

**Perancangan dan Implementasi Aplikasi Pengajuan Advokasi
Online Memanfaatkan *Google Cloud Messaging* Pada Platform
Android
(Studi Kasus: BPMF LK FTI-UKSW)**

Artikel Ilmiah



Peneliti:

Nilam Ardhia Wiradhani(672011020)

Radius Tanone, S.Kom., M.Cs

**Program Studi Teknik Informatika
Fakultas Teknologi Informasi
Universitas Kristen Satya Wacana
Salatiga
April 2015**

**Perancangan dan Implementasi Aplikasi Pengajuan Advokasi
Online Memanfaatkan *Google Cloud Messaging* Pada Platform
Android
(Studi Kasus: BPMF LK FTI-UKSW)**

Artikel Ilmiah

**Diajukan kepada
Fakultas Teknologi Informasi
untuk memperoleh Gelar Sarjana Komputer**



Peneliti:

Nilam Ardhia Wiradhani(672011020)

Radius Tanone, S.Kom., M.Cs

**Program Studi Teknik Informatika
Fakultas Teknologi Informasi
Universitas Kristen Satya Wacana
Salatiga
April 2015**

**Perancangan dan Implementasi Aplikasi Pengajuan Advokasi
Online Memanfaatkan Google Cloud Messaging Pada Platform
Android
(Studi Kasus: BPFM LK FTI-UKSW)**

Oleh :
Nilam Ardhia Wiradhani
NIM : 672011020

ARTIKEL ILMIAH

Diajukan Kepada Program Studi Teknik Informatika guna memenuhi sebagian
dari persyaratan untuk mencapai gelar Sarjana Komputer

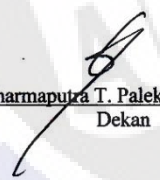
Disetujui oleh,



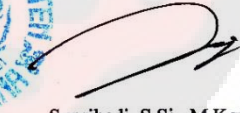
Radius Tanone, S.Kom., M.Cs.

Pembimbing 1

Diketahui oleh,



Dr. Dharmaputra T. Palekahelu, M.Pd.
Dekan



Supriyadi, S.Si., M.Kom.
Ketua Program Studi

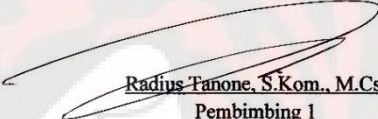
1956
**FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN SATYA WACANA
SALATIGA
2015**

Lembar Pengesahan

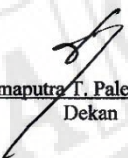
Judul Artikel : Perancangan dan Implementasi Aplikasi Pengajuan Advokasi Online Memanfaatkan Google Cloud Messaging Pada Platform Android (Studi Kasus: BPMF LK FTI-UKSW).


Nama Mahasiswa : Nilam Ardhia Wiradhani
NIM : 672011020
Program Studi : Teknik Informatika
Fakultas : Teknologi Informasi

Menyetujui,


Radius Tanone, S.Kom., M.Cs.
Pembimbing I

Mengesahkan,

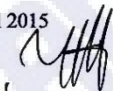
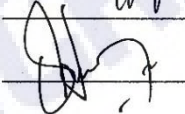

Dr. Dharmaputra T. Palekahelu, M.Pd.
Dekan


Suprihadi, S.Si., M.Kom.
Ketua Program Studi

Dinyatakan Lulus Ujian Pada Tanggal : 29 April 2015

Penguji :

1. Dr. Sri Yulianto J.P., S.Si., M.Kom.
2. Hendro S.Tampake, S.Kom., M.Cs.



PERNYATAAN TIDAK PLAGIAT

Saya yang bertanda tangan di bawah ini:

Nama : Nilam Ardha Wiradhani
NIM : 672011020 Email : nilamardhaw@gmail.com
Fakultas : Teknologi Informasi Program Studi : Teknik Informatika
Judul tugas akhir : Perancangan dan Implementasi Aplikasi Pengajuan Advokasi
Online Memanfaatkan Google Cloud Messaging pada Platform
Android (Studi Kasus : BPMF LK FTI-UKSW)
Pembimbing : 1. Radius Tanone, S.Kom., M.Cs.
2. _____

Dengan ini menyatakan bahwa:

1. Hasil karya yang saya serahkan ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar kesarjanaan baik di Universitas Kristen Satya Wacana maupun di institusi pendidikan lainnya.
2. Hasil karya saya ini bukan saduran/terjemahan melainkan merupakan gagasan, rumusan, dan hasil pelaksanaan penelitian/implementasi saya sendiri, tanpa bantuan pihak lain, kecuali arahan pembimbing akademik dan narasumber penelitian.
3. Hasil karya saya ini merupakan hasil revisi terakhir setelah diujikan yang telah diketahui dan disetujui oleh pembimbing.
4. Dalam karya saya ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali yang digunakan sebagai acuan dalam naskah dengan menyebutkan nama pengarang dan dicantumkan dalam daftar pustaka.

Pernyataan ini saya buat dengan sesungguhnya. Apabila di kemudian hari terbukti ada penyimpangan dan ketidakbenaran dalam pernyataan ini maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya saya ini, serta sanksi lain yang sesuai dengan ketentuan yang berlaku di Universitas Kristen Satya Wacana.

Salatiga, 23 Mei 2015





PERNYATAAN PERSETUJUAN AKSES

Saya yang bertanda tangan di bawah ini:

Nama : Nilam Ardha Wiradhani
NIM : 672011020 Email : nilamardhiaw@gmail.com
Fakultas : Teknologi Informasi Program Studi : Teknik Informatika
Judul tugas akhir : Perancangan dan Implementasi Aplikasi Pengajaran Adhokasi
Online Memanfaatkan Google Cloud Messaging Pada Platform
Android (Studi Kasus: BAMP LK FTI -UKSW)

Dengan ini saya menyerahkan hak *non-eksklusif** kepada Perpustakaan Universitas – Universitas Kristen Satya Wacana untuk menyimpan, mengatur akses serta melakukan pengelolaan terhadap karya saya ini dengan mengacu pada ketentuan akses tugas akhir elektronik sebagai berikut (beri tanda pada kotak yang sesuai):

- a. Saya mengizinkan karya tersebut diunggah ke dalam aplikasi Repositori Perpustakaan Universitas, dan/atau portal GARUDA
- b. Saya tidak mengizinkan karya tersebut diunggah ke dalam aplikasi Repositori Perpustakaan Universitas, dan/atau portal GARUDA**

* Hak yang tidak terbatas hanya bagi satu pihak saja. Pengajar, peneliti, dan mahasiswa yang menyerahkan hak non-eksklusif kepada Repositori Perpustakaan Universitas saat mengumpulkan hasil karya mereka masih memiliki hak copyright atas karya tersebut.

** Hanya akan menampilkan halaman judul dan abstrak. Pilihan ini harus dilampiri dengan penjelasan/ alasan tertulis dari pembimbing TA dan diketahui oleh pimpinan fakultas (dekan/kaprodi).

Demikian pernyataan ini saya buat dengan sebenarnya.

