

PEMBUATAN APLIKASI WISATA SEJARAH PERTEMURAN SURABAYA 1945 BERBASIS ANDROID

Sandy Sunaryo¹, Andreas Handojo², Justinus Andjarwirawan³

Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Kristen Petra

Jl. Siwalankerto 121 – 131 Surabaya 60236

Telp. (031) – 2983455, Fax. (031) - 8417658

E-Mail: light.sandy@gmail.com¹, handojo@petra.ac.id², justin@petra.ac.id³

ABSTRAK: Terdapat banyak lokasi-lokasi bersejarah di Surabaya yang merupakan peninggalan pertempuran Surabaya pada tahun 1945. Hal ini menarik para wisatawan untuk berkunjung ke Surabaya. Para wisatawan pada umumnya mencari informasi melalui berbagai media seperti *internet* sebelum melakukan kunjungan. Namun, informasi yang didapat pada umumnya kurang terorganisir untuk menggambarkan sejarah pertempuran Surabaya dengan baik. Dengan memanfaatkan *smartphone* yang banyak digunakan saat ini, sebuah aplikasi dirancang untuk membantu para wisatawan dalam mencari lokasi dan informasi mengenai lokasi sejarah pertempuran Surabaya. Aplikasi ini menyediakan akses ke informasi lokasi, sejarah lokasi, galeri foto lokasi, daftar lokasi yang telah dikunjungi, menjalankan *game*, *virtual stamping*, simulasi pertempuran Surabaya dan *download* konten. Adapun aplikasi ini dirancang untuk beroperasi pada *smartphone* dengan sistem operasi Android. Berdasarkan hasil pengujian, aplikasi ini dapat menjalankan fungsi *login*, peta, *game*, *virtual stamping*, simulasi sejarah dan *download* konten serta fungsi lain yang telah disebutkan.

Kata Kunci: Pertempuran Surabaya , Android, SQLite, GPS.

ABSTRACT: *There are a lot of historical places in Surabaya which are legacies from battle of Surabaya. This fact attracts tourists to visit Surabaya. Generally they searched for information related to the places through a variety of media such as internet before making a visit. However, obtained information are usually not well organized to describe the battle of Surabaya properly. By utilizing the widely used smartphone, an application is designed to aid the tourists in looking for location and information about battle of Surabaya.*
The application provides access to location's information, location's history, photo gallery of location, list of visited locations, run a mini game, virtual stamping, simulation of history on battle of Surabaya and downloading new contents for the application. This application is designed to work on smartphone with Android operating system.
Based on testing result, this application is running login, map, game, virtual stamping, history simulation, content download and other functions that have been mentioned properly.

Keywords: *Battle of Surabaya, Android, SQLite, GPS.*

1. PENDAHULUAN

Surabaya adalah kota yang dikenal sebagai kota pahlawan dimana banyak peristiwa bersejarah yang telah terjadi di Surabaya, salah satu peristiwa yang paling terkenal adalah pertempuran Surabaya pada tanggal 10 November 1945 yang sekarang diperingati sebagai Hari Pahlawan.

Dewasa ini semakin banyak wisatawan yang berkunjung ke lokasi-lokasi bersejarah di Surabaya. Para wisatawan biasanya mencari informasi melalui *internet* atau orang yang sudah pernah berkunjung ke Surabaya sebelum melakukan kunjungan. Informasi yang dicari mulai dari lokasi objek bersejarah, rute serta cerita sejarah menyangkut situs tersebut. Seiring dengan pesatnya perkembangan teknologi, informasi di *internet* dapat diakses dimana saja baik menggunakan *personal computer* maupun *mobile device*. Sehingga informasi dapat dengan mudah untuk didapatkan, terlebih lagi dengan menggunakan *mobile device* yang seperti telpon genggam.

Berdasarkan permasalahan di atas, untuk membantu para wisatawan dalam menemukan lokasi serta informasi dari objek bersejarah di kota Surabaya, maka di buatlah sebuah aplikasi wisata sejarah pada *mobile device* berbasis. Aplikasi ini dapat memberikan informasi tentang objek bersejarah tersebut. Selain itu, terdapat fitur simulasi wisata sejarah dan *virtual stamp* sebagai tanda bahwa *user* pernah mengunjungi sebuah lokasi bersejarah tertentu serta *mini-game* dan konten yang dapat di-*download* oleh *user*.

