

APLIKASI PENUNJUK ARAH LOKASI KAMPUS STIMATA MENGUNAKAN TEKNOLOGI AUGMENTED REALITY

Andika Satya Pratama, Khoerul Anwar

Program Studi Teknik Informatika
STMIK PPKIA Pradnya Paramita Malang
Jl. Laksda Adi Sucipto No. 249-A Malang
Email : macho9499@yahoo.com

ABSTRACT

STMIK Pradnya Paramitha (STIMATA) is one of the private universities in the city of Malang. In mid-2008 , STIMATA campus occupies a new building that is on the road Admiral Adi Sucipto . Quite a lot of people who do not know exactly where the location of the new campus Stimata , especially those residing outside the city of Malang . One reason is the location of the campus is located quite far from the center of the city of Malang.

Of these problems , solutions that can be applied is to develop an application on a mobile device that can be a signpost on Stimata campus locations . The application can be put on the official website Stimata so it can be downloaded by anyone who wants to know the location of the campus Stimata easily . Manufacturing applications such directions would be more attractive and easier to use if built using Augmented Reality technology.

This study aims to build and develop applications on mobile devices by utilizing Augmented Reality technology that can be used as a pointing device for its direction . Applications will be implemented on a device built Android smartphone operating system . In operation this application uses GPS coordinates to two points , the coordinates of the location coordinates of the user and Stimata campus , while the implementation of Augmented Reality technology on the application will be used Mixare engine . This study starts from the design workflow applications , the main display interface design , core application development , to the creation of the menu interface provides useful features that are provided to the user.

Key words : *mobile applications, directions, smartphones, Android, Augmented Reality, Mixare .*

PENDAHULUAN

Dewasa ini, banyak sekali bermunculan teknologi-teknologi baru di bidang Teknologi Informasi. Teknologi-teknologi tersebut terus berkembang dengan sangat pesat, sehingga mampu memperlihatkan sesuatu yang awalnya seperti tidak mungkin, menjadi mungkin dan menjadi kenyataan. Begitu juga dengan teknologi yang saat ini sedang berkembang, yaitu *Augmented Reality* (AR) atau yang dalam bahasa Indonesia kita sebut dengan Realitas Tertambah.

Augmented Reality (selanjutnya akan penulis singkat dengan AR) merupakan cabang ilmu komputer yang mempelajari mengenai penggabungan data di dunia nyata, dengan data yang dihasilkan oleh komputer. Biasanya data yang disajikan adalah penggabungan data grafis (Contoh : Rekaman video, foto, dll.) yang berasal dari dunia nyata dengan grafis yang dihasilkan oleh komputer (Contoh : Animasi gambar, animasi text, dll.).

Penerapan dari teknologi AR cukup luas, dimulai dari sebagai penyedia informasi mengenai suatu lokasi, hingga dipergunakan

dalam sebuah permainan dalam *Reality Game*. Perangkat elektronik yang dapat digunakan untuk penerapan teknologi AR juga sangat beragam, selain pada *Personal Computer*, teknologi AR juga dapat diterapkan pada perangkat *mobile* seperti *smartphone* dan perangkat Game Konsol seperti PlayStation 3.

STMIK Pradnya Paramitha (STIMATA) adalah salah satu perguruan tinggi swasta yang berada di kota Malang yang dalam pengajarannya terfokus pada bidang informatika dan komputer.

Pada pertengahan tahun 2008, kampus STIMATA menempati gedung baru yang berada pada jalan Laksda Adi Sucipto. Cukup banyak orang yang belum mengetahui dengan pasti dimana lokasi kampus Stimata yang baru tersebut, terutama yang bertempat tinggal di luar kota Malang. Hal tersebut kemungkinan disebabkan oleh lokasi kampus yang berada cukup jauh dari pusat kota Malang. Oleh karena itu, banyak calon mahasiswa Stimata yang ingin mendaftar kebingungan saat mencari di mana lokasi kampus Stimata berada.

Dari permasalahan di atas, penulis mempunyai gagasan yang dapat memberikan solusi terkait penyediaan informasi mengenai di mana letak kampus Stimata berada, yaitu dengan menciptakan aplikasi *mobile* sebagai penunjuk arah yang dilengkapi dengan teknologi *Augmented Reality* (AR). Dengan demikian aplikasi ini akan memberikan informasi mengenai letak kampus Stimata kepada pengguna dengan mudah, cepat dan akurat.

Gadget yang penulis pilih dalam pembuatan aplikasi ini adalah *smartphone* yang menggunakan sistem operasi Android. Sistem operasi Android dipilih penulis dikarenakan memiliki beberapa keunggulan dibandingkan dengan sistem operasi yang lain, diantaranya: (1) Sistem operasi Android bersifat *open source* sehingga lebih *up-to-date*. Setiap ada “bug” yang baru ditemukan, para pengembang akan menyediakan “patch” untuk memperbaikinya. Berbeda dengan sistem operasi lain yang *non open source*, para pengguna harus menunggu rilis terbaru untuk mendapatkan update terbaru, (2) *Multi tasking* yang lebih handal daripada sistem operasi yang lain. Sistem operasi Android terbukti lebih handal dalam hal *multi tasking*, terlebih lagi dengan adanya aplikasi *Task Manager* yang mempermudah pengaturan aplikasi yang sedang aktif. (3) Android lebih stabil. Linux yang menjadi dasar dari Android sudah terbukti menjadi Sistem operasi yang banyak digunakan sebagai sistem operasi untuk server karena kestabilannya. (4) Karena sifatnya yang *open source*, banyak library-library yang disediakan oleh pengembang, sehingga memudahkan para developer yang ingin membuat aplikasi berbasis Android. Berbeda dengan sistem operasi yang lain dimana library yang disediakan sangat terbatas baik jumlah maupun fungsinya.