Salatiga, 29 Mei 2015

Nilam Ardha.w

Tanda tangan & nama terang mahasiswa

Mengetahui,

1956

Rados Tamre, S.Kom, M.Cs

Tanda tangan & nama terang pembimbing II



FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN SATYA WACANA
Jalan Diponegoro 52 - 60
Phone. (0298) 321212 (Hunting)
Fax. (0298) 321433
E-mail: fti@uksw.edu
Salatiga 50711 - INDONESIA



LEMBAR PERSETUJUAN PUBLISH JURNAL

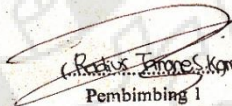
Dengan mempertimbangkan isi dari jurnal mahasiswa :

Nama Mahasiswa : Nilam Ardha W
NIM : 61201020

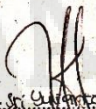
Maka jurnal ini dinyatakan :

LAYAK TERBIT TIDAK LAYAK TERBIT

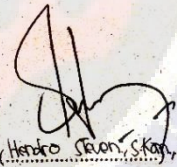
Menyetujui,


(Ratus Tono, S.Kom, M.Cs)
Pembimbing 1

(.....)
Pembimbing 2


(Dr. Sri Ulfahanto, P.Si, M.Ts)
Penguji 1




(Hendro Setono, S.Kom, M.Cs)
Penguji 2

Perancangan dan Implementasi Aplikasi Pengajuan Advokasi Online Memanfaatkan Google Cloud Messaging Pada Platform Android (Studi Kasus: BPMF LK FTI-UKSW)

¹⁾Nilam Ardhia Wiradhani, ²⁾Radius Tanone

Fakultas Teknologi Informasi

Universitas Kristen Satya Wacana

JL.Diponegoro 52- 60, Salatiga 50711, Indonesia

Email : ¹⁾672011020@student.uksw.edu ²⁾radiustanone@gmail.com

Abstract

Nowadays factor the collage students FTI-SWCU are not using the advocacy services is shame come face to face with functionaries BPMF LK-FTI SWCU sector advocacy, and they are have a hindrance to come in office of BPMF LK-FTI. To serve advocacy BPMF LK-FTI SWCU still using manual systems. It becomes an inhibiting factors. Using Google Cloud Messaging(GCM) services will designed mobile applications can be used to media for advocacy submission. Users application will receive message in the form of push notifications from a web-server about process advocacy progress and they are can reply the message. In the application to development process using research methods that is identification of problems, system design, create application, testing and implementation system, write research report. While the design method use prototype model. The research result is a client application for collage student FTI-SWCU that is application to submission online advocacy type native in Android platform which can overcome inhibiting factors to provide ease of students to submitted advocacy.

Keyword : Google Cloud Messaging, Push Notifications, Android, Advocacy Online

Abstrak

Saat ini faktor mahasiswa FTI-UKSW tidak menggunakan hak advokasinya adalah malu untuk datang langsung berhadapan dengan fungsionaris BPMF LK-FTI UKSW bidang advokasi, dan sedang berhalangan untuk datang langsung ke kantor BPMF LK-FTI UKSW. Untuk melayani advokasi BPMF LK-FTI UKSW masih menggunakan sistem manual. Hal tersebut menjadi faktor penghambat berjalannya layanan advokasi dengan baik. Memanfaatkan layanan *Google Cloud Messaging(GCM)* dirancang aplikasi *mobile* yang dapat digunakan sebagai media pengajuan advokasi. Pengguna aplikasi akan menerima pesan berupa *push notifications* dari *web-server* tentang perkembangan advokasinya dan pengguna dapat membalas pesan tersebut. Dalam proses pengembangan aplikasi digunakan metode penelitian yaitu mengidentifikasi masalah, perancangan sistem, implementasi dan pengujian sistem, serta penulisan laporan hasil penelitian. Sedangkan metode perancangan yang digunakan adalah *prototype*. Penelitian ini menghasilkan aplikasi *client* untuk mahasiswa FTI-UKSW yaitu aplikasi pengajuan advokasi *online* pada *platform* Android berjenis *native* yang dapat mengatasi faktor penghambat sehingga memberi kemudahan mahasiswa mengajukan advokasi.

Kata Kunci : Google Cloud Messaging, Push Notifikasi, Android, Advokasi Online

1) Mahasiswa Fakultas Teknologi Informasi Jurusan Teknik Informatika, Universitas Kristen Satya Wacana Salatiga.

2) Staff Pengajar Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana Salatiga.

1. Pendahuluan

Perkembangan teknologi informasi saat ini sangatlah cepat. Perkembangan tersebut mempengaruhi aktivitas pada kehidupan sehari-hari salah satunya untuk mempermudah penyampaian informasi. Pada Badan Perwakilan Mahasiswa Lembaga Kemahasiswaan Fakultas Teknologi Informasi Universitas Satya Wacana (BPMF LK-FTI UKSW), terdapat bidang Advokasi sebagai sarana untuk mahasiswa menyelesaikan masalah yang terkait dengan perkuliahan, dosen, nilai, kerja praktek, dan skripsi. Saat ini terdapat dua jalur pada layanan advokasi yaitu bertemu langsung dengan fungsionaris BPMF LK-FTI UKSW bidang advokasi menggunakan sistem kerja mahasiswa mengutarakan permasalahan yang dimiliki, menyebutkan data diri dan fungsionaris BPMF LK-FTI UKSW bidang advokasi akan mencatatnya kemudian akan diproses. Jalur berikutnya adalah menggunakan kotak advokasi yang terletak didepan kantor LK-FTI UKSW dengan sistem kerja mahasiswa menulis dikertas data dirinya, dan permasalahannya kemudian fungsionaris BPMF LK-FTI UKSW bidang advokasi akan memeriksa setiap advokasi yang masuk pada kotak advokasi kemudian akan memprosesnya.

Untuk mengetahui lebih jelas yang menjadi masalah dalam penelitian ini dilakukan wawancara dengan mahasiswa Fakultas Teknologi Informasi (FTI UKSW) serta fungsionaris BPMF LK-FTI UKSW bidang advokasi, dan penyebaran kuisisioner pra-penelitian terhadap 30 responden yaitu mahasiswa FTI-UKSW yang pernah mengajukan advokasi, dan memiliki masalah. Berdasarkan hasil wawancara dan kuisisioner pra-penelitian tersebut dengan diterapkan kedua jalur advokasi saat ini terdapat faktor penghambat yaitu belum terdapat sistem untuk melayani layanan pengajuan advokasi pada BPMF LK-FTI UKSW. Sistem pengajuan advokasi saat ini masih dilakukan secara manual. Sehingga waktu yang dibutuhkan lebih lama dalam pengolahan data dan pertukaran informasi yang terhambat jika dibandingkan dengan menggunakan teknologi. Hal tersebut menjadi penghambat layanan advokasi dapat berjalan dengan baik, yang seharusnya sangat bermanfaat dalam menyelesaikan masalah mahasiswa.

