

# Aplikasi Informasi Posko Bencana Alam Berbasis Android

Irma Yunita Setiawati<sup>1</sup>, Andreas Handojo<sup>2</sup>, Anita Nathania Purbowo<sup>3</sup>  
Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Krisen Petra  
Jalan Siwalankerto 121-131 Surabaya 60236  
Telp. (031)-2983455, Fax. (031)-8417658  
m26411091@john.petra.ac.id<sup>1</sup>, handojo@petra.ac.id<sup>2</sup>, anitaforpetra@gmail.com<sup>3</sup>

## ABSTRAK

Dalam kurun 10 tahun terakhir telah terjadi berbagai bencana alam seperti tsunami, meletusnya gunung merapi di Yogyakarta. Karena hal tersebut, banyak didirikan posko untuk menampung orang – orang yang terkena bencana. Banyaknya posko yang ada menimbulkan masalah dalam mengolah informasi mengenai bantuan yang ada. Untuk membantu mengolah informasi posko bencana yang ada maka dibuat aplikasi Informasi Posko Bencana Alam.

Aplikasi ini merupakan aplikasi yang berguna bagi donatur dan orang posko. Aplikasi ini memiliki fitur untuk memberikan informasi data yang ada dari setiap posko sehingga donatur dapat mengetahui bantuan apa yang sedang dibutuhkan posko tertentu. Aplikasi ini memanfaatkan device Android untuk memberikan informasi mengenai posko kepada user. Informasi yang ada dapat berupa logistik posko maupun data pengungsi pada posko. Selain itu bagi penanggung jawab posko, aplikasi ini membantu mengatur logistik posko seperti transfer logistik. Aplikasi ini mampu mengirim broadcast kepada pengguna yang telah melakukan login. Aplikasi ini juga menyediakan informasi dalam bentuk website. Aplikasi ini dibuat dengan Eclipse IDE dalam bahasa pemrograman Java untuk Android dan menggunakan bahasa pemrograman PHP untuk website dengan framework CodeIgniter.

Hasil pengujian menunjukkan bahwa aplikasi ini berjalan baik pada beberapa device seperti Android 4.2 (Jelly Bean) dan Android 4.4(Kitkat). Aplikasi dapat menjalankan fitur informasi posko, broadcast, maintain data logistik dan mencari lokasi posko dengan baik.

**Kata Kunci:** Android, *Google Cloud Messaging*, Posko Bencana Alam, *Google Map*.

## ABSTRACT

*Within the last 10 years, there have been so many natural disasters such as tsunami, earthquakes, and the eruption of Mount Merapi in Yogyakarta. Many emergency shelters were built to accommodate the disasters survivors. It caused problems in processing information regarding existing aid. An application named "Post Natural Disaster Information" was created to make it easier to process the information.*

*This is an useful application for both the donors and the posts. This app has a feature to provide information on available data from each post, so that donors know the needs of certain posts. It takes advantage of Android devices to provide information about the posts to the users. It could be posts' logistics or data shelters refugees in post. In addition to the person in charge of the post,*

*this application helps organizing the logistics command post as logistical transfer. This application could send a broadcast to the users who have been already logged in. It also provides information in the form of a website. This application was made by Eclipse IDE in the Java programming language for Android and using the PHP programming language for websites with CodeIgniter framework.*

*The test results show that the application is running well on some devices such as Android 4.2 (Jelly Bean), Android 4.4 (Kitkat). Applications can run the update feature post, broadcast, maintain logistics data and find posts locations well.*

**Keywords:** Android, *Google Cloud Messaging*, *Shelter*, *Google Map*.

## 1. PENDAHULUAN

Indonesia merupakan negara yang memiliki total 17.508 pulau hingga saat ini. Selain itu, Indonesia secara geografis terletak di antara 2 samudra yakni Samudra Hindia dan Samudra Pasifik serta berada di antara 2 benua yaitu Benua Asia dan Benua Australia. Secara geologis, Indonesia terletak pada pertemuan tiga lempeng utama dunia yaitu lempeng Indo-Australia, lempeng Eurasia, dan lempeng Pasifik. Dengan letak geografis dan geologis tersebut membuat Indonesia berada pada daerah rawan bencana seperti gempa bumi, banjir, tsunami, tanah longsor serta letusan gunung berapi. Dengan adanya banyak bencana, didirikan posko untuk menampung orang – orang yang terkena bencana. Tim penyelamat dapat mendirikan beberapa posko pada suatu daerah.