Berdasarkan dari uraian latar belakang diatas, fokus dari penelitian ini adalah merancang dan membangun aplikasi *mobile* yang berbasis Android yang berfungsi sebagai penunjuk arah yang tepat untuk menuju ke lokasi kampus Stimata dengan memanfaatkan

teknologi *Augmented Reality*. Oleh karena itu, penulis menetapkan judul dari penelitian ini, yaitu: “**APLIKASI PENUNJUK ARAH LOKASI KAMPUS STIMATA MENGGUNAKAN TEKNOLOGI AUGMENTED REALITY**”.

Rumusan Masalah

Berdasarkan dari uraian Latar Belakang, dapat dirumuskan beberapa rumusan masalah seperti berikut:

1. Bagaimana merancang dan membangun aplikasi Penunjuk Arah Lokasi Kampus Stimata pada *smartphone* Android?
2. Bagaimana mengimplementasikan teknologi *Augmented Reality* pada aplikasi Penunjuk Arah Kampus Stimata pada *smartphone* Android?

Tujuan

Tujuan dari penelitian yang akan dilakukan adalah untuk menciptakan sebuah aplikasi berteknologi *Augmented Reality* yang dapat memberikan informasi mengenai lokasi kampus Stimata kepada pengguna dengan cepat, tepat dan mudah.

Manfaat

Sedangkan manfaat yang bisa di dapat adalah dapat memudahkan calon mahasiswa yang belum mengetahui letak kampus Stimata untuk menemukan lokasi kampus Stimata, terutama bagi calon mahasiswa yang berasal dari luar kota.

TINJAUAN PUSTAKA

Pengertian Informasi

Secara umum informasi dapat didefinisikan sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian yang nyata yang digunakan untuk pengambilan keputusan. Informasi merupakan data yang telah diklasifikasikan atau diolah atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan.

AR (*Augmented Reality*)

Augmented Reality (AR) adalah penggabungan benda-benda nyata dan maya di lingkungan nyata, berjalan secara interaktif dalam waktu nyata, dan terdapat integrasi antarbenda dalam tiga dimensi, yaitu benda maya terintegrasi dalam dunia nyata. Penggabungan benda nyata dan maya dimungkinkan dengan teknologi tampilan yang sesuai, interaktivitas dimungkinkan melalui perangkat-perangkat input tertentu, dan integrasi yang baik memerlukan penjejak yang efektif. Selain menambahkan benda maya dalam lingkungan nyata, realitas ditambah juga berpotensi menghilangkan benda-benda yang sudah ada. Menambah sebuah lapisan gambar maya dimungkinkan untuk menghilangkan atau menyembunyikan lingkungan nyata dari pandangan pengguna. Misalnya, untuk menyembunyikan sebuah meja dalam lingkungan nyata, perlu digambarkan lapisan representasi tembok dan lantai kosong yang diletakkan di atas gambar

meja nyata, sehingga menutupi meja nyata dari pandangan pengguna.

AR dibedakan menjadi empat bagian menurut media yang digunakan yaitu *simple AR*, *MarkerBased AR*, *Markerless AR* dan *Augmented Vision*. *Markerless AR* umumnya diterapkan dalam *mobile device* seperti *smartphone*. Sesuai dengan namanya *markerless AR* tidak membutuhkan marker yang terlihat secara fisik untuk mengetahui posisi suatu objek. Sebagai gantinya, digunakan informasi dari GPS atau kompas dan cara ini dikenal dengan nama *geotagging* dan *geolocation*. Melalui *geotagging* dan *geolocation* yang bisa kita sebut sebagai marker yang tidak terlihat inilah konten seperti tulisan, video, maupun audio kemudian ditampilkan dilayar *mobile device* tersebut.

Salah satu metode AR yang saat ini sedang berkembang adalah metode “*Markerless Augmented Reality*“, dengan metode ini pengguna tidak perlu lagi menggunakan sebuah marker untuk menampilkan elemen-elemen digital. Seperti yang saat ini dikembangkan oleh perusahaan *Augmented Reality* terbesar di dunia Total Immersion, mereka telah membuat berbagai macam teknik *Markerless Tracking* sebagai teknologi andalan mereka, seperti *Face Tracking*, *3D Object Tracking*, *Motion Tracking*, dan *GPS Based Tracking*. Dalam skripsi ini, teknologi yang digunakan adalah *GPS Based Tracking*.

Teknik *GPS Based Tracking* saat ini mulai populer dan banyak dikembangkan pada aplikasi *smartphone* (iPhone dan Android). Dengan memanfaatkan fitur GPS dan Kompas

yang ada didalam *smartphone*, aplikasi akan mengambil data dari GPS dan Kompas kemudian menampilkannya dalam bentuk arah yang kita inginkan secara realitme, bahkan ada beberapa aplikasi menampilkannya dalam bentuk 3D. Salah satu pelopor *GPS Based Tracking* adalah aplikasi yang bernama *Layar*.



Gambar 1 Aplikasi *Markerless AR* pada *smartphone*

Sistem Operasi Android

Pengenalan Android

Android adalah *software* untuk perangkat *mobile* yang mencakup sistem operasi, *middleware* dan aplikasi kunci. SDK (*Software Development Kit*) Android menyediakan alat dan API (*Application Programming Interface*) yang diperlukan untuk mengembangkan aplikasi pada platform Android dengan menggunakan bahasa pemrograman Java (Stephanus Hermawan, 2011). Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam *mobile devices*.