Dari hasil wawancara dan kuisisioner pra-penelitian dibutuhkan sebuah aplikasi *mobile* untuk membantu proses advokasi. Aplikasi *mobile* dipilih karena dapat digunakan oleh mahasiswa dan fungsionaris BPMF LK FTI-UKSW kapanpun dibutuhkan. Aplikasi *mobile* di implementasi pada perangkat Android berjenis *native*. *Google Cloud Messaging* (GCM) dipilih sebagai layanan pengirim *push* notifikasi perkembangan advokasi, karena *Google Cloud Messaging* (GCM) memiliki kelebihan dibanding layanan SMS atau telepon. Pada layanan SMS pengguna dapat mengirim pesan dengan jumlah karakter yang terbatas yaitu 160 karakter, jika 7-bit *character encoding* yang digunakan dan 70 karakter, jika 16-bit *Unicode UCS2 character encoding* yang digunakan[1]. Hal tersebut mengakibatkan pertukaran informasi antara mahasiswa FTI-UKSW dan fungsionaris BPMF LK FTI-UKSW terbatas pada pesan yang dikirimkan. Sedangkan pada layanan telepon penyampaian pesan tidak terbatas tetapi pada pihak fungsionaris BPMF LK FTI-UKSW terkendala untuk dijadikan sebuah hasil data pengaduan. Dari kedua layanan tersebut *Google Cloud Messaging* (GCM) memiliki beberapa fungsi yang sama dengan layanan SMS dan telepon untuk

pertukaran informasi, tetapi pada *Google Cloud Messaging* (GCM) tidak memiliki batas pada karakter dalam pesan yang dikirimkan, dan data tersebut lebih mudah dijadikan sebuah hasil data pengaduan dalam bentuk teks. Pada aplikasi yang akan di implementasi *Google Cloud Messaging* (GCM) digunakan agar mahasiswa FTI-UKSW dan fungsionaris BPMF LK-FTI UKSW bidang advokasi untuk menerima *push* notifikasi pada pertukaran informasi yang berguna ketika setiap terdapat pesan baru.

Dari latar belakang masalah yang ada, dilakukan penelitian yang bertujuan memanfaatkan perkembangan teknologi untuk merancang dan implementasi sebuah aplikasi *mobile* untuk digunakan sebagai media pengajuan advokasi antara mahasiswa FTI UKSW dengan fungsionaris BPMF LK-FTI UKSW bidang advokasi memanfaatkan layanan *Google Cloud Messaging* (GCM). Dengan menggunakan aplikasi ini diharapkan layanan advokasi dapat menjadikan pengolahan data lebih cepat dan pertukaran informasi menjadi tidak terhambat sehingga layanan advokasi yang bermanfaat dalam menyelesaikan masalah mahasiswa dapat berjalan dengan baik.

Rumusan permasalahan pada penelitian yang dilakukan yaitu bagaimana menghasilkan aplikasi pengajuan advokasi *online* FTI-UKSW memanfaatkan Layanan *Google Cloud Messaging* pada *Platform* Android.

Dalam penelitian ini, permasalahan dibatasi pada pengajuan advokasi secara *online* dengan menerapkan *Google Cloud Messaging* sebagai pengiriman *push* notifikasi, pengguna dapat melakukan obrolan sederhana berupa teks, aplikasi yang dibangun pada *platform* android berjenis *native*, tidak membahas mengenai kecepatan *bandwith* internet dan aplikasi tidak membahas keamanan data.

2. Kajian Pustaka

Penelitian sebelumnya yang berjudul Penerapan *Google Cloud Messaging Service* Sebagai Pengiriman Pesan Singkat pada Aplikasi Penjualan Batik, menyatakan bahwa perangkat *mobile* yang sebelumnya hanya dapat digunakan untuk berkomunikasi. Saat ini, banyak fungsi yang bisa dimanfaatkan untuk mempermudah segala aktivitas yang ada. Sebagai contoh bentuk pemanfaatan teknologi *mobile* dalam kehidupan sehari-hari adalah aplikasi jual beli *online*[2]. Kelebihan pada aplikasi dalam penelitian ini adalah meningkatkan mobiltas pengguna aplikasi yaitu penjual dan pembeli batik.

Pada penelitian yang berjudul *Personnel Schedule Distribution Using Google Cloud Messaging* membahas tentang aplikasi *mobile* untuk mengatur jadwal memanfaatkan teknologi GCM pada Android. GCM bekerja dengan menerima permintaan dari aplikasi *server* dan meneruskannya ke aplikasi *client* Android di perangkat *mobile*[3]. Kelebihan pada aplikasi ini membuat pengguna dapat selalu mengatur jadwal sehari-harinya dengan notifikasi berguna sebagai pengingat yang bersifat *realtime*.

Pada dua penelitian sebelumnya, aplikasi hanya dapat menerima notifikasi yang berisi pesan tanpa pengguna bisa memberi respon kembali. Perbedaan dengan penelitian yang dilakukan saat ini adalah pada aplikasi penelitian sebelumnya, pengguna dapat menerima notifikasi berupa pesan tetapi tidak bisa

memberi respon. Sedangkan aplikasi pada penelitian ini pengguna dapat memberi respon balasan setelah menerima pesan yang berupa notifikasi.

Google Cloud Messaging (GCM) adalah sebuah layanan servis yang dikembangkan oleh Google Inc. GCM dapat membantu para pengguna dalam mengirimkan data dari *server* ke aplikasi pada perangkat android. Pesan yang dikirimkan melalui GCM memberikan informasi mengenai suatu data baru dalam aplikasi. Sebagai contoh ketika ada pesan baru yang masuk maka akan muncul notifikasi ke perangkat *mobile* android. Layanan GCM melakukan pengaturan semua hal yang berhubungan dengan antrian pesan atau notifikasi dan pengirimnya ke perangkat *mobile* yang menjadi target. GCM mempunyai dua konsep utama yaitu *components* dan *credentials*. *Components* adalah sebuah entitas yang memiliki peran penting dalam GCM. Sedangkan *credentials* merupakan ID dan token dari setiap perangkat android yang digunakan oleh GCM untuk memastikan apakah semua pesan yang dikirim sampai pada perangkat Android. Pada *components* terdapat *Client App*, *3rd-party Application Server*, dan *GCM Connection Server*. Sedangkan pada *credentials* terdapat *sender id*, *application id*, *registration id*, *Google account user*, dan *sender auth token*. Untuk menggunakan layanan GCM dibutuhkan perangkat Android bersistem operasi minimal 2.2, dan memiliki akun *Google* didalamnya[4]. Dalam implementasi GCM, terdapat beberapa bagian yang terlibat yaitu : *Google-provided connection server*, *3rd-party application server*, dan *client-app*.



Gambar 1 Arsitektur *Google Cloud Messaging*

Gambar 1 merupakan gambar arsitektur *Google Cloud Messaging*(GCM) yang menggambarkan bahwa Google menyediakan layanan *GCM Connection Server* yang akan berguna untuk menerima pesan dari *3rd-party application server* dan mengirimkan pesan tersebut pada aplikasi yang sudah terpasang pada perangkat Android yang berperan sebagai *Client app*. Google hanya menyediakan API (Application Programming Interface) *Cloud Messaging* hanya untuk perangkat Android.