Banyaknya posko yang ada menimbulkan masalah dalam mengolah informasi mengenai bantuan yang ada. Seringkali yang terjadi adalah ada beberapa posko yang berlimpah barang bantuan seperti makanan maupun obat – obatan. Sementara itu posko yang lain akhirnya menjadi kekurangan bantuan karena tidak kurangnya bantuan yang diberikan pada posko tersebut. Masyarakat tidak mengetahui barang apa saja yang diperlukan, fasilitas yang telah dimiliki, letak posko, dan data mengenai pengungsi. Selain itu tidak adanya data pengungsi membuat kerabat atau teman sulit untuk melakukan pencarian. Hal ini disebabkan karena kurangnya informasi yang didapat oleh masyarakat.

Melihat kebutuhan dan teknologi yang ada maka akan dibuat aplikasi berbasis Android untuk mengolah informasi mengenai posko bencana alam. Dari hasil pembuatan ini diharapkan dapat membantu melihat informasi suatu posko pada seluruh posko, melihat letak posko tersebut berada, dan melihat secara spesifik data mengenai posko di antaranya letak posko, barang dan fasilitas yang tersedia pada posko, barang yang diperlukan, serta

informasi mengenai pengungsi. Selain itu dapat membantu pengguna dengan memberikan informasi kepada pengguna jika terdapat bencana alam.

## 2. LANDASAN TEORI

### 2.1 Sistem Operasi Android

Android merupakan sistem operasi yang dibuat oleh Linux. Sistem operasi ini diperuntukkan bagi *smartphone* maupun tablet. Android telah dirilis sejak tahun 2007. Android dikembangkan oleh Android, Inc untuk menyaingi Symbian dan Windows mobile. Kelebihan sistem operasi Android adalah menyediakan *platform* terbuka di mana pengembang dapat menciptakan berbagai aplikasi yang nantinya dapat dipergunakan dalam berbagai macam *mobile device*.

Android menggunakan bahasa XML serta dasar scripting Apache Ant untuk proses build [1]. Dengan adanya Apache ini pengembang aplikasi Android diperbolehkan untuk mendistribusikan aplikasi di bawah skema lisensi yang diinginkan.

Terdapat 4 hal mendasar yang harus dipahami ketika membangun sebuah aplikasi Android [3]:

1. *Activity*: merupakan tampilan grafis yang dilihat ketika menjalankan sebuah aplikasi. *Activity* yang ada dalam sebuah aplikasi dapat lebih dari 1.
2. *Intent*: merupakan *value* yang menggambarkan apa yang harus dilakukan apabila melakukan perpindahan layar
3. *Service*: merupakan layanan yang akan bekerja pada *backend*.
4. *Content Provider*: *content* ini memungkinkan aplikasi dapat menerima dan menyimpan data yang didapat dari *database*.

Android SDK merupakan API tools yang digunakan untuk pengembangan aplikasi Android. Bahasa yang digunakan dalam Android SDK ini adalah Java. Beberapa fitur yang tersedia di Android adalah:

- Kerangka aplikasi yang memungkinkan penggunaan dan penghapusan komponen yang tersedia.
- Grafik berupa grafik 2D dan 3D yang berdasarkan OpenGL.
- SQLite yang digunakan untuk penyimpanan data.
- Kamera, *Global Positioning System* (GPS) serta kompas
- Lingkungan *development* yang lengkap untuk melakukan pengembangan seperti tersedianya *plugin* untuk Eclipse IDE
- Terdapat *integrated browser* yang didasarkan pada WebKit (*open source engine*).