2. DASAR TEORI

2.1 Pertempuran Surabaya

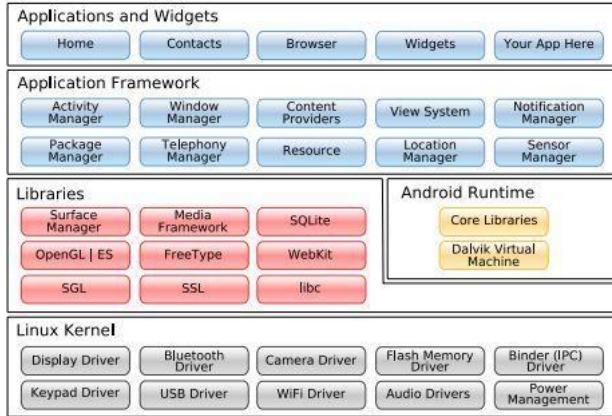
Pertempuran Surabaya merupakan perang yang terjadi di Surabaya pada tanggal 10 November 1945 antara pemuda dan tentara Indonesia dengan tentara Inggris yang dipicu oleh kematian salah Brigjen A.W.S. Mallaby dalam sebuah insiden tanggal 30 Oktober 1945 dalam gencatan senjata antara Indonesia dan Inggris. Pertempuran Surabaya berakhir dengan dikuasainya Surabaya oleh pasukan Inggris setelah 3 minggu perperangan. Peristiwa inilah yang memicu terjadinya perlawanan di berbagai daerah dan diperingati sebagai hari pahlawan setiap tanggal 10 November [1].

2.2 Pengertian Android

Android adalah sebuah sistem operasi untuk perangkat *mobile* berbasis Linux yang mencakup sistem operasi, *middleware* dan aplikasi. Android menyediakan *platform* terbuka bagi para pengembang untuk membuat aplikasi mereka. Awalnya, Google

Inc. membeli Android Inc. yang merupakan pendatang baru yang membuat *software* untuk ponsel/*smartphone*. Kemudian untuk mengembangkan Android, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan *hardware*, *software*, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia [2].

Arsitektur Android terdiri dari *Applications*, *Application Framework*, *Libraries*, *Android Runtime* dan *Kernel*. Arsitektur lengkap Linux dapat dilihat pada Gambar 1.



Gambar 1. Arsitektur Android

2.3 The Dalvik Virtual Machine (DVM)

Salah satu elemen kunci dari Android adalah *Dalvik Virtual Machine* (DVM). Android berjalan di dalam DVM bukan di *Java Virtual Machine* (JVM). DVM adalah “*register based*” sementara JVM adalah “*stack based*”, DVM. *Dalvik Virtual Machine* menggunakan *kernel* Linux untuk menangani fungsionalitas tingkat rendah termasuk keamanan, *threading*, dan proses serta manajemen memori. Semua *hardware* yang berbasis Android dijalankan dengan menggunakan *Virtual Machine* untuk menjalankan aplikasi [2].

2.4 Android SDK (Software Development Kit)

Android SDK adalah *tools API* (*Application Programming Interface*) yang diperlukan untuk mulai mengembangkan aplikasi pada *platform* Android menggunakan bahasa pemrograman Java [2].

2.5 Android Development Tools (ADT)

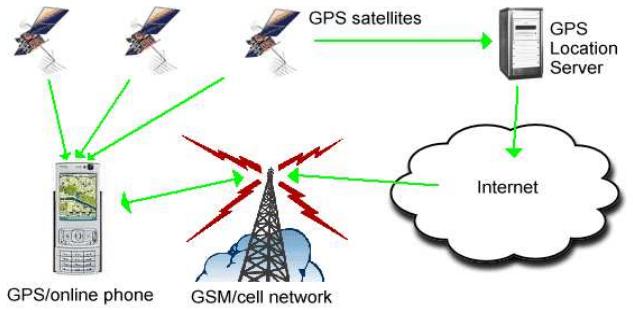
Android Development Tools (ADT) adalah *plugin* yang didesain untuk IDE Eclipse untuk memberikan kemudahan dalam mengembangkan aplikasi Android dengan menggunakan IDE Eclipse. Dengan menggunakan ADT akan lebih mudah dalam membuat aplikasi *project* Android, membuat GUI sebuah aplikasi, dan menambahkan berbagai komponen lainnya. Selain itu, dengan ADT, *user* dapat membuat *package* Android (.apk) yang digunakan untuk mendistribusikan aplikasi Android yang telah dibuat [2].