JSON (*JavaScript Object Nation*) adalah sebuah objek data yang dapat dipanggil dan ditampilkan oleh sebuah program, baik oleh PHP, JavaScript, C#, Java, dan bahasa pemrograman lainnya yang sudah mendukung dan memiliki *JSON library*. *JSON* juga bisa memasukkan data ke dalam *database* dan menampilkannya secara dinamis dalam suatu program. Objek yang terdapat pada *JSON* secara keseluruhan merupakan *array* di mana setiap objek tersebut *JSON* merubah setiap tipe data lalu menampilkannya dalam bentuk *string* agar mudah terbaca oleh program[5].

Android adalah sebuah sistem operasi *mobile* yang mengadopsi sistem operasi Linux tetapi telah dimodifikasi. Android telah diambil alih oleh Google pada tahun 2005. Keuntungan utama dari Android adalah adanya pendekatan

aplikasi secara terpadu. Pengembangan hanya berkonsentrasi pada aplikasi dan bisa berjalan pada beberapa perangkat yang berbeda selama masih berbasis Android[6].

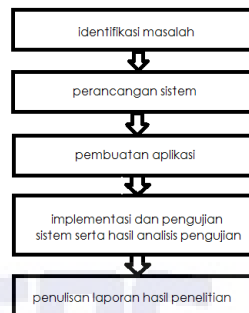
Bahasa pemrograman yang digunakan pada bagian *server* sebagai penghubung aplikasi dengan *database* adalah *Hypertext Preprocessor* (PHP). Script PHP adalah bahasa program yang berjalan pada sebuah *webserver*, atau sering disebut *server-side* dimana agar dapat menjalankan script ini dibutuhkan tiga program utama, yaitu *webserver* (dapat berupa IIS dari windows atau apache), modul PHP dan juga *web-browser*. Oleh karena itu, PHP dapat melakukan apa saja yang bisa dilakukan program CGI lain[7].

MySQL adalah sebuah *server database open source* yang terkenal dan digunakan berbagai aplikasi terutama untuk *server* atau membuat *web*. MySQL berfungsi sebagai SQL (*Structured Query Language*). Kehandalan suatu sistem basisdata (DBMS) dapat diketahui dari cara kerja pengoptimasi-nya dalam melakukan proses perintah-perintah SQL yang dibuat oleh pengguna maupun program-program aplikasi yang memanfaatkannya. MySQL mendukung operasi basisdata transaksional maupun operasi basisdata non-transaksional[8]. Untuk mengelola basis data dalam MySQL, salah satunya adalah menggunakan phpMyAdmin. Dengan phpMyAdmin, seseorang dapat membuat *database*, membuat tabel, mengisi data, dan lain-lain dengan mudah, tanpa harus menghafal baris perintahnya. phpMyAdmin merupakan bagian untuk mengelola basis data MySQL yang ada di komputer[9].

Badan Perwakilan Mahasiswa Fakultas (BPMF) adalah Organisasi Kemahasiswaan yang memiliki kewenangan untuk merumuskan GBPHLK di aras fakultas, serta membantu ketua Senat Mahasiswa Fakultas (SMF) terpilih untuk membentuk kepengurusan SMF. Tugas dan wewenang yaitu Mengutus wakil mahasiswa fakultas untuk duduk di Badan Perwakilan Mahasiswa Universitas (BPMU), menarik kembali wakil mahasiswa fakultas yang duduk di BPMU, membantu ketua SMF terpilih untuk membentuk kepengurusan SMF, mengajukan nama fungsionaris SMF terpilih untuk diangkat oleh Senat Mahasiswa Universitas (SMU), dan memilih juga menetapkan dan memberhentikan ketua SMF. Pada BPMF LK-FTI UKSW memiliki bidang advokasi, yaitu yang berguna sebagai media penyelesaian masalah yang dimiliki mahasiswa FTI-UKSW[10].

3. Metode Penelitian

Metode penelitian yang dilakukan dalam merancang aplikasi menggunakan lima tahapan penelitian yaitu: 1) Identifikasi Masalah. 2) Perancangan Sistem. 3) Pembuatan Aplikasi. 4) Implementasi dan Pengujian Sistem serta Analisis Hasil Pengujian. 5) Menulis Laporan Penelitian. Tahapan-tahapan yang dilakukan dalam metode penelitian ini dapat ditunjukkan pada Gambar 1.



Gambar 2 Tahapan Penelitian

Pada Gambar 2 merupakan metode penelitian yang digunakan dalam merancang aplikasi. Penjelasan tahapan metode penelitian adalah sebagai berikut :

1. Identifikasi Masalah

Tahap ini yang dilakukan adalah mengamati dan meneliti efektifitas pengajuan advokasi dengan dua jalur yang saat ini digunakan yaitu menggunakan kotak advokasi di kantor LK-FTI UKSW, bertemu langsung dengan fungsionaris BPMF LK-FTI UKSW bidang advokasi dan faktor-faktor yang menjadi pengambatnya.

2. Perancangan Sistem

Penelitian ini digunakan *Unified Modelling Language* (UML) yang meliputi *usecase* diagram, *activity* diagram, *class* diagram dan *deployment* diagram. Selain itu dilakukan perancangan *database*, dan *user interface* aplikasi.

3. Pembuatan Aplikasi

Tahap ini dilakukan pembuatan aplikasi sesuai dengan perancangan sistem yang dibuat. Pembuatan aplikasi *web-server* yang akan diletakkan pada komputer BPMF LK-FTI UKSW terdiri dari dua yaitu aplikasi *server* beserta *web service*-nya yang berbasis *web* dibuat dengan menggunakan framework PHP dan framework Javascript untuk *user interface*. Sedangkan untuk aplikasi *client* berbasis *mobile* dibuat dengan menggunakan bahasa pemrograman Java Android menggunakan *tools* Eclipse Luna, memanfaatkan *library Google Play Services* untuk mendapatkan GCM.jar didalam folder *extras*, dan *AChartEngine* untuk membuat grafik pada *report*.

4. Pada Implementasi dan Pengujian Sistem serta Analisis Hasil Pengujian

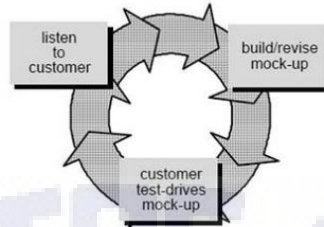
Tahap ini dilakukan pengimplementasian aplikasi yang telah selesai dibuat pada perangkat *mobile* Android, kemudian dilakukan pengujian/*testing* untuk mengetahui apakah aplikasi telah sesuai dengan perancangan yang dilakukan dan apakah sudah tidak ditemukan kesalahan/*bug* pada aplikasi.

5. Penulisan Laporan Hasil Penelitian

Memiliki tujuan yaitu mendokumentasikan setiap proses yang dilakukan di dalam penelitian yang telah dilakukan dalam bentuk laporan tertulis dan akan menjadi laporan hasil penelitian.