### 2.2 Web Services

*Web Services* memungkinkan sebuah fungsi yang berada di dalam sistem dapat diakses oleh aplikasi lain dimana aplikasi tersebut tidak perlu mengetahui *source code* yang terdapat di dalam sistem. *Web services* diadaptasi pada *website* sehingga *protocol* jaringan defaultnya adalah HTTP[2].

Keuntungan menggunakan *web services* adalah sebagai berikut:

1. *Web services* dapat digunakan untuk mentransformasikan satu atau beberapa objek yang terpisah dalam satu ruang lingkup
2. Dalam *web services* tingkat keamanan dapat ditangani dengan baik.

3. *Web services* memiliki kemudahan dalam proses deployment.

*Web services* berjalan di port 80 yang merupakan protokol standar HTTP. Dengan demikian *web services* tidak memerlukan konfigurasi khusus di sisi firewall.

### 2.3 REST Web Service

REST (*Representational State Transfer*) merupakan sebuah teknik arsitektur *software* untuk sistem terdistribusi seperti *World Wide Web*. REST mendekati kumpulan *constraint* (batasan) yang ketika diterapkan pada desain sebuah sistem, akan menjadi jenis arsitektur perangkat lunak. *RESTful web services* berhubungan dengan mentransfer representasi *resource*. Representasi tersebut dapat berbentuk JSON atau XML. Setiap *record* data individu dalam sebuah sistem merupakan *resource* [5]. *RESTful API* memiliki atribut sebagai berikut [4]:

- Merupakan *client / server* : RESTful API membedakan peran antara *client*, *entity* yang membuat permintaan untuk *service*, dari *server*, *entity* yang mendengarkan permintaan tersebut dan memasok *service*.
- Bersifat *stateless* : Setiap *request client* berisi semua informasi yang diperlukan untuk server memproses *request* sepenuhnya.
- Menggambarkan pertukaran representasi nama *resource*, bukan pertukaran *resource sendiri*.
- Memiliki *uniform interface* : Dikenal sebagai *constraint* dari REST. RESTful API mendukung empat CRUD standar metode *insert*, *update*, *delete*, dan *get* (PUT, POST, DELETE dan GET pada HTTP).

### 2.4 Google Maps API

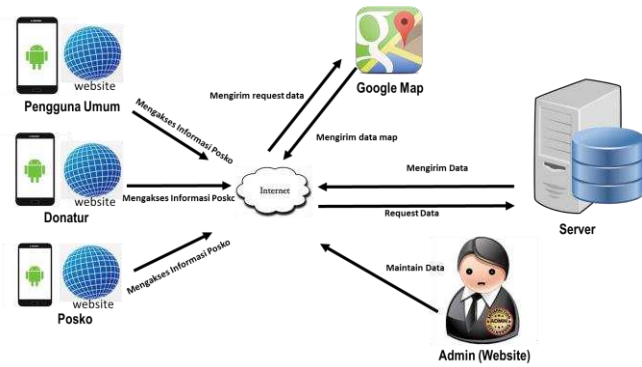
*Google Maps* diperkenalkan dalam *blogspot* pada Februari 2005. *Google Maps API* memungkinkan memanfaatkan kekuatan *Google Maps* untuk digunakan dalam aplikasi sendiri untuk menampilkan data sendiri [6]. Beberapa fitur yang disediakan oleh *Google Maps API*:

- *Google Maps Javascript API v3*: *Google Map API* ini tersedia dalam bahasa *javascript*.
- *Google Maps API web service*: tersedianya layanan *Google Maps API* untuk fungsi lanjutan seperti *direction*, *Geocoding*, *Distance Matrix API* dan *elevation API*.
- *Google Place API*: API yang digunakan untuk membuat aplikasi *location based service*.