2.6 Global Positioning Service (GPS)

Global Positioning System adalah sistem untuk menentukan posisi di permukaan bumi dengan bantuan sinkronisasi sinyal satelit. Sistem ini menggunakan 24 satelit yang

mengirimkan sinyal gelombang mikro ke Bumi. Sinyal ini diterima oleh alat penerima di permukaan.

GPS adalah sistem radio navigasi dan penentuan posisi dengan menggunakan satelit navigasi yang dimiliki dan dikelola oleh Departemen Pertahanan Amerika Serikat. Nama formalnya adalah NAVSTAR GPS (*Navigation Satellite Timing and Ranging Global Positioning System*). GPS terdiri atas tiga segmen utama, yaitu *space segment*, *control system segment*, dan *user segment*. Pembagian segmen pada GPS dapat dilihat pada Gambar 2 [3].



Gambar 2. Arsitektur GPS

2.7 SQLite

SQLite adalah sebuah *embedded engine* database SQL yang tidak seperti kebanyakan database SQL lainnya, dimana SQLite tidak memiliki proses *server* yang terpisah. SQLite melakukan *read* dan *write* secara langsung pada *disk file*. Sebuah database SQL yang lengkap dengan beberapa tabel, *index*, *trigger*, *view* yang hanya berbentuk *file* biasa pada *disk*. Format *file* database yang digunakan adalah cross-platform, dimana SQLite bebas menyalin database antara sistem 32-bit dan 64-bit atau antara arsitektur *big-endian* dan *little-endian* [4].

2.8 JSON (JavaScript Object Notation)

JSON (*JavaScript Object Notation*) merupakan format untuk pertukaran data seperti halnya XML. Penggunaan JSON mudah dipahami oleh manusia karena formatnya yang cukup sederhana. JSON dibuat berdasarkan bahasa pemrograman JavaScript, Standar ECMA-262 Edisi ke-3 Desember 1999. JSON tidak bergantung pada bahasa pemrograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh *programmer*, seperti C, C++, C#, Java, JavaScript, Perl, Python dan lain-lain. Oleh karena memiliki sifat-sifat tersebut, JSON ideal digunakan untuk melakukan pertukaran data [5].

3. DESAIN SISTEM

3.1 Perancangan Arsitektur Sistem

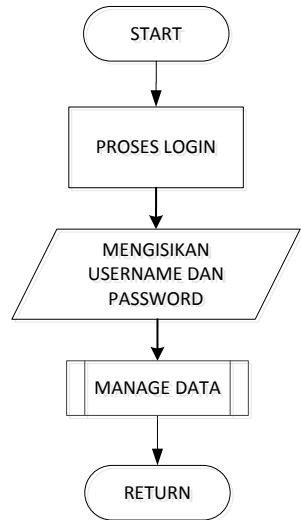
Pada proses sistem ini, Android dapat melakukan *request* kepada *network provider* untuk mendapatkan koordinat GPS sekarang. Android melakukan transaksi dengan server berupa *login*, *registrasi* dan *download* konten. Admin bertugas untuk melakukan *update* data konten yang terdapat pada *server*. Arsitektur sistem dapat dilihat pada Gambar 3.



Gambar 3. Perancangan Arsitektur Sistem

3.2 Flowchart Sistem Pada Web Admin

Sistem Web Admin dimulai dengan menu *login* yang digunakan untuk mengakses halaman *admin* dengan menginputkan *username* dan *password*. Setelah melakukan *login* *admin* dapat mengakses tabel-tabel pada database *web server*. Tabel-tabel tersebut adalah *tUser*, *tAdmin*, *tLocation*, *tStamps* dan *tUpdates*. Masing-masing tabel berada di halaman yang berbeda yang dapat diakses melalui tombol navigasi. *Admin* dapat melakukan *manage data* berupa *view*, *insert*, *update* dan *delete* data pada tabel-tabel tersebut.. Flowchart sistem pada *web admin* dapat dilihat pada Gambar 4.