Pada penelitian ini, metode perancangan sistem yang digunakan untuk membangun aplikasi adalah model *prototyping*. Metode ini memungkinkan

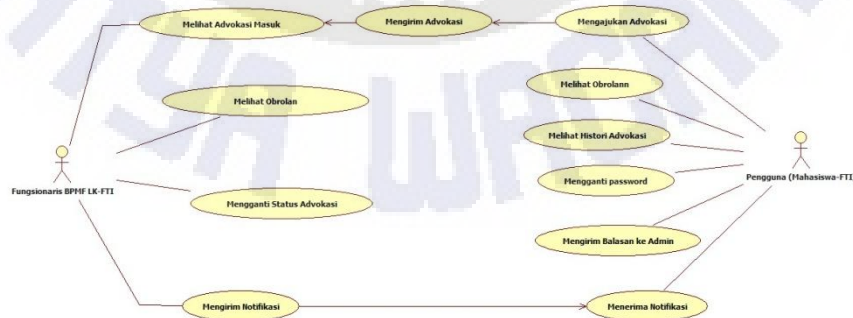
pengembang dan pelanggan dapat saling melakukan interaksi selama pembuatan sistem. Bagan mengenai *prototype* dapat dilihat pada gambar 3.



Gambar 3 Bagan *Prototype Model*

Proses atau tahapan dalam penyelesaian masalah pada prototyping model adalah : 1) Pengumpulan kebutuhan di mana pihak pengembang aplikasi mencari tahu kebutuhan pengguna, tujuan umum dan gambar bagian-bagian yang akan dibutuhkan berikutnya. 2) Pengkodean sistem merupakan suatu tahap di mana jika *prototyping* sudah disepakati maka *prototype* tersebut akan diterjemahkan ke dalam bahasa pemrograman yang sesuai. Pengujian sistem dilakukan apabila sistem sudah dibentuk menjadi suatu perangkat lunak, maka pada tahap selanjutnya adalah menguji sistem tersebut dengan variabel-variabel nyata sesuai dengan kebutuhan *client*. 3) Evaluasi sistem merupakan suatu tahap dimana pengguna akan mengevaluasi kerja sistem yang sudah selesai, apakah sesuai dengan yang diharapkan. Apabila sistem tidak sesuai dengan yang diharapkan oleh pengguna maka sistem akan dibangun ulang maupun diperbaiki melalui tahap pengkodean sistem dan selanjutnya. Perulangan ketiga proses ini terus berlangsung hingga semua kebutuhan terpenuhi[11].

Pada tahap perancangan sistem UML (*Unified Modelling Language*) yang berfungsi menggambarkan prosedur dan proses kerja dari aplikasi yang dirancang. *Usecase* diagram adalah gambaran dari beberapa atau seluruh actor, *usecase*, dan interaksi antara komponen-komponen tersebut yang menampilkan bagaimana sistem akan dibangun[12]. *Usecase* diagram pada aplikasi ini terdapat 2 aktor yaitu admin dan pengguna.

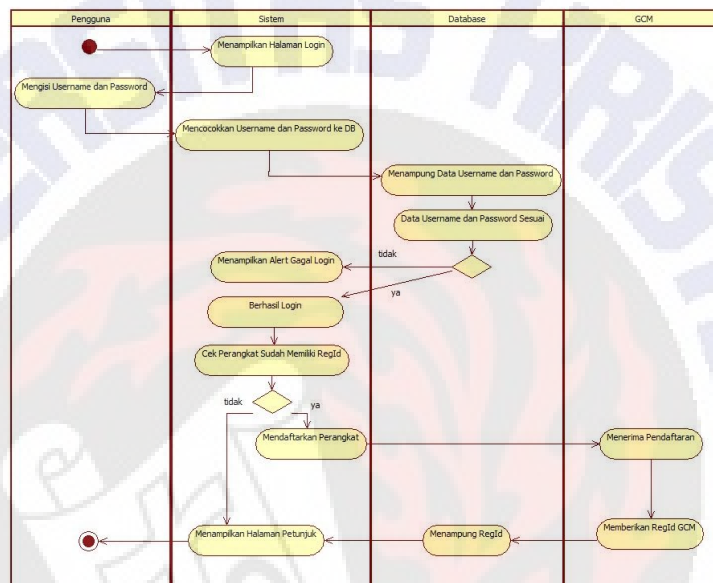


Gambar 4 *Usecase Diagram* Aplikasi

Gambar 4 menunjukkan bahwa pada aplikasi pengajuan advokasi *online* pada aktor pengguna dapat melihat pesan yang dikirim oleh *admin*, berbalas pesan dengan admin, melihat *history* pengajuan advokasi yang telah dikirim, mengganti *password* akun advokasi, *logout* aplikasi, mengirim advokasi, melihat *report*

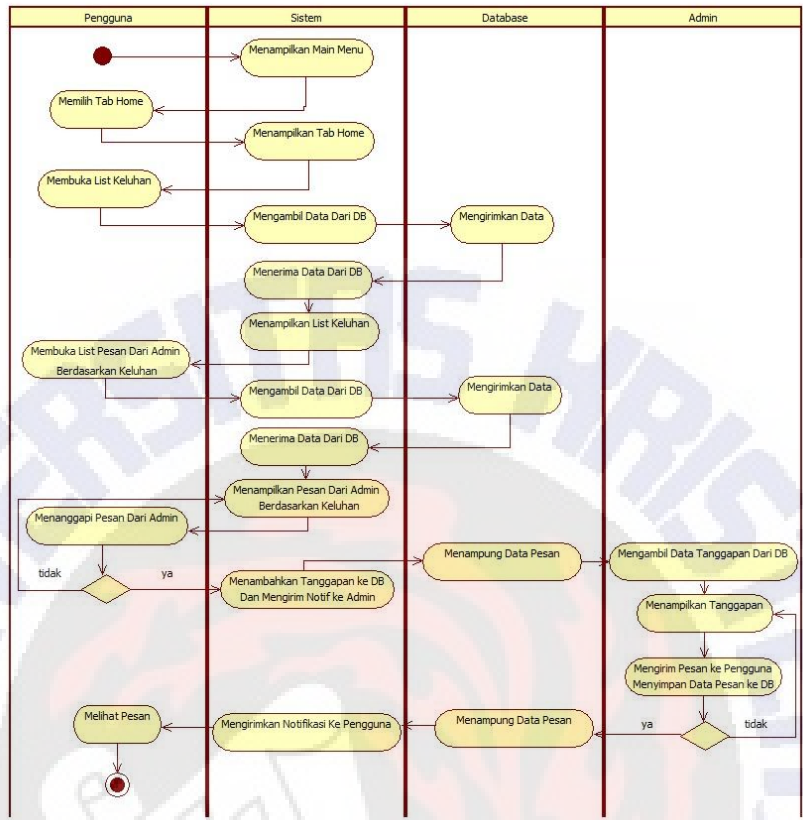
sebagai *report*, dan menerima notifikasi berisi pesan yang dikirim oleh admin. Sedangkan pada aktor *admin* menunjukkan bahwa pada aplikasi pengajuan advokasi *online* admin dapat mengelola dan menampilkan *report* advokasi, melihat pengajuan advokasi yang sudah masuk, mengirim *push* notifikasi pada pengguna, mengganti status advokasi dan mengirim pesan sekaligus notifikasi pada pengguna.