## 3. DESAIN SISTEM

### 3.1 Garis Besar Sistem

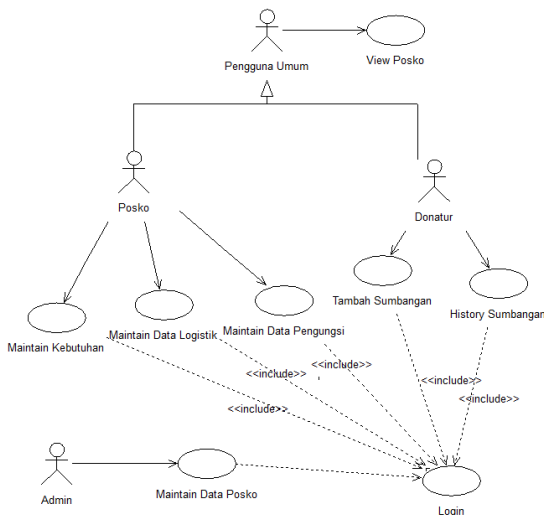
Aplikasi Informasi Posko Bencana Alam ini merupakan aplikasi yang bertujuan agar masyarakat (*user*) dapat mengakses informasi mengenai posko bencana alam dengan lebih mudah. Sistem aplikasi ini terdiri dari 2 komponen utama yaitu pengguna, *Android device* serta *server*. Pengguna dapat mengakses informasi posko dengan menggunakan *Android device*. *Android device* akan meminta data yang diinginkan melalui *server*. *Server* akan memberikan data yang diinginkan. Akses data yang dilakukan menggunakan REST. Admin bertugas untuk melakukan *maintenance* data pada server. Rancangan desain arsitektur aplikasi dapat terlihat pada Gambar 1.



Gambar 1. Desain Arsitektur Aplikasi

### 3.2 Use Case Diagram

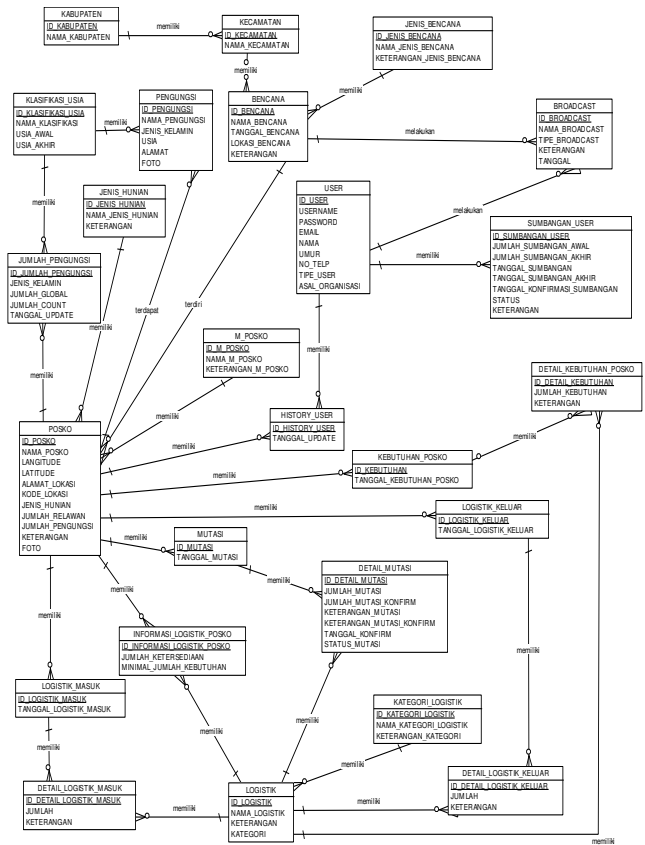
Dalam *use case diagram* ini terdapat 4 aktor yang berperan dalam aplikasi ini. Aktor tersebut antara lain *user* umum, penanggung jawab posko, donatur serta administrator. Penanggung jawab posko dan donatur merupakan generalisasi dari *user*. *User* umum merupakan *user* yang dapat mengakses data mengenai posko. Penanggungjawab posko memiliki hak akses untuk memasukkan data logistik, kebutuhan serta data pengungsi. Selain itu penanggung jawab posko juga memiliki fungsi untuk melakukan konfirmasi sumbangan. Sedangkan donatur/penyumbang memiliki hak akses untuk menggunakan fitur sumbangan. Dalam fitur sumbangan ini terdapat 2 fungsi yaitu menambah data sumbangan serta melihat history dari sumbangan. Dapat dilihat pada Gambar 2.