Gambar 4. Flowchart Sistem Pada Web Admin

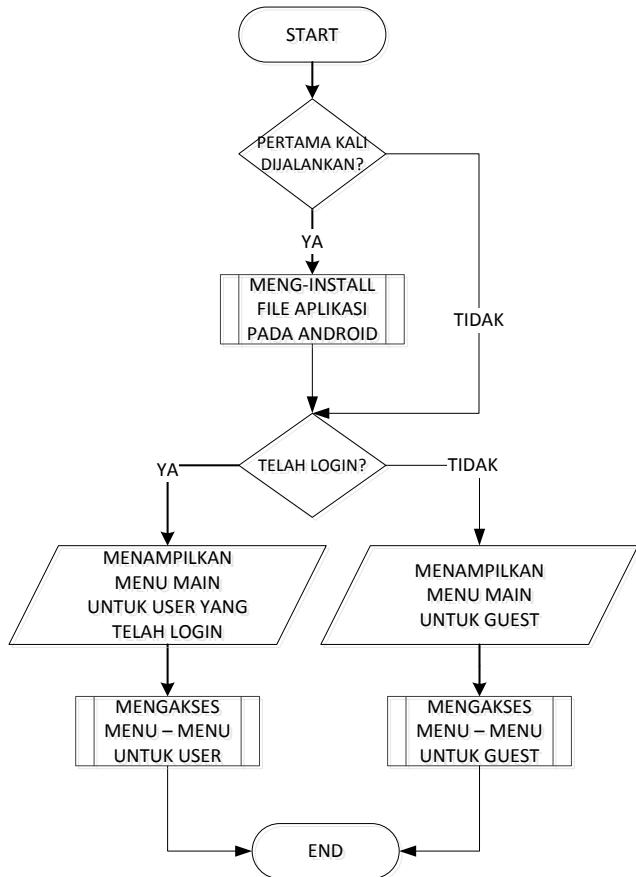
3.3 Flowchart Sistem Pada Android

Sistem dari Android dimulai dengan menampilkan menu *main* yang akan meng-*install* data aplikasi ketika aplikasi pertama kali dijalankan. Untuk *user guest* menu *main* dapat digunakan untuk mengakses beberapa menu lain, yaitu: menu lokasi, menu sejarah Surabaya, menu simulasi sejarah, menu *download* konten dan menu *login* dan daftar.

Pada menu lokasi, *user* dapat melihat informasi, foto, memainkan *game* dan mendapatkan arahan menuju lokasi sejarah. Pada menu sejarah Surabaya *user* dapat melihat sejarah dari kota Surabaya. Menu simulasi sejarah digunakan untuk menjalankan

simulasi sejarah pertempuran Surabaya. Menu *download* konten digunakan men-*download* konten baru dari *server*.

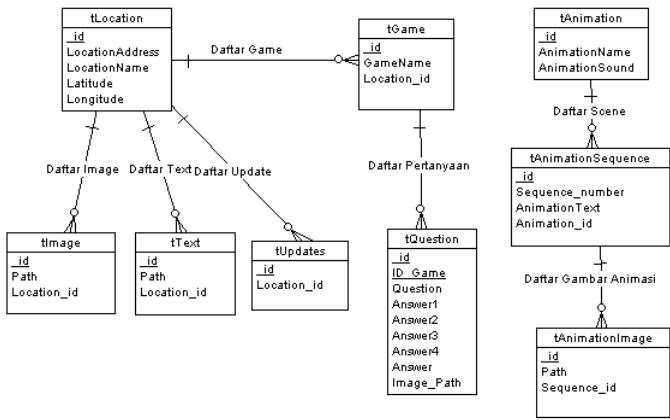
Untuk *user* yang telah melakukan *login* menu *main* akan menampilkan menu tambahan dari menu yang ditampilkan untuk *guest*, yaitu: menu tempat yang telah dikunjungi dan menu *logout*. Setelah melakukan *login* menu lokasi dapat digunakan untuk melakukan *virtual stamping* untuk menyimpan kunjungan *user* pada lokasi berasal dan men-*download* konten khusus dari lokasi tersebut. Flowchart sistem pada Android dapat dilihat pada Gambar 5.