Activity diagram menggambarkan aliran aktifitas dalam sistem yang sedang dirancang, bagaimana masing-masing alur berawal, *decision* yang mungkin terjadi, dan bagaimana sistem berakhir [13].



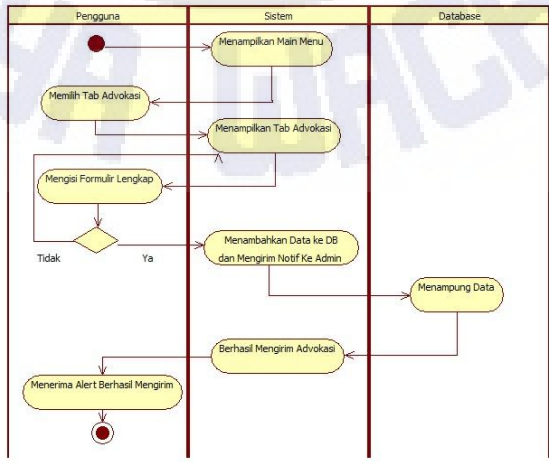
Gambar 5 Activity Diagram pengguna Login

Gambar 5 menunjukkan aktivitas pengguna ketika melakukan *login* aplikasi. Saat melakukan *login* sistem mencocokkan data dari *database* dengan *default username* yaitu NIM dan *password* yaitu tanggal lahir, jika keduanya cocok pada data dalam *database* maka akan berhasil masuk kedalam sistem. Kemudian sistem akan memeriksa apakah perangkat sudah terdaftar pada GCM. Jika perangkat belum terdaftar pada GCM otomatis akan sekaligus mendaftarkan perangkat pada GCM.



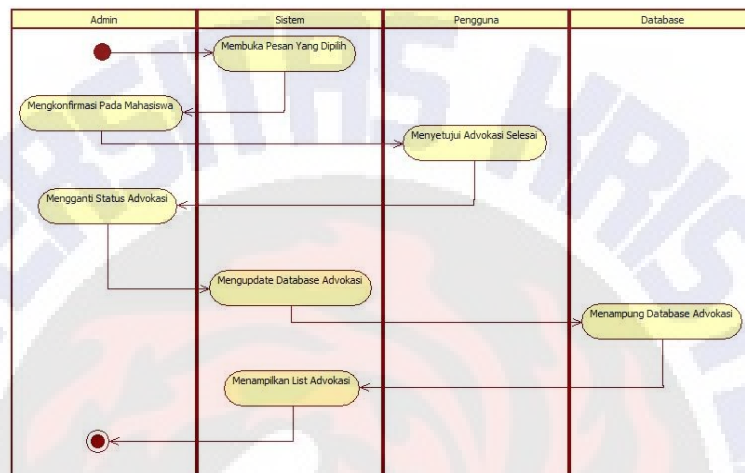
Gambar 6 Activity Diagram pengguna melihat pesan

Gambar 6 menunjukkan aktivitas pengguna ketika melihat pesan, sistem akan memanggil data pengajuan advokasi dari *database* berdasarkan NIM dari pengguna ketika *login* yang ditampilkan dalam *listview*. Kemudian, saat pengguna memilih advokasi pada *listview*, maka akan muncul pesan dari admin berdasarkan advokasi yang telah dipilih. Jika pengguna ingin memberi tanggapan atau melakukan obrolan dengan cara mengirim pesan berupa teks maka sistem akan menyampaikan pesan kepada admin dan menampungnya didalam *database*.



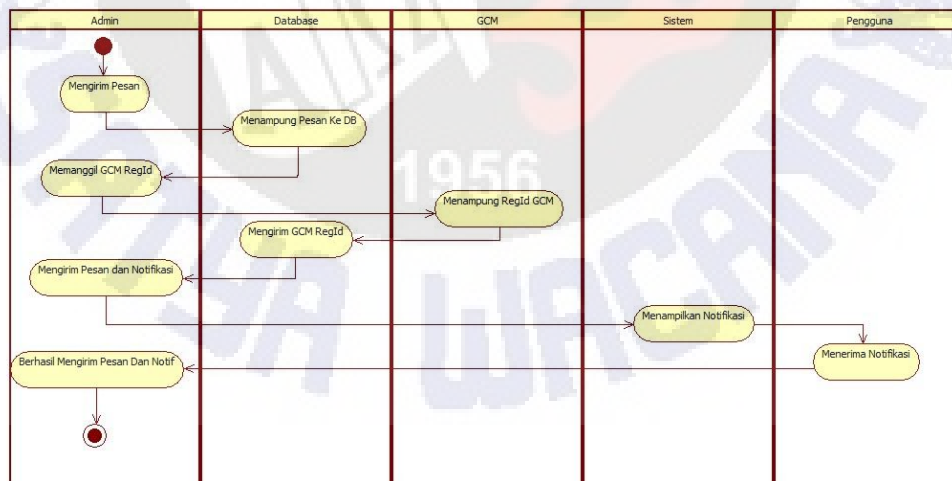
Gambar 7 Activity Diagram pengguna mengajukan advokasi

Gambar 7 menunjukkan aktifitas pengguna saat mengajukan advokasi. Pada pengajuan advokasi nama, nim, progdi terisi secara otomatis yang diambil dari data pengguna saat *login* dan tidak dapat dirubah. Pengguna diharuskan mengisi secara lengkap, untuk dapat mengirim advokasi. Data pengajuan advokasi yang dikirim akan ditampung dalam *database* untuk dapat dilihat dan diproses oleh admin. Ketika pengajuan advokasi berhasil dikirim maka akan mendapat tanda sukses dari sistem.



Gambar 8 Activity Diagram Admin Mengganti Status Advokasi

Gambar 8 menunjukkan aktifitas pengguna saat admin mengganti status advokasi. Setelah status advokasi diganti, maka advokasi tersebut akan berstatus selesai atau sukses.

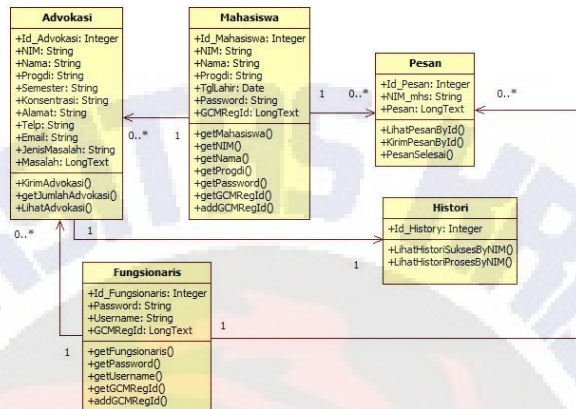


Gambar 9 Activity Diagram pengguna menerima notifikasi

Gambar 9 menunjukkan aktifitas pengguna saat menerima notifikasi. Saat melakukan *login* otomatis akun akan mendaftarkan perangkat pada GCM yang ditampung dalam *database* jika perangkat belum terdaftar pada GCM. Ketika admin mengirim pesan, sistem akan mengambil id registrasi GCM yang sudah didapatkan ketika *login*. Kemudian sistem aplikasi *mobile* yang dikendalikan

admin akan meminta GCM untuk merespon mengirimkan notifikasi dan pesan. Kemudian, sistem akan menerima notifikasi dan pesan dari admin.