Gambar 2. Use Case Diagram

### 3.3 Entity Relationship Diagram

*Entity Relationship Diagram* (ERD) menggambarkan mengenai data dalam entitas serta hubungan antara entitas satu dengan yang lainnya. Ketika entitas berhibungan dengan entitas lain akan terdapat relasi tertentu seperti *one to one*, *one to many* dan *many to many*. Setiap relasi tersebut akan digambarkan pada *Entity Relationship Diagram* yang ada. Gambar 3 merupakan desain *Entity Relationship Diagram* dari aplikasi yang dibuat.



Gambar 3. Entity Relationship Diagram

## 4. IMPLEMENTASI SISTEM

Implementasi sistem terdiri dari implementasi *web service*, aplikasi Android, dan *website*.

### 4.1 Web Services

Sebagai perantara antara aplikasi Android dan *server*, *web service* terdiri dari beberapa *function*. *Function* ini terdapat pada controller. *Function* ini diakses oleh aplikasi android dengan metode HTTP GET dan POST. Selain itu *function* yang ada juga digunakan untuk mengambil data dari aplikasi Android dan menyimpan data pada *server*.

Aplikasi pada Android menggunakan *web services* REST. Fungsi yang digunakan adalah GET dan POST. Fungsi GET digunakan untuk mengambil data pada *server*. Untuk mengambil data pada *server* terhubung yang digunakan adalah mvc CodeIgniter. Fungsi GET antara lain digunakan untuk menampilkan jenis bencana, nama posko dan daftar pengungsi. Sedangkan fungsi POST digunakan untuk mengirim data dari Android pada *server*. Dalam Android *services* ini berjalan pada *background*. Ketika data yang diambil tidak terpenuhi maka akan akan ditampilkan pesan pengambilan data gagal. *Services* ini menggunakan fungsi *asyncTask* yang telah terdapat pada Android. Fungsi inilah yang akan berjalan pada *background*.

### 4.2 Aplikasi Android

Implementasi aplikasi Android berdasarkan perancangan yang telah dibuat, seperti tertulis pada bab sebelumnya. Aplikasi Android ini dibuat dengan menggunakan bahasa pemrograman

Java dan menggunakan Eclipse IDE. Aplikasi Android ini menggunakan *web service* REST sebagai jembatan untuk berhubungan *server*. Fungsi yang digunakan antara lain untuk mengambil data bencana, posko, pengungsi maupun untuk memasukkan data logistik.