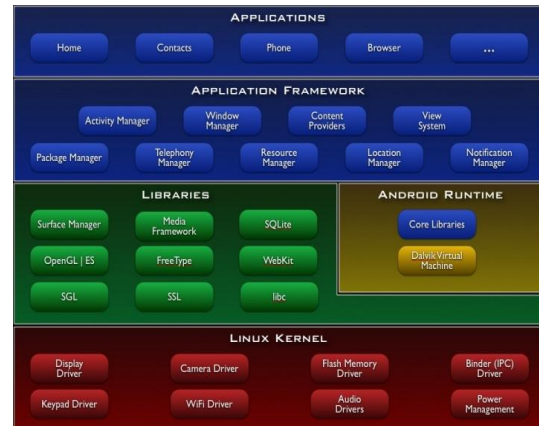
Fitur Android

Android juga memiliki beberapa fitur sebagai berikut (Stephanus Hermawan, 2011)

1. **Application framework** memungkinkan penggunaan kembali dan penggantian komponen
2. **Dalvik virtual machine** optimalisasi untuk perangkat bergerak
3. **Integrated browser** berdasarkan open source WebKit mesin
4. **Optimized graphics** didukung *library* grafis 2D ; grafis 3D berbasis OpenGL ES 1.0
5. **SQLite** untuk penyimpanan data terstruktur
6. **Media support** untuk audio, video, dan gambar masih dalam format (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF)
7. **GSM Telephony**
8. **Bluetooth, EDGE, 3G, and WiFi**
9. **Camera, GPS, compass, and accelerometer**
10. **Rich development environment** termasuk *device emulator*, alat untuk *debugging*, *memory* dan *performance profiling*, dan *plugin* untuk Eclipse IDE

Arsitektur Android

Diagram Arsitektur Android ditunjukkan pada gambar 2.



Gambar 2 Arsitektur Android

Sesuai ilustrasi yang ditunjukkan pada gambar 2. Arsitektur Android dapat dibagi menjadi lima bagian besar, yaitu:

1. Applications

Android memuat seperangkat aplikasi inti mencakup *email client*, program SMS, kalender, peta, *browser*, kontak, dan lain lain. Semua aplikasi ditulis menggunakan bahasa pemrograman Java.

2. Application Framework

Android menawarkan kemampuan untuk membangun aplikasi yang sangat kaya dan inovatif. Pengembang bebas untuk mengakses perangkat keras, informasi akses lokasi, menjalankan *background services*, mengatur alarm, tambahkan pemberitahuan ke *statusbar*, dan banyak lagi.

Pengembang memiliki akses penuh ke *framework API* yang sama yang digunakan oleh aplikasi inti. Arsitektur aplikasi dirancang untuk menyederhanakan penggunaan kembali komponen, aplikasi apapun dapat mempublikasikan kemampuannya dan aplikasi lain kemudian dapat menggunakan kemampuan mereka (diatur oleh batasan keamanan yang diberlakukan oleh

framework). Mekanisme ini mengizinkan *user* mengganti komponen.

Berikut adalah contoh servis dan system yang mendasari seluruh aplikasi Android:

- ❖ *View* : tampilan yang dapat digunakan untuk membangun aplikasi, termasuk didalamnya *list*, *grid*, kotak teks, tombol, dan sebagainya
- ❖ *Content Providers* : memungkinkan aplikasi untuk mengakses data dari aplikasi lain (misal Kontak), atau untuk membagi data yang dimiliki
- ❖ *Resource Manager* : menyediakan akses ke *non-code resources* seperti grafik dan *layout* data
- ❖ *Notification Manager* : memungkinkan aplikasi untuk menampilkan peringatan pada *status bar*
- ❖ *Activity Manager* : mengatur *lifecycle* aplikasi

3. Libraries

Libraries adalah *layer* dimana fitur-fitur Android berada, biasanya para pembuat aplikasi Android mengakses *libraries* untuk menjalankan aplikasinya. Android mencakup set *library* C / C++ yang digunakan oleh berbagai komponen dari sistem Android. Beberapa *library* tercantum seperti dibawah ini:

- ❖ Sistem C library : *library* standard C (*libc*)
- ❖ Media Libraries : *library* untuk mendukung memutar dan merekam berbagai format *audio* dan *video*, juga file gambar

- ❖ Surface Manager : mengatur akses ke tampilan *subsistem* dan memadukan grafik 2D dan 3D pada beberapa aplikasi
- ❖ LibWeb Core : *web browser engine* yang mendukung Android *browser*
- ❖ SGL : mendasari *engine* grafik 2D
- ❖ 3D libraries : implementasi OpenGL ES 1.0 APIs berfungsi mengoptimalkan fungsionalitas 3D
- ❖ FreeType : *rendering bitmap* dan *font*
- ❖ SQLite : *engine* basis data yang ampuh dan ringan yang tersedia untuk semua aplikasi

4. Android Runtime

Android mencakup seperangkat *library* inti yang menyediakan sebagian besar fungsi yang tersedia di *library* inti dari bahasa pemrograman Java. Setiap aplikasi Android berjalan dalam prosesnya sendiri, dengan Dalvik *Virtual Machine*-nya (VM) sendiri. Dalvik VM mengeksekusi *file* dalam format Dalvik *executable* (.dex) yang dioptimalkan untuk meminimalisir jejak memori.

5. Linux Kernel

Android bergantung pada Linux versi 2.6 untuk layanan sistem inti seperti keamanan, manajemen memori, manajemen proses, jaringan *stack*, dan *driver model*.

Fundamental Aplikasi

Aplikasi Android ditulis dalam bahasa pemrograman java. Kode java dikompilasi bersama dengan data file resource yang dibutuhkan oleh aplikasi, dimana proses di-package oleh tools yang dinamakan “apt tools”

ke dalam paket Android sehingga menghasilkan file dengan ekstensi apk. File apk itulah yang kita sebut dengan aplikasi, dan nantinya dapat diinstall diperangkat mobile.

Ada enam jenis komponen pada aplikasi Android, yaitu (Nazruddin H. Safaat, 2011) :

1. *Activities*

Suatu *activity* akan menyajikan *user interface* (UI) pada pengguna, sehingga pengguna dapat melakukan interaksi. Sebuah aplikasi android bisa jadi hanya memiliki satu *activity*, tetapi umumnya aplikasi memiliki banyak *activity* tergantung pada tujuan aplikasi dan desain dari aplikasi tersebut. Satu *activity* biasanya akan dipakai untuk menampilkan aplikasi atau yang bertindak sebagai *user interface* (UI) saat aplikasi diperlihatkan kepada *user*. Untuk pindah dari satu *activity* ke *activity* lain kita dapat melakukannya dengan satu *event*, misalnya klik tombol, memilih opsi atau menggunakan *triggers* tertentu. Secara hirarki sebuah windows *activity* dinyatakan dengan method *Activity setContentView()*.