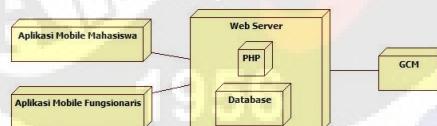
Class diagram menggambarkan struktur dan dekripsi *class*, *package*, dan objek beserta hubungan satu sama lain seperti *containtmen*, pewarisan, asosiasi, dan lain-lain[14]. *Class* diagram pada aplikasi dapat dilihat pada Gambar 9.



Gambar 10 Class Diagram Aplikasi Pengajuan Advokasi Online

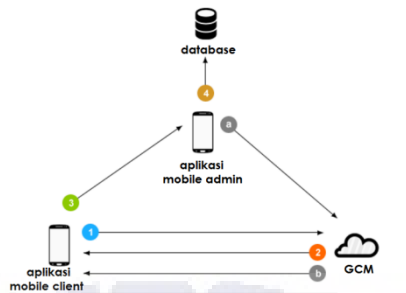
Gambar 10 merupakan *class* diagram dari aplikasi pengajuan advokasi online. *Class* diagram terdiri dari model, *view* dan controller. Model adalah penghubung antara fungsi dengan *database*. Controller adalah penggerak yang berfungsi menerima perintah dari *view* lalu meneruskan ke model. Pada *class* diagram dapat dilihat bahwa satu mahasiswa dapat mengirim banyak pengajuan advokasi, satu fungsionaris dapat memproses banyak advokasi.

Deployment diagram merupakan susunan fisik dari sebuah sistem, yang menunjukkan tata letak pada bagian-bagian *software* yang berjalan pada bagian-bagian pada *hardware*.



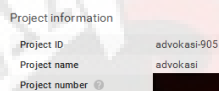
Gambar 11 Deployment Diagram Aplikasi Pengajuan Advokasi Online

Gambar 11 menggambarkan *deployment* diagram dari aplikasi pengajuan advokasi online. Pada *deployment* diagram terdapat dua perangkat lunak yaitu aplikasi *mobile*, dan aplikasi *web*. Kedua aplikasi terhubung oleh GCM dan *web service*. Disini, *web-service* dan GCM sebagai penghubung antara kedua aplikasi dengan *database*. Pada diagram terdapat *webserver* yang menghasilkan *web-service* untuk akses data ke aplikasi *mobile* dan aplikasi *web*. *Webserver* ini dibangun oleh bahasa pemrograman PHP yang juga terhubung oleh *database server*. Berdasarkan *deployment* diagram ini dirancang arsitektur sistem dari aplikasi pengajuan advokasi online.



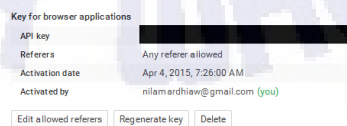
Gambar 12 Arsitektur Sistem Aplikasi Pengajuan Advokasi Online

Gambar 12 yang merupakan arsitektur sistem. Terdapat dua aplikasi dalam arsitektur ini yaitu aplikasi *web* dan aplikasi *mobile*. Aplikasi *web* diimplementasikan pada laptop atau PC di kantor LK FTI-UKSW yang ditujukan untuk fungsionaris BPFM LK-FTI UKSW bidang advokasi. Sedangkan aplikasi *mobile* diimplementasikan pada perangkat *mobile* Android yang ditujukan oleh pengguna dimana pada kasus ini adalah mahasiswa FTI-UKSW. Untuk menjalankan aplikasi kedua perangkat harus terhubung dengan koneksi *internet*, terhubung *web service* dan GCM yang berperan menghubungkan aplikasi *web* dan *mobile* dengan *database* pada *web server*. Untuk mendapat registrasi ID GCM dibutuhkan langkah sebagai berikut : 1) Pertama perangkat Android akan mengirimkan *sender id*, *application id* ke GCM untuk didaftarkan pada sistem. *Sender id* didapatkan dengan cara membuat proyek baru pada *Google Developers Console*. Proyek yang sudah berhasil dibuat secara otomatis akan melakukan *generateSender id* secara otomatis. *Sender id* digunakan pada saat proses registrasi ke GCM untuk mengidentifikasi *3rd-party application server* untuk melakukan pengiriman pesan ke perangkat tujuan yang dapat dilihat pada Gambar 13.



Gambar 13 Sender Id yang didapatkan dari Google Developers Console

Sedangkan *APIs key* diperoleh dengan cara membuat *key* baru pada proyek yang terdapat pada *Google Developers Console*. *API key* akan digunakan oleh *server* pada saat melakukan *request* ke layanan GCM. *API Key* dapat dilihat pada gambar 14.



Gambar 14 Hasil API Key yang Digunakan Sebagai Application Id

2) Setelah sukses proses registrasi GCM, maka akan dikirimkan *registration id* ke perangkat Android. *Registration id* akan digunakan untuk kode identitas tiap *account* pengguna. 3) *Registration id* didapatkan dari GCM pada saat pengguna melakukan registrasi ke GCM kemudian akan dikirimkan ke aplikasi *server* untuk disimpan ke dalam *database*. 4) Aplikasi *server* akan menyimpan *registration id* pada *database* dengan mengeksekusi perintah SQL.

4. Hasil dan Pembahasan

Penelitian ini menghasilkan dua aplikasi yang diimplementasikan pada platform Android yaitu aplikasi *mobile client* untuk mahasiswa FTI UKSW dan aplikasi *mobile admin* untuk fungsionaris BPMF LK FTI-UKSW bidang advokasi. Aplikasi dibangun menggunakan library *gcm.jar* yang didapatkan setelah mengunduh *Google Play Services* pada tools Eclipse Luna yang digunakan untuk membangun aplikasi *mobile*.

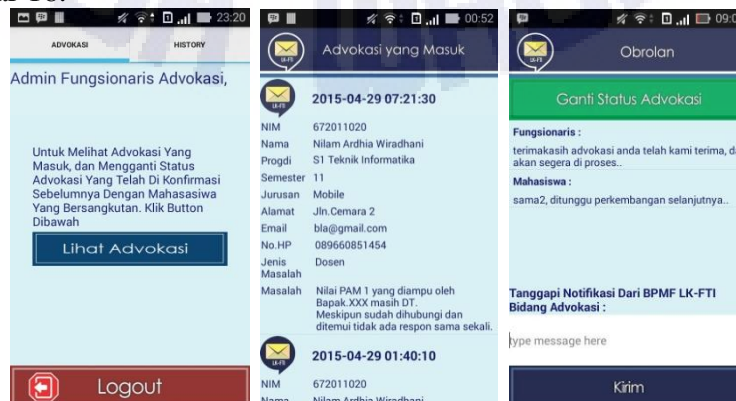
Pada aplikasi *mobile admin*, diharuskan *login* dengan *username* nama fungsionaris dan *password* yang dapat dilihat pada Gambar 15.



