilkan sesuai dengan yang diharapkan. Pengujian *blackbox* untuk cloud server ini dibantu dengan menggunakan *REST consule*. *REST consule* merupakan aplikasi yang melakukan simulasi pengiriman data berupa *HTTP* dan mengikuti aturan *REST*. Berikut merupakan hasil lampiran dari pengujian *blackbox* pada *REST consule*.

Tabel 1 analisis pengujian black box

| No | Skenario Pengujian | Hasil yang diharapkan | Hasil Pengujian | Kesimpulan |
|----|---|--|---|------------|
| 1 | Melakukan request data ke URL/selecthistory dengan salah satu parameter "id"= \$id dengan tipe method <i>GET</i> | Memberikan hasil dengan tampilan json "status" = 1 dan response berupa informasi korban sesuai dengan parameter yang diajukan. | Status 1 dan hasil yang masuk merupakan data korban dengan parameter yang s-esuai | Diterima |
| 2 | Melakukan request data ke URL/select dengan parameter "id"= \$id dengan tipe metode <i>GET</i> | Memberikan hasil dengan tampilan json "status" = 1 dan response berupa informasi korban yang sudah mendapatkan penanganan sesuai dengan parameter yang diajukan. | Status 1 dan hasil yang masuk merupakan data korban yang sudah mendapatkan penanganan | Diterima |
| 3 | Melakukan request data ke URL/his dengan salah satu parameter "status"= true dengan tipe method <i>POST</i> | Memmerikan hasil "status"= 1 dan response mengupdate data koban yang sudah mendapatkan penanganan | Status =1 dan data yang diupdate sesuai dengan parameter yang telah ditentukan | Diterima |
| 4 | Melakukan request data ke URL/daftar dengan salah satu parameter "id" = \$id dengan tipe method <i>GET</i> | Memberikan hasil "status" = 1 dan data berhasil ditambahkan kedalam database berdasarkan parameter yang telah ditentukan | Status = 1 dan data berhasil ditambahkan dengan parameter yang ditentukan | Diterima |
| 5 | Melakukan request data ke URL/About dengan salah satu parameter "id_alat" = \$id_alat dengan tipe method <i>GET</i> | Memberikan hasil "status" = 1 dan data yang dihasilkan sesuai dengan parameter yang telah ditentukan | Status = 1 dan data yang dihasilkan sesuai dengan parameter | Diterima |

Berdasarkan hasil pengujian data yang dihasilkan sesuai dengan apa yang diharapkan. Pengiriman data dari cloud server sesuai dengan penerimaan data di *rest console*. Data yang sudah diuji akan dipanggil di android menggunakan *HTTP* berupa data *json*.

4.2.2 Server stress

Pengujian server stress merupakan pengujian *response server* saat data masuk. Skema pengujian dilakukan dengan simulasi virtual pengiriman data menggunakan salah satu alamat <http://tadnadi.esy.es/select.php?uid=1>. Data pengujian dilakukan menggunakan bantuan tools dari situs <http://loader.io>. Skema tes adalah diakses 200 pengguna virtual secara simultan dalam satu menit. Berikut merupakan gambaran hasil pengujian tersebut.



Gambar 2 Pengujian server stress

Berdasarkan hasil pengujian pada gambar 4.1 dengan skenario pengujian skema 200 akses pengguna virtual dalam satu menit didapatkan penggunaan klien yang stabil dengan *average time*.

4.3 Pengujian aplikasi smartphone

4.3.1 Pengujian blackbox

Pengujian smartphone dilakukan dengan cara *blackbox(alpha)* dan *whitebox(beta)*. Dimana perancangan blackbox berdasarkan scenario perancangan aplikasi dan dapat dilihat hasil skenarion tersebut sesuai dengan keluaran sistem aplikasi. Berikut hasil pengujian *blackbox*.