ContentView adalah objek yang berada pada *root* hirarki.

2. *Services*

Service tidak memiliki *Graphic User Interface* (GUI), tetapi *service* berjalan secara *background*, sebagai contoh dalam memainkan musik, *service* mungkin memainkan

musik atau mengambil data jaringan, tetapi *service* harus berada dalam kelas induknya. Misalnya, media player sedang memutar lagu dari *list* yang ada, aplikasi ini akan memiliki dua atau lebih *activity* yang memungkinkan *user* untuk memilih lagu misalnya, atau *sms* sambil *player* sedang jalan.

3. *Broadcast Receiver*

Broadcast receiver berfungsi menerima dan bereaksi untuk menyampaikan notifikasi. Contoh *broadcast* seperti notifikasi zona waktu berubah, baterai *low*, gambar telah selesai diambil oleh kamera, atau mengubah referensi bahasa yang digunakan. *Broadcast receiver* tidak memiliki *user interface* (UI), tetapi memiliki *activity* untuk merespon informasi yang mereka terima.

4. *Content Provider*

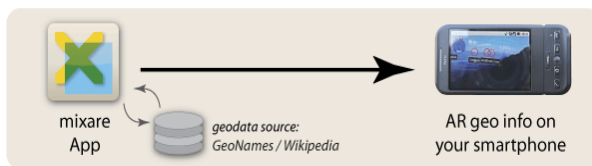
Content provider membuat kumpulan aplikasi data secara spesifik sehingga bisa digunakan oleh aplikasi lain. Data disimpan dalam *file* sistem seperti *SQLite*. *Content provider* menyediakan cara untuk mengakses data yang dibutuhkan oleh suatu *activity*, misalnya ketika kita menggunakan aplikasi yang membutuhkan peta (*Map*), atau aplikasi yang membutuhkan untuk mengakses data kontak dan navigasi, maka disinilah fungsi *content provider*.

Mixare

Mixare (Mix Augmented Reality Engine) adalah open source augmented reality browser yang diterbitkan dibawah GPLv3. Mixare tersedia untuk Android dan iPhone 3GS dan diatasnya. Mixare bekerja sebagai aplikasi yang sangat otonom dan tersedia untuk pengembangan dari implementasi sendiri.

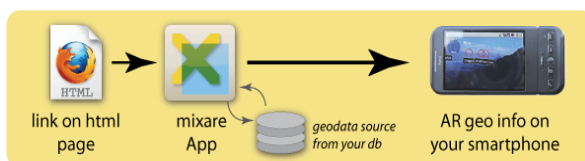
Ada beberapa cara dalam menggunakan mixare.

a) Mixare adalah sebuah aplikasi otonom yang menampilkan POI Wikipedia terhadap lingkungannya. Mixare dijalankan di perangkat *mobile* secara langsung.



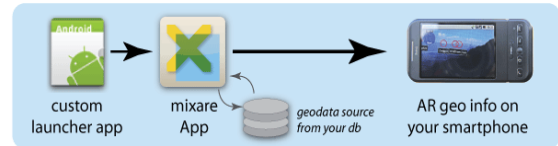
Gambar 3 Penggunaan Mixare I

b) Mixare dapat diakses oleh link pada situs HTML, dimana sumber data ditransfer ke aplikasi.



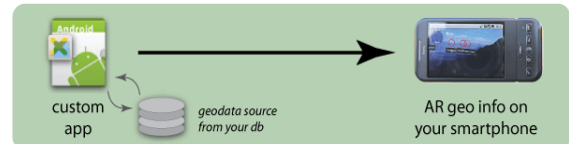
Gambar 4 Penggunaan Mixare II

c) Mixare dapat diakses oleh launcher-app kita sendiri, data source ditransfer ke aplikasi yang kita buat.



Gambar 5 Penggunaan Mixare III

d) Mixare secara bebas di upgrade dan bahkan dapat dimodifikasi ke aplikasi individu.



Gambar 6 Penggunaan Mixare IV

ANALISIS DAN PEMODELAN

Metode Penelitian

Langkah-langkah yang digunakan dalam penelitian ini adalah sebagai berikut :

1. Studi literatur

Literatur yang digunakan adalah yang terkait dengan *Augmented Reality* (AR). Pembelajaran tersebut dilakukan dengan cara mencari literatur-literatur di perpustakaan, *browsing* di internet, serta bertanya secara langsung kepada ahli-ahli yang memiliki kompetensi di bidangnya.

2. Perancangan sistem

Pada tahap ini dilakukan perancangan algoritma dari program yang akan dibuat hingga diperoleh hasil program seperti yang diharapkan.

3 Pembuatan program

Dari algoritma pembuatan program yang telah dilakukan pada tahap perancangan program, maka dilakukan pembuatan program atau perangkat lunak.

4. Pengujian sistem

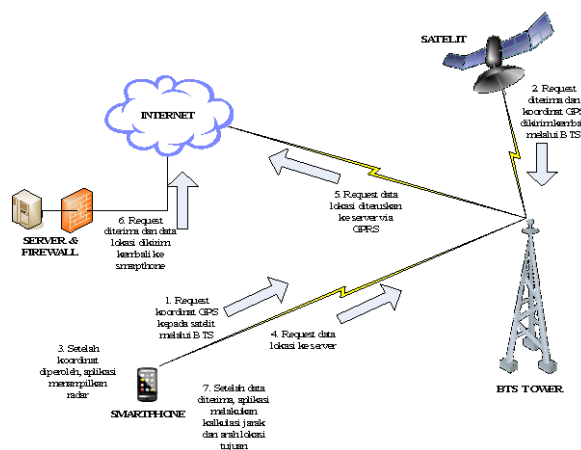
Setelah program selesai dibuat, maka langkah selanjutnya adalah dilakukan pengujian, apakah program telah sesuai dengan yang diharapkan. Pengujian program dilakukan di saat menjelang selesainya pembuatan program. Bila program belum memenuhi kriteria yang diharapkan, maka dilakukan perbaikan terhadap program.