### 4.3 Website

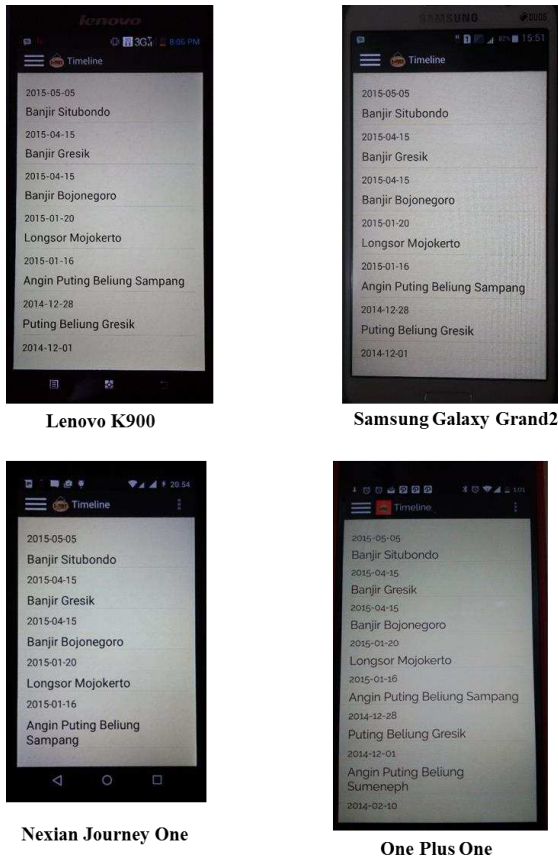
Implementasi *website* ini dapat diakses oleh pengguna umum, penanggungjawab posko serta administrator. *Website* ini menggunakan PHP *framework*, yaitu CodeIgniter. CodeIgniter tersebut menerapkan pendekatan *Model-View-Controller* (M-V-C), sehingga program terdiri dari 3 bagian, yaitu *Model*, *View*, dan *Controller*. *Website administrator* ini terdiri dari proses CRUD (*create, read, update, delete*) terhadap *database server*. Fungsi yang dimiliki *website* hampir sama dengan yang ada pada Android.

## 5. PENGUJIAN SISTEM

Penulis menggunakan beberapa *device* Android untuk melakukan pengujian sistem aplikasi Android. *Device* yang digunakan adalah Samsung Galaxy Grand 2, One Plus One, Lenovo K900, dan Nexian Journey One. Pengujian dilakukan untuk setiap fitur aplikasi dan dilakukan pada setiap *device* tersebut.

### 5.1 Pengujian Halaman Awal Android

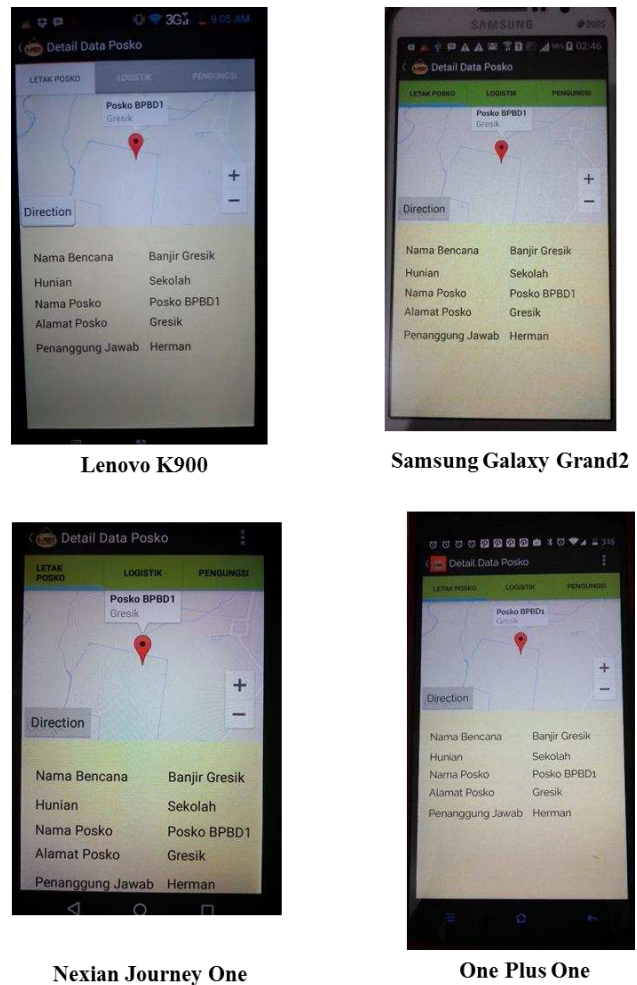
Halaman Awal ketika *user* membuka aplikasi ini. Ini merupakan daftar bencana yang terjadi. Dari daftar bencana ini *user* dapat memilih posko yang diinginkan. Pengujian halaman awal Android dapat dilihat pada Gambar 4.