Table 2 Pengujian Blackbox Aplikasi

| Menu Uji | Hasil Pengujian |
|--------------|--|
| Menu Signup | Memasukan account data diri untuk melakukan login |
| Menu Login | Melakukan login |
| | Melakukan auto login |
| Menu Utama | Memanggil menu history |
| | Memanggil menu measure |
| | Memanggil menu distance |
| | Memanggil menu account |
| | Memanggil menu settings |
| | Memanggil menu search |
| Menu history | Memanggil menu logout |
| | Memanggil data history korban yang sudah teridentifikasi |

Tabel diatas menandakan pengujian pada *blackbox(alpha)*. Untuk pengujian pada *whitebox(beta)* diperlukan adanya survey untuk meminta pendapat orang perihal sistem aplikasi yang dibuat berupa pertanyaan kuisioner. Seseorang yang menggunakan aplikasi ini diharuskan merupakan anggota tim sar yang bertugas menangani bencana alam.

6. Kesimpulan dan Saran

6.1 Kesimpulan

Berdasarkan hasil pengujian pada bab iv maka dapat ditarik beberapa kesimpulan sebagai berikut :

1. Informasi yang dikirimkan ke aplikasi berdasarkan alat *sensor pulse* yang digunakan.
2. Berdasarkan pengujian *server stress* menggunakan virtual. Kestabilan server berada pada 200 *user access* dengan error rate 0.0%
3. Response Time Server untuk salah satu pengambilan data memerlukan waktu 2.2 detik untuk jaringan 3G/HSDPA untuk pengambilan data jaringan 4G LTE dibutuhkan waktu 0.93 detik dan terakhir untuk pengambilan data jaringan WLAN dibutuhkan waktu 0.68 detik.
4. Dalam pengiriman penerimaan data aplikasi jaringan WLAN lebih cepat dibandingkan dengan kedua jaringan lainnya yaitu 3G/HSDPA dan 4G LTE. Dengan selisih 0.59 detik pada pengiriman data riwayat korban.
5. Berdasarkan pengujian beta pada aplikasi. Aplikasi sudah mencapai tujuan meskipun adanya kekurangan untuk tampilan dan penerimaan data aplikasi menggunakan internet. Dengan rating pencapaian tingkat kebutuhan aplikasi adalah 37,86 dan user experience adalah 30,01 dari skala 5,00
6. Berdasarkan pengujian operating system . Aplikasi dapat digunakan di kelima version android yaitu jelly bean, kitkat, lollipop, marshmallow dan juga nougat. fungsi yang ditampilkan pada kelima version tersebut sudah sesuai dengan perancangan aplikasi.

6.2 Saran

Dari penelitian tugas akhir ini dirasa masih banyak kekurangan dalam proses pengujian maupun analisis. Diharapkan ada pengembangan untuk beberapa aspek seperti berikut :

1. Sistem aplikasi masih tergantung kepada internet diharapkan pengembangan untuk sistem jaringan.
2. Pengembangan lebih lanjut untuk pengguna aplikasi yang tidak ditentukan.
3. Mengetahui lebih banyak parameter yang harus digunakan.
4. Dilakukan nya pemutaran video agar tau kondisi korban di tempat kejadian.

7. Daftar Pustaka

- [1] D. M. Elkstein, "Learn REST: A Tutorial: 1. What is REST?," 9 2 2008. [Online]. Available: <http://rest.elkstein.org/2008/02/what-is-rest.html> [Akses 27 Maret 2017].
- [2] H. Hamad, M. Saad and R. Abed, "Performance Evaluation of RESTful Web Services for Mobile Devices," *International Arab Journal of e-Technology*, vol. 1, no. 3, pp.72-78, 2010.
- [3] Kartiko Edhi, "SOAP VS REST" ", November 29,2012.[Online].Available : <https://kartikoedhi.wordpress.com/2012/11/29/soap-vs-rest/> [Akses 27 Maret 2017]
- [4] Wakgus, "Pengertian, Manfaat, Cara kerja dan contoh cloud computing" [Online], Available : <http://pusatteknologi.com/pengertian-manfaat-cara-kerja-dan-contoh-cloud-computing.html> [Akses 27 Maret 2017].
- [5] 2013," An Introduction to the Internet of Things (IoT)". Lopez Research LLC: San Fransisco.