5. Penyusunan laporan

Setelah program tersebut dibuat dan diuji, maka dilakukan pembuatan laporan penelitian.

Perancangan Sistem Aplikasi

Secara garis besar alur dari aplikasi yang telah dibangun adalah sebagai berikut:



Gambar 7 Alur Proses Aplikasi

Berikut adalah keterangan yang lebih detail tentang proses yang berlangsung :

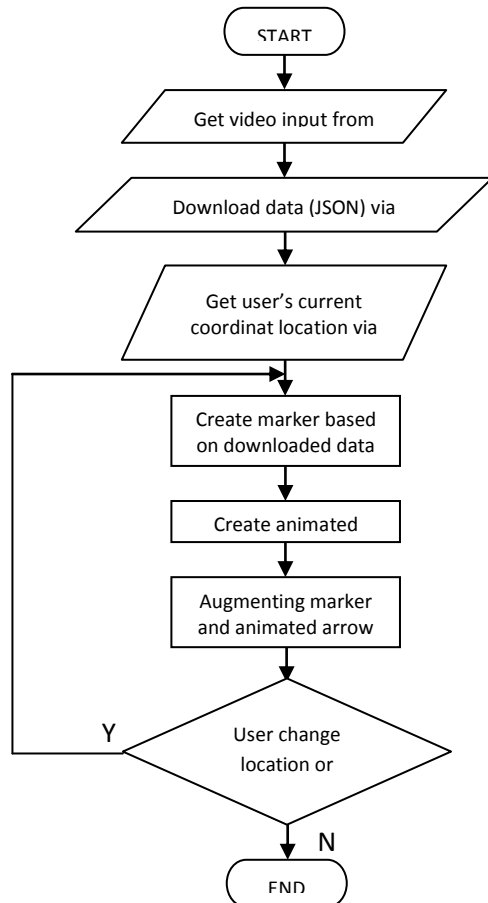
1. Smartphone *me-request* titik koordinat GPS pada tempat pengguna aplikasi berada kepada satelit dengan perantara BTS dengan harapan proses penitikan koordinat

dapat berjalan lebih cepat (menggunakan sistem AGPS).

2. BTS mengirimkan titik koordinat yang telah di dapat oleh satelit kepada *smartphone* dan menjadikannya sebagai titik pusat dari radius.
3. Aplikasi memproses data koordinat dan menentukan radius dari radar yang akan dibuat.
4. *Smartphone* *men-download* data informasi berupa titik koordinat beserta keterangan-keterangan dari lokasi kampus STIMATA melalui metode *HttpRequest* dari situs yang telah disediakan sebelumnya.
5. Setelah koordinat tujuan didapat, aplikasi akan memproses titik koordinat dan menampilkan titik tujuan pada radar yang telah dibuat, kemudian menjadikan titik koordinat tujuan tersebut sebagai *marker* yang akan dijadikan sebagai penunjuk arah.
6. Berdasarkan *marker* yang telah terbentuk, aplikasi akan *generate* / *me-render* tampilan *augmented reality* pada kamera *smartphone* sehingga dapat memudahkan penggunaanya untuk mencari lokasi kampus Stimata

Flowchart Cara Kerja Aplikasi

Secara garis besar, berikut merupakan gambaran alur kerja atau *flowchart* dari aplikasi yang dibuat:



Gambar 8 Flowchart Aplikasi

Perancangan Program Aplikasi

Pada aplikasi *Augmented Reality*, sebuah objek tiga dimensi akan di-render pada marker yang telah dibuat, dan akan bergerak sesuai dengan pergerakan *marker*. Secara terperinci, perancangan dan aplikasi dibagi menjadi beberapa bagian, yaitu sebagai berikut:

Proses Penitikan GPS

Pada desain aplikasi yang akan dirancang, hal pertama yang dilakukan aplikasi adalah penitikan GPS pada posisi pengguna aplikasi. Proses ini sangat penting, karena titik koordinat tersebut nantinya akan digunakan sebagai titik pusat dari radar yang akan dibuat. Proses tersebut dilakukan pada *class MixView.java* oleh kumpulan sintak berikut :

```

try {
    GeomagneticField gmf = new GeomagneticField((float) mixContext.curLoc
        .getLatitude(), (float) mixContext.curLoc.getLongitude(),
        (float) mixContext.curLoc.getAltitude(), System
        .currentTimeMillis());

    angleY = Math.toRadians(-gmf.getDeclination());
    m4.set((float) Math.cos(angleY), 0f,
        (float) Math.sin(angleY), 0f, 1f, 0f, (float) -Math
        .sin(angleY), 0f, (float) Math.cos(angleY));
    mixContext.declination = gmf.getDeclination();
} catch (Exception ex) {
    Log.d("mixare", "GPS Initialize Error", ex);
}

```

Download Data

Proses selanjutnya adalah proses pen-download-an data-data lokasi tujuan, termasuk di dalamnya titik koordinat, alamat website STIMATA dan lain sebagainya. Data-data tersebut dibentuk dalam sintak berformat JSON. Berikut format data yang akan di download:

```

{
  "status": "OK",
  "num_results": 1,
  "results": [
    {
      "id": "1",
      "lat": "-7.974258987723271",
      "lng": "112.66039609909058",
      "elevation": "0",
      "title": "Kampus Stimata",
      "distance": "1",
      "has_detail_page": "1",
      "webpage":
      "http%3A%2F%2Fbeningnet.info%2Ftool%
      2Fstimata.php"
    }
  ]
}