Gambar 4. Halaman Awal Aplikasi

### 5.2 Pengujian Informasi Posko

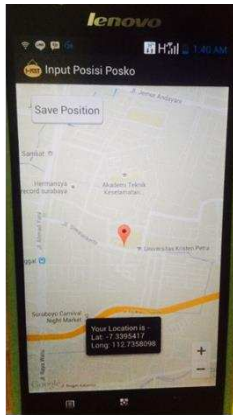
Informasi posko yang dapat dilihat adalah informasi mengenai lokasi posko, posisi posko, keadaan logistik yang ada pada posko serta pengungsi yang ada di posko tersebut. Informasi ini dapat dilihat pada 3 tab yang berbeda. Tab pertama merupakan tab 'Letak Posko'. Tab ini berisi mengenai informasi posko termasuk didalamnya penanggungjawab posko dan lokasi posko. Tab yang kedua berisi mengenai logistik posko. Pada tab ini *user* dapat melihat informasi logistik yang ada pada posko. Informasi ini diharapkan dapat membantu *user* untuk menentukan bantuan apa yang ingin diberikan pada posko tersebut. Yang ketiga merupakan tab pengungsi. Melalui tab ini *user* dapat melihat informasi pengungsi yang terdapat pada posko. Pengujian informasi mengenai letak posko dapat dilihat pada Gambar 5.



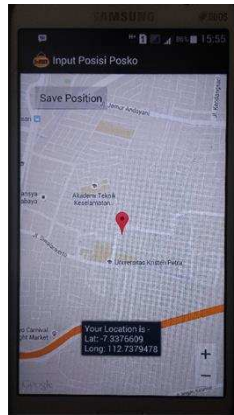
Gambar 5. Halaman Informasi Posko

### 5.3 Pengujian Posisi Posko

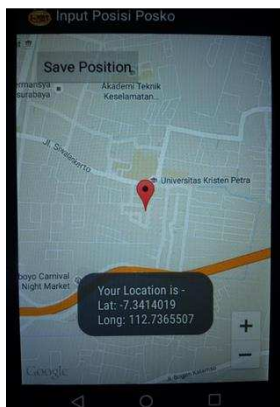
Pengujian ini adalah pengujian halaman posisi posko. Posisi ini digunakan untuk menentukan posisi posko yang ada dengan bantuan GPS. Fitur ini dapat digunakan oleh penanggungjawab posko. Ketika penanggungjawab posko menekan tombol 'Save Position' maka aplikasi akan meminta untuk mengaktifkan GPS. Aplikasi akan mencari posisi pengguna. Halaman ini dapat dilihat pada Gambar 6.



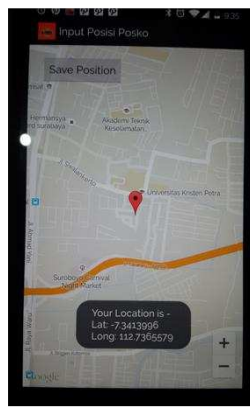
Lenovo K900



Samsung Galaxy Grand2



Nexian Journey One



One Plus One

Gambar 6. Memasukkan Posisi Posko

### 5.4 Pengujian Halaman Administrator

Halaman Administrator digunakan oleh administrator untuk melakukan CRUD (*create, update, delete*) master data. Master data yang ada antara lain master jenis bencana, bencana, posko, kabupaten/kota, kecamatan dan hak akses user. Gambar 7 merupakan tampilan pengujian halaman administrator.