Gambar 5. Flowchart Sistem Pada Android

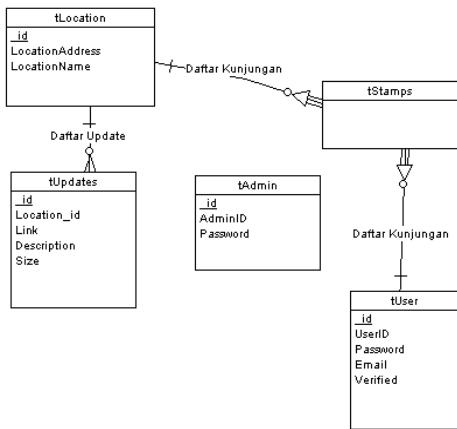
3.4 Desain Struktur Database

Entity Relation Diagram adalah diagram yang digunakan untuk menggambarkan tabel-tabel yang dibutuhkan oleh sistem beserta relasi antara tabel yang terbentuk antar tabel. Desain struktur *database* yang digunakan pada Android dapat dilihat pada Gambar 6.



Gambar 6. Entity-Relationship Diagram Android

Desain struktur database yang digunakan pada *Web Server* dapat dilihat pada Gambar 7.



Gambar 7. Entity-Relationship Diagram Web Server

4. IMPLEMENTASI SISTEM

4.1 Implementasi Pada Server

Implementasi pada *server* berperan sebagai perantara komunikasi antara Android dengan *database* pada *server*. Seluruh proses *view*, *insert*, *update*, dan *delete* terhadap *database* dijalankan dalam bentuk fungsi pada PHP. Komunikasi antara *server* dengan Android dilakukan dengan menggunakan metode HTTP GET yang mengembalikan nilai dalam format JSON.

4.2 Implementasi Pada Android

Implementasi dari sistem Android yang dijelaskan pada bagian ini adalah bagian *download* konten. *Download* konten dimulai dengan men-request daftar konten yang tersedia dengan menjalankan fungsi *CheckUpdate* yang membutuhkan parameter ID lokasi. Fungsi *CheckUpdate* mengembalikan variabel

menggunakan format JSON, yang kemudian *di-parse* dan dibandingkan dengan data konten yang telah tersimpan di dalam *device*. Untuk menampilkan daftar konten dipanggil fungsi *loadlistkonten*. Dari daftar yang diperoleh dari fungsi *loadlistkonten*, *user* dapat memilih konten yang akan *download*. Fungsi untuk men-*load* daftar konten dapat dilihat pada Pseudocode 1.

Pseudocode 1. Fungsi untuk men-*load* daftar konten.

```

    Return GET ALL FROM TABLE T_UPDATES THAT
    location_id IS EQUAL user_locationid.
    AND NOT EXIST IN device's database.
  
```

Ketika user memilih sebuah konten untuk *di-download*, fungsi *DownloadFile* akan dipanggil. Fungsi ini memerlukan parameter berupa *link download* dari data konten yang dipilih untuk melakukan *download*.

Pseudocode 2. Fungsi untuk men-*download* konten.

```

DownloadLink = Selected Konten Download Link;
Data = GET File From DownloadLink
FileLength = GET ContentLength;
While (Count < File Length)
    Write Data to DownloadPath;
  
```

Setelah file berhasil *di-download* pada fungsi *DownloadFile*, maka fungsi *update* untuk menyalin data konten ke *device* dan mengeksekusikan file SQL yang terdapat pada konten yang *di-download* untuk meng-*install* konten tersebut. Fungsi untuk men-*install* konten dapat dilihat pada Pseudocode 3.