```

Data JSON diatas kemudian akan di parsing sebagai data utama untuk pembuatan *marker* pada *class Json.java* dengan sintak sebagai berikut :

```

public Marker processMixareJSONObject(JSONObject jo, DataSource datasource) throws JSONException {
    Marker ma = null;
    if (jo.has("title") && jo.has("lat") && jo.has("lng")
        && jo.has("elevation")) {
        Log.v(MixView.TAG, "processing Mixare JSON object");
        String link=null;

        if(jo.has("has_detail_page") && jo.getInt("has_detail_page")!=0 && jo.has("webpage"))
            link=jo.getString("webpage");

        if(datasource.getDisplay() == DataSource.DISPLAY.CIRCLE_MARKER) {
            ma = new PoiMarker(
                unescapeHTML(jo.getString("title"), 0),
                jo.getDouble("lat"),
                jo.getDouble("lng"),
                jo.getDouble("elevation"),
                link,
                datasource);
        } else {
            ma = new NavigationMarker(
                unescapeHTML(jo.getString("title"), 0),
                jo.getDouble("lat"),
                jo.getDouble("lng"),
                0,
                link,
                datasource);
        }
    }
    return ma;
}

```

Pembuatan Radar

Radar yang dimaksud disini adalah radar yang terletak pada pojok kiri atas pada aplikasi yang berfungsi untuk menunjukkan arah/posisi dari lokasi yang dituju. Radar dapat terbentuk berdasarkan 2 komponen utama, yaitu :

- Titik Pusat : Ditentukan dari titik koordinat pemakai.
- Radius (jari-jari radar) : didapat dari value pada menu **Zoom Level** dengan nilai default 20 km.

Berikut sintak untuk membuat radar yang terletak pada *class RadarPoints.java*:

```

public class RadarPoints implements ScreenObj {
    /** The screen */
    public DataView view;
    /** The radar's range */
    float range;
    /** Radius in pixel on screen */
    public static float RADIUS = 40;
    /** Position on screen */
    static float originX = 0, originY = 0;
    /** Color */
    static int radarColor = Color.argb(100, 0, 0, 200);

    public void paint(PaintScreen dw) {
        /** radius is in KM. */
        range = view.getRadius() * 1000;
        /** Draw the radar */
        dw.setFill(true);
        dw.setColor(radarColor);
        dw.paintCircle(originX + RADIUS, originY + RADIUS, RADIUS);

        /** put the markers in it */
        float scale = range / RADIUS;

        DataHandler jLayer = view.getDataHandler();

        for (int i = 0; i < jLayer.getMarkerCount(); i++) {
            Marker pm = jLayer.getMarker(i);
            float x = pm.getLocationVector().x / scale;
            float y = pm.getLocationVector().y / scale;

            if (pm.isActive() && (x * x + y * y < RADIUS * RADIUS)) {
                dw.setFill(true);
            }
        }
    }
}

```

```

// For OpenStreetMap the color is changing based on the URL
dw.setColor(pm.getColour());

dw.paintRect(x + RADIUS - 1, y + RADIUS - 1, 2, 2);
}
}

/** Width on screen */
public float getWidth() {
    return RADIUS * 2;
}

/** Height on screen */
public float getHeight() {
    return RADIUS * 2;
}

```

Pembuatan Marker.

Marker merupakan sebuah obyek yang berfungsi sebagai pemicu (trigger) yang mana dalam aplikasi ini difungsikan sebagai penunjuk arah. Proses pembuatan *marker* dimulai setelah aplikasi mendapatkan titik koordinat dari proses pen-*download*-an data. Titik koordinat yang telah didapat akan dijadikan sebagai marker yang diberi nama sesuai dengan lokasi yang dituju. Dalam kasus ini, marker tersebut akan diberi nama “Kampus STIMATA”. Marker tersebut akan menjadi acuan atas pergerakan penunjuk arah yang diciptakan dari proses *rendering*.

Object Rendering

Proses inilah yang benar-benar menjadi ciri khas dari *Augmented Reality*, yakni 'menempatkan' obyek tiga-dimensi sintetis pada lingkungan dunia nyata yang tertangkap oleh kamera. Dalam kasus ini, obyek tersebut adalah arah panah yang berfungsi sebagai penunjuk arah bagi pengguna aplikasi.

Berikut sintak untuk pembuatan obyek yang berbentuk arah panah yang terdapat pada *class NavigationMarker.java* :

```

public void drawArrow(PaintScreen dw) {
    if (isVisible) {
        float currentAngle = MixUtils.getAngle(cMarker.x, cMarker.y, signMarker.x, signMarker.y);
        float maxHeight = Math.round(dw.getHeight() / 10f) + 1;

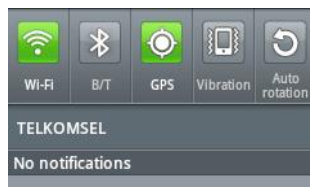
        //dw.setColor(dataSource.getColor(type));
        dw.setStrokeWidth(maxHeight / 10f);
        dw.setFill(false);

        Path arrow = new Path();
        float radius = maxHeight / 1.5f;
        float x=0;
        float y=0;
        arrow.moveTo(x-radius/3, y+radius);
        arrow.lineTo(x+radius/3, y+radius);
        arrow.lineTo(x+radius/3, y);
        arrow.lineTo(x+radius, y);
        arrow.lineTo(x, y-radius);
        arrow.lineTo(x-radius, y);
        arrow.lineTo(x-radius/3, y);
        arrow.close();
        dw.paintPath(arrow, cMarker.x, cMarker.y, radius*2, radius*2, currentAngle+90, 1);
    }
}