Data Bencana							
Add Record							
Show 10 entries							
Jenis	Nama	Tanggal	Lokasi	Kabupaten	Kecamatan	Keterangan	Actions
Gempa Bumi	Banjir Gresik	15/04/2015	Gresik	Gresik	Cermal	Gresik	View Edit Delete
Gempa Bumi	Erupsi Gunung Kelud	10/02/2014	Kediri	Kediri	Cermal	Melutusnya anak gunung kelud	View Edit Delete
Gempa Bumi	Banjir Bojonegoro	15/04/2015	Bojonegoro	Gresik	Benjeng	Banjir	View Edit Delete

Gambar 7. Pengujian Halaman Administrator

### 5.5 Pengujian Informasi Pada Website

Ketika membuka halaman awal *website*, tampilan yang muncul adalah fitur untuk informasi logistik posko. Fitur ini menampilkan informasi logistik yang ada pada posko tertentu. Pengguna memasukkan nama posko yang diinginkan melalui fitur *search*. Di kanan tampilan terdapat kategori logistik yang ada pada posko. Jika button kategori logistik ditekan maka akan muncul informasi

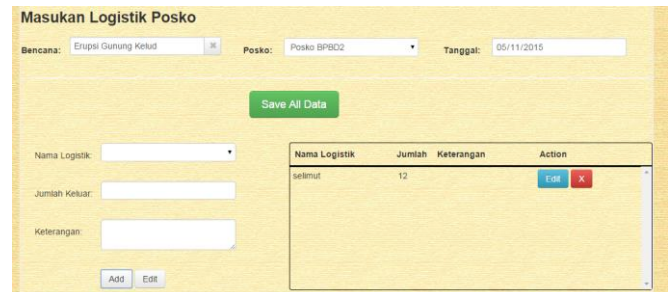
logistik pada kategori tersebut. Secara *default* ketika pengguna menekan tombol *search* yang akan muncul adalah informasi logistik posko pada kategori paling atas. Pengujian halaman Logistik Posko dapat dilihat pada Gambar 8 Pengujian Halaman Informasi Posko.



Gambar 8. Halaman Informasi Posko

### 5.6 Pengujian Halaman Data Logistik Masuk

Data logistik juga dapat dimasukkan melalui *website*. Untuk memasukkan data *user* perlu menentukan posko dan tanggal input data. Ketika semua data logistik telah diinputkan, *user* dapat menekan tombol 'Save All' untuk menyimpan semua data ke dalam *database*. Gambar 9 merupakan tampilan pengujian halaman logistik masuk.



Gambar 9. Pengujian Halaman Administrator

## 6. KESIMPULAN

Berdasarkan hasil pengujian, dapat diambil kesimpulan sebagai berikut:

- Fitur – fitur pada aplikasi ini sudah dapat berjalan dengan baik, fitur meliputi input logistik, kebutuhan, transfer logistik, informasi pengungsi.
- Aplikasi ini mampu memberikan informasi berupa broadcast melalui *email* dan *Google Cloud Messaging*.
- Aplikasi ini mampu menunjukkan rute menuju posko dengan bantuan GPS dan google map.
- Aplikasi ini dapat memberikan informasi kepada pengguna mengenai data logistik dan pengungsi yang ada
- Aplikasi ini berjalan baik pada website dan Android dengan versi 4.1.2, 4.4.2 dan 5.1.2
- Aplikasi harus selalu terkoneksi dengan internet untuk mendapat kan data yang akurat
- Rata – rata pengiriman data adalah 2.5 kb/detik untuk jaringan 2G. Sedangkan untuk jaringan 3G rata – rata pengiriman data adalah 10.61 kb/detik.

## 7. REFERENSI

- [1] Burton, M., & Felker, D. 2012. *Android Application Development For Dummies, 2nd Edition*. Hoboken: John Willey & Sonc,Inc.
- [2] Cerami, E. 2002. *Web Services Essentials*. United State: O'Reilly Media.
- [3] Cinar, O. 2012. *Android Apps with Eclipse*. United State of America: Apress.
- [4] Mednieks, Z., Meike, G., Dornin, L., & Zan. 2013. *Programming Android Database Applications for the Enterprise*. Indianapolis, Indiana: John Wiley & Sons, Inc.
- [5] Mitchell, J. L. 2013. *PHP Web Services*. Sebastopol: O'Reilly Media.
- [6] Svennerberg, G. 2010. *Beginning Google Maps API 3*. United State of America: Apress.