Pseudocode 3. Fungsi untuk men-*install* konten.

```

Extract File (File Path, Destination Path);
Execute SQL (SQL Path);
DELETE FILE;
  
```

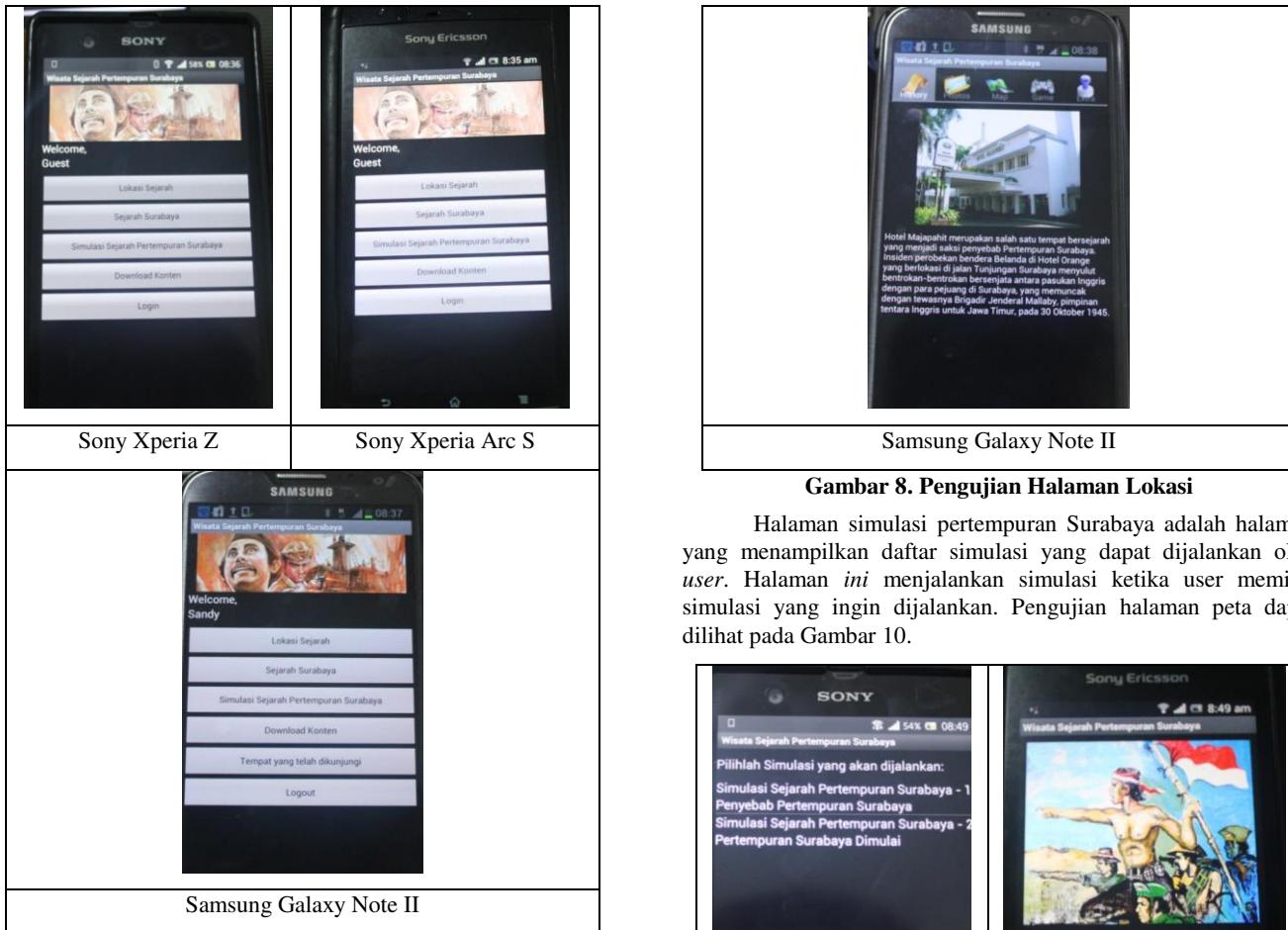
5. PENGUJIAN SISTEM

Pengujian ini dilakukan dengan menggunakan *smartphone* Sony Xperia Z, Sony Xperia Arc S dan Samsung Galaxy Note II untuk melakukan pengujian aplikasi.