```

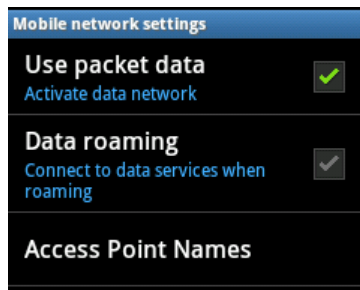
PENGUJIAN DAN HASIL

Sebelum menjalankan aplikasi, telah dipastikan device (*Smartphone*) yang digunakan dalam kondisi baik. Oleh karena aplikasi yang telah dibangun membutuhkan koneksi internet dan fitur GPS, maka sebelum dilakukan pengujian kedua fitur tersebut telah dipastikan dalam keadaan aktif, sebagaimana digambarkan pada gambar di bawah ini:



Gambar 9 Aktivasi Fitur Wi-Fi dan GPS

Pada gambar diatas, menunjukkan bahwa fitur Wi-Fi dan GPS telah aktif. Apabila tidak tersedia koneksi wireless, maka untuk koneksi internet dapat menggunakan koneksi GPRS yang disediakan oleh jaringan profider



Gambar 10 Aktivasi Koneksi GPRS

Pengujian Tampilan Awal Aplikasi

Setelah aplikasi dijalankan, maka langkah pertama adalah menampilkan gambar yang telah ditangkap oleh lensa kamera, dan kemudian dilanjutkan dengan pembuatan radar berwarna biru pada pojok kiri atas aplikasi. Pengujian ditampilkan dalam gambar di bawah ini:

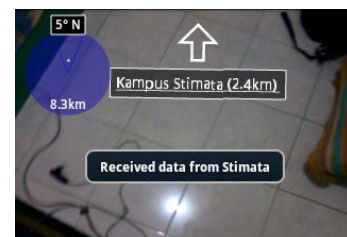


Gambar 11 Tampilan Awal Aplikasi

Kesimpulan yang dapat diambil adalah aplikasi dapat menampilkan gambar yang ditangkap oleh lensa kamera dan radar berwarna biru berhasil dibuat pada aplikasi.

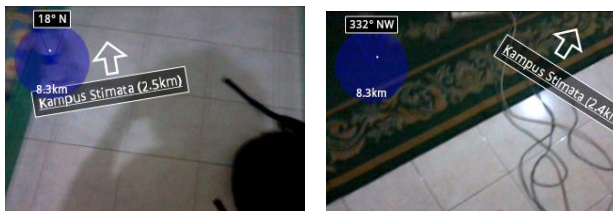
Pengujian Proses Download Data

Setelah tampilan awal aplikasi telah siap, langkah selanjutnya adalah proses *download* data lokasi tujuan dengan menggunakan koneksi internet. Setelah data ter-download, maka akan didapatkan beberapa data informasi mengenai lokasi yang akan dituju. Seperti titik lokasi GPS, alamat situs, dan lain sebagainya. Berikut adalah tampilan dari hasil pengujian yang telah dilakukan:



Gambar 12 Proses Download Data

Dari gambar di atas, dapat disimpulkan bahwa proses *download* data berjalan dengan baik. Sebagaimana muncul pesan “Received data from Stimata” dan sesegera mungkin aplikasi mengkalkulasi berapa jarak antara posisi pengguna dengan lokasi kampus Stimata beserta arah mana yang harus dituju oleh pengguna.

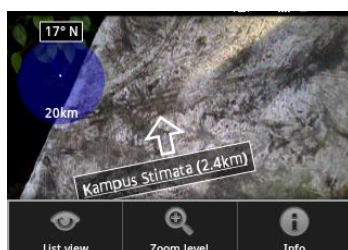


Gambar 13 Percobaan Respon Aplikasi Terhadap Pergerakan Pengguna

Dari beberapa gambar diatas dilakukan pengujian apakah reaksi apa yang akan dilakukan oleh aplikasi apabila pengguna mengarahkan kamera menuju arah yang berlawanan terhadap lokasi kampus Stimata. Kesimpulannya aplikasi bereaksi dengan baik. Ketika pengguna bergerak atau menghadap ke arah lain, obyek arah panah pada aplikasi selalu mengarah kepada lokasi kampus Stimata.

Pengujian Menu-Menu Pada Aplikasi

Berikut adalah tampilan menu dari aplikasi yang telah dibangun:

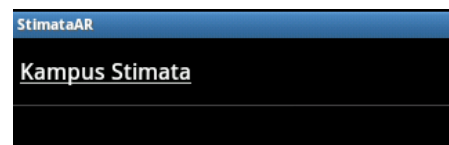


Gambar 14 Pengujian Tampilan Menu aplikasi

Dapat disimpulkan bahwa proses penampilan menu aplikasi berjalan dengan baik. Tampak dari kiri adalah menu List view, Map view, Zoom level, Search dan Info.

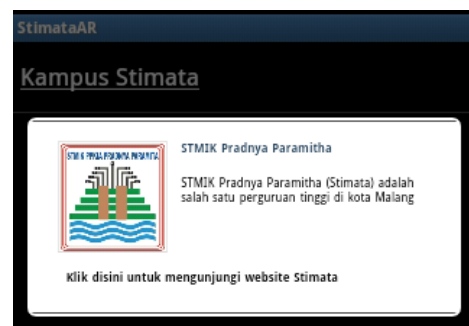
Pengujian Menu List View

Menu ini merupakan menu yang berfungsi untuk menunjukkan list atau daftar tempat yang ingin dituju. Oleh karena kampus Stimata hanya berlokasi pada satu alamat saja, maka daftar tujuan yang tampil hanya ada satu.