5.1 Pengujian Pada Android

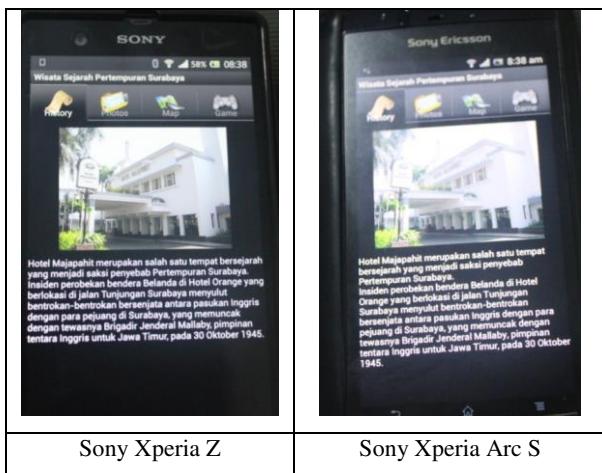
Proses pengujian ini dilakukan pada *device* Android untuk melihat hasil yang telah didesain. Pengujian yang dilakukan pada Android adalah sebagai berikut.

Halaman *main* menampilkan lima menu utama yaitu lokasi sejarah, sejarah Surabaya, simulasi sejarah pertempuran Surabaya, *download* konten dan *login*. Setelah melakukan *login*, menu *login* akan dihilangkan dan halaman *main* akan menampilkan dua menu tambahan yaitu menu *logout* dan menu tempat yang telah dikunjungi.. Pengujian halaman *main* dapat dilihat pada Gambar 7.



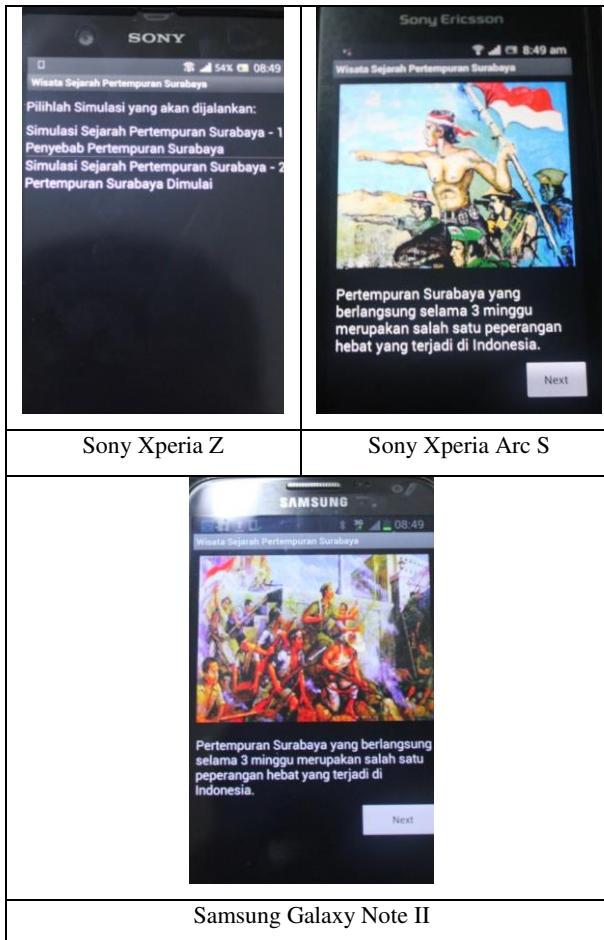
Gambar 7. Pengujian Halaman Main

Halaman sejarah lokasi adalah halaman pertama yang ditampilkan ketika *user* menekan tombol masuk pada halaman lokasi. Halaman lokasi menampilkan gambar serta sejarah menyangkut pertempuran Surabaya dari suatu lokasi.. Pengujian halaman lokasi dapat dilihat pada Gambar 8.



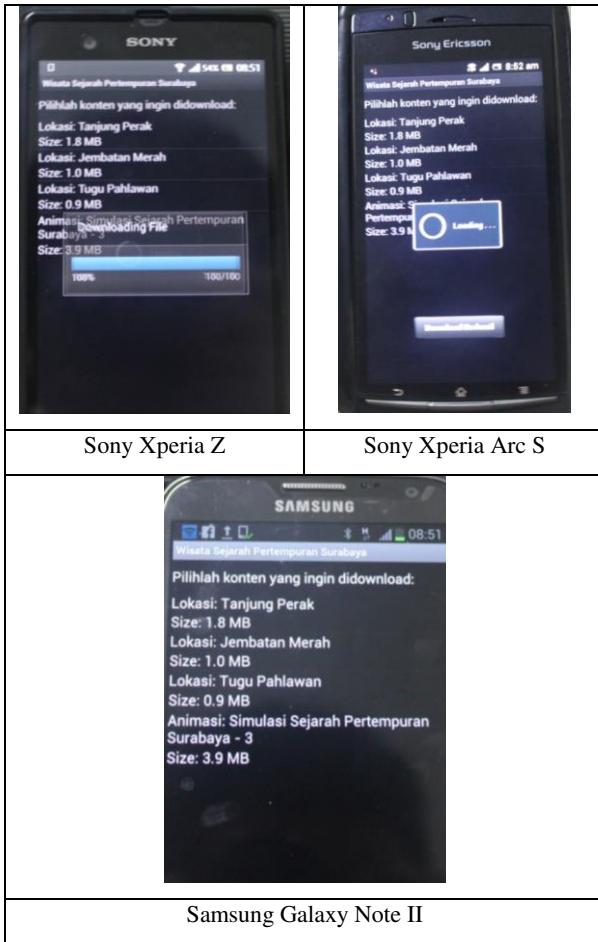
Gambar 8. Pengujian Halaman Lokasi

Halaman simulasi pertempuran Surabaya adalah halaman yang menampilkan daftar simulasi yang dapat dijalankan oleh *user*. Halaman ini menjalankan simulasi ketika user memilih simulasi yang ingin dijalankan. Pengujian halaman peta dapat dilihat pada Gambar 10.



Gambar 9. Pengujian Halaman Simulasi Sejarah

Halaman *download* konten adalah halaman yang menampilkan daftar konten yang dapat di-*download* oleh *user*. Halaman *download* konten menjalankan fungsi *download* ketika *user* memilih konten yang ingin di-*download*. Pengujian halaman *download* konten dapat dilihat pada Gambar 11.



Gambar 10. Pengujian Halaman *Download* Konten

6. KESIMPULAN

Dari proses perancangan sistem hingga pengujian aplikasi yang telah dilakukan, dapat ditarik kesimpulan yaitu sebagai berikut.

- Berdasarkan hasil pengujian, aplikasi yang dibuat telah dapat melakukan fungsi *login*, *logout*, melihat informasi lokasi bersejarah, menampilkan galeri, peta dengan arahan menuju lokasi berdasarkan posisi *user*, menjalankan *game*, menjalankan fungsi *virtual stamping*, men-*download* konten serta mengakses data konten tersebut tanpa men-*download* file tersebut kembali, menjalankan simulasi pertempuran Surabaya serta melihat lokasi yang telah dikunjungi *user*.
- Kecepatan *internet* yang buruk mempengaruhi kinerja dari aplikasi, seperti men-*download* konten.
- Lokasi dari *user* serta cuaca yang buruk mempengaruhi keakuratan dari koordinat GPS yang diperoleh.

7. REFERENSI

- [1] Ariwibowo, Budi. (2009). Pertempuran Surabaya 10 November 1945. Retrieved June 19 2013 from <http://umum.kompasiana.com/2009/11/23/pertempuran-surabaya-10-november-1945-28773.html>
- [2] Safaat, Nazruddin. (2012). Pemograman Aplikasi *Mobile Smartphone* dan *Tablet PC* Berbasis Android. Bandung: Penerbit Informatika.
- [3] Singhal, Manav; Shukla Anupam. (2012). *Implementation of Location Based Services in Android Using GPS and Web Services*. Retrieved June 20, 2013 from <http://ijcsi.org/papers/IJCSI-9-1-2-237-242.pdf>
- [4] SQLite. (2013). *About SQLite*. Retrieved June 19 2013, from <http://www.sqlite.org/about.html>.
- [5] JSON. (2013). *Introducing JSON*. Retrieved June 19, 2013 from <http://www.json.org/>