Gambar 15 Pengujian Menu List View

Pada gambar diatas, apabila tulisan “Kampus Stimata” diklik, maka akan muncul tampilan detail informasi seperti dibawah ini:



Gambar 16 Pengujian Tampilan Detail List View

Dari dua gambar diatas, menampilkan bahwa menu yang dimaksud telah dapat ditampilkan dengan sebagaimana mestinya, sehingga dapat disimpulkan bahwa menu List View telah berjalan dengan baik

Pengujian Menu Zoom Level

Menu ini berfungsi untuk mengatur diameter jangkauan dari radar yang tampil

pada pojok kiri atas aplikasi. Berikut adalah *screenshot*-nya:

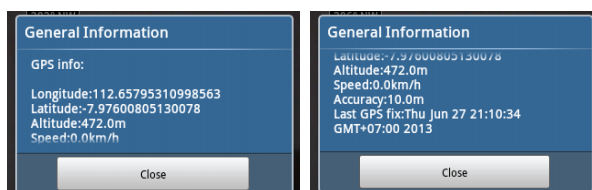


Gambar 17 Pengujian Menu *Zoom Level*

Dari gambar yang telah ditampilkan diatas, tampak bahwa aplikasi bereaksi dengan baik apabila tombol dibawah digeser sesuai dengan diameter yang diinginkan. Oleh karena itu, dapat disimpulkan bahwa bahwa menu *Zoom Level* telah berfungsi dengan baik.

Pengujian Menu *Info*

Menu ini mempunyai fungsi untuk memberikan informasi kepada pengguna, diantaranya: titik koordinat GPS, ketinggian, kecepatan, akurasi penitikan, waktu dan tanggal penitikan pada saat pengguna menggunakan aplikasi. Berikut adalah tampilan *screenshot* dari hasil pengujiannya:



Gambar 18 Pengujian Tampilan Menu-
Menu Aplikasi

Pada dua gambar diatas, menunjukkan bahwa aplikasi telah dapat menampilkan informasi-informasi sebagaimana mestinya, sehingga dapat

disimpulkan bahwa menu *Info* telah diimplementasikan dengan baik.

Pengujian Pada Daerah Bersinyal Lemah

Setelah semua menu pada aplikasi telah terimplementasikan dengan baik, maka selanjutnya dilakukan pengujian pada daerah yang berpotensi menimbulkan masalah dari aplikasi yang telah dibangun. Dalam hal ini daerah yang dimaksud adalah daerah yang mempunyai sinyal lemah, misalnya: lantai paling dasar dari gedung bertingkat, dimana tembok yang besar dan tebal menjadi penghalang sinyal *provider* untuk dapat ditangkap oleh perangkat *smartphone*.

Pada saat dilakukan pengujian pada daerah yang mempunyai sinyal lemah, pada aplikasi terdapat beberapa jeda waktu lebih lama pada proses *pen-download-an* data. Hal tersebut diakibatkan oleh berkurangnya kualitas jalur transportasi data yang tersedia sehingga mempengaruhi kecepatan transportasi data yang sedang berlangsung.

Sinyal lemah juga menjadi hambatan pada aplikasi pada saat dilakukan pengujian menu *Info*. Pada menu tersebut diperlukan beberapa proses yang memerlukan data GPS untuk menampilkan informasi kepada pengguna. Karena sinyal GPS yang diperlukan sebagai jalur komunikasi data terhalang, maka kecepatan transportasinya pun akan berkurang, sehingga akan menyebabkan jeda waktu yang lebih lama pada aplikasi

Dari hasil pengujian yang telah dilakukan, maka dapat disimpulkan bahwa aplikasi tetap dapat berjalan dengan baik meskipun sinyal yang tersedia dalam keadaan

lemah. Namun hal tersebut sudah pasti akan sangat mempengaruhi kecepatan dari aplikasi dalam proses *download* data-data yang diperlukan.

Namun, aplikasi ini tidak akan bisa berjalan pada daerah yang tidak terdapat sinyal sama sekali, contohnya: di dalam lift yang tertutup, daerah *blankspot* dan daerah-daerah lain yang tidak terdapat sinyal sama sekali.

Kesimpulan

Dari aplikasi yang telah dibangun dan setelah dilakukan pengujian, maka dapat diperoleh kesimpulan sebagai berikut:

1. Dengan menggunakan teknologi augmented reality, pembuatan aplikasi penunjuk jalan dapat dibuat semenarik dan seatraktif mungkin, selain itu integrasi teknologi augmented reality juga dapat memudahkan pengguna dalam mengoperasikan aplikasi.
2. Dengan keberhasilan pembuatan aplikasi ini, menunjukkan bahwa teknologi Augmented Reality dapat diimplementasikan dengan baik pada *smartphone* yang mempunyai Sistem Operasi Android

Saran

Aplikasi yang telah dibangun tidak lebih dari sekedar kompas digital yang berfungsi untuk menunjukkan arah tanpa memperhatikan halangan-halangan atau jalan yang dilalui pengguna. Diharapkan untuk kedepannya aplikasi ini dapat dikembangkan sehingga dapat memberikan petunjuk lebih detail kepada pengguna berdasarkan medan atau halangan pada jalan yang dilalui oleh pengguna.

Pada beberapa versi Android yang telah dicoba untuk menjalankan aplikasi ini tidak semua berjalan dengan baik. Hal tersebut dikarenakan engine Mixare yang digunakan tidak *compatible* dengan beberapa versi Android. Diharapkan suatu saat aplikasi ini dapat dikembangkan kembali agar dapat berjalan dengan baik pada semua versi Android yang ada.

DAFTAR PUSTAKA

- Azuma, Ronald T. 1997, A Survey of Augmented Reality Presence: Teleoperators and Virtual Environment, <http://www.cs.unc.edu/~azuma/ARpresence.pdf>, Tanggal Akses 23 Juli 2013.
- Safaat H, Nazruddin. 2011. *Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*, Bandung :Penerbit INFORMATIKA.
- Situs resmi Android. 2013. Android. <http://www.android.com/>. Tanggal akses 23 Juli 2013.
- Situs resmi Mixare. 2013. Mixare. <http://www.mixare.org/>. Tanggal akses 23 Juli 2013
- Situs resmi SIMATA. 2013. Lokasi Kampus. <http://www.pradnya.ac.id/kontak>. Tanggal akses 23 Juli 2013.