Gambar 15 Halaman Login Aplikasi Mobile Admin

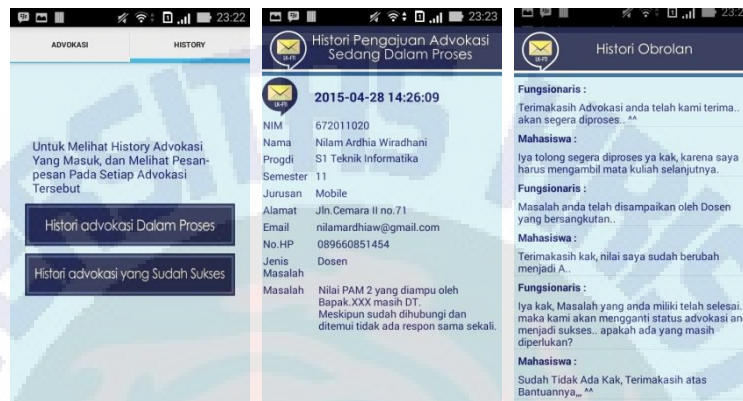
Gambar 15 merupakan tampilan dari halaman *login* aplikasi admin. Sistem akan mencocokkan *username* dan *password* berdasarkan data pada *database*. Jika terdapat data yang cocok maka pengguna akan berhasil *login*. Data tersebut diambil dari *database* yang dirubah dalam format JSON agar dapat diakses oleh aplikasi. Saat *login*, pengguna akan sekaligus mendapatkan ID registrasi *GCM*. Saat melakukan *login* sistem memeriksa apakah perangkat *mobile* sudah terdaftar pada *GCM*, jika belum maka perangkat akan didaftarkan terlebih dahulu agar dapat menerima notifikasi.

Setelah berhasil *login* akan muncul halaman utama dari aplikasi yang terdapat menu Tab Advokasi dan Tab *History*. Untuk Tab Advokasi dapat dilihat pada Gambar 16.



Gambar 16 Tab Advokasi Aplikasi Mobile Admin

Gambar 16 menampilkan menu Tab Advokasi bahwa admin dapat melihat advokasi yang masuk berdasarkan urutan waktu, mengganti status advokasi, dan membuka obrolan pada user. Saat admin mengganti status advokasi, maka advokasi tersebut akan berbuah menjadi telah selesai. Sedangkan untuk Tab *History* dapat dilihat pada Gambar 17.



Gambar 17 Tab *Report* Aplikasi *Mobile Admin*

Gambar 17 menampilkan menu Tab *History*, admin dapat melihat advokasi yang masuk berdasarkan advokasi yang telah selesai dan advokasi sedang dalam proses penyelesaian. Admin juga bisa melihat riwayat obrolan setiap advokasi.



Gambar 18 Notifikasi Pesan Aplikasi *Mobile Admin*

Gambar 18 menampilkan notifikasi saat aplikasi *mobile client* yang digunakan oleh mahasiswa FTI-UKSW mengirim advokasi baru atau merespon pesan yang dikirim oleh admin. Notifikasi merupakan *push* notifikasi yang masuk bersifat *realtime* dan tidak mengurangi kuota pengguna aplikasi. Ketika admin membuka notifikasi, aplikasi akan menampilkan obrolan.

Pada aplikasi *mobile Client* yang digunakan mengajukan advokasi secara *online* mahasiswa diharuskan *login* sebelum masuk dalam sistem.



Gambar 19 Halaman *Login* Aplikasi *Mobile Client*

Gambar 19 merupakan tampilan dari halaman *login* aplikasi. Sistem akan mencocokkan *username* dan *password* berdasarkan data pada *database*. Jika terdapat data yang cocok berdasarkan *username* yaitu NIM mahasiswa dan *password* adalah tanggal lahir mahasiswa, maka pengguna akan berhasil *login*. Data tersebut diambil dari *database* yang dirubah dalam format JSON agar dapat diakses oleh aplikasi. Saat *login*, pengguna akan sekaligus mendapatkan ID registrasi *GCM*. Saat melakukan *login* sistem memeriksa apakah perangkat *mobile* sudah terdaftar pada *GCM*, jika belum maka perangkat akan didaftarkan terlebih dahulu agar dapat menerima notifikasi. Pada aplikasi *mobile* ini terdapat dua tab menu yaitu *Home*, *Advokasi*.

Kode Program 1 Mendaftarkan Perangkat Pada *GCM*

```

01 GCMRegistrar.checkDevice(getApplicationContext());
02 GCMRegistrar.checkManifest(getApplicationContext());
03
04 final String RegId =
05     GCMRegistrar.getRegistrationId(getApplicationContext());
06     registerReceiver(mHandleMessageReceiver,
07         new IntentFilter(DISPLAY_MESSAGE_ACTION));

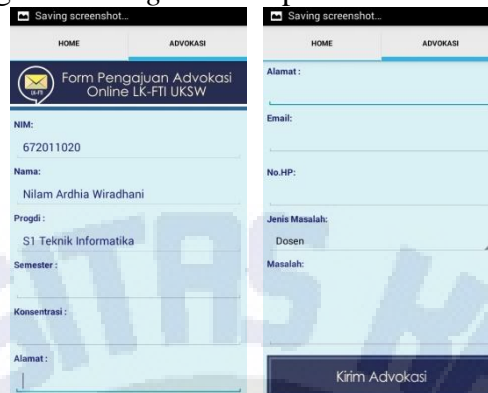
```



Gambar 20 Tab *Home* Aplikasi *Mobile Client*

Gambar 20 menampilkan menu tab *Home* terdapat tombol lihat pesan yang berfungsi untuk pengguna dapat melihat pesan yang dikirim oleh *admin*, dan pengguna dapat berbalas pesan dengan *admin*. Tombol histori pengiriman advokasi berfungsi melihat *history* pengajuan advokasi yang telah dikirim.

Tombol ganti *password* berfungsi mengganti *password* akun advokasi. Sedangkan tombol *logout* berfungsi untuk *logout* akun pada sistem.



Gambar 21 Tab Advokasi Aplikasi *Mobile Client*

Gambar 21 menampilkan menu tab Advokasi pengguna dapat mengirimkan pengajuan advokasi dengan efektif. Dengan cara mengisi formulir pengajuan advokasi yang harus diisi dengan lengkap hal ini bertujuan agar memudahkan dalam proses penyelesaian advokasi yang telah diajukan. Pada formulir advokasi nim, nama, dan prodi ditampilkan otomatis sesuai dengan akun yang berada dalam sistem. Data tersebut diambil dari *database* yang dirubah dalam format JSON agar dapat diakses oleh aplikasi.



Gambar 22 Notifikasi Pesan Aplikasi *Mobile Client*

Gambar 22 menampilkan notifikasi yang dikirim dari aplikasi *web* oleh Fungsionaris BPMF LK-FTI UKSW Bidang Advokasi. Notifikasi merupakan *push* notifikasi yang masuk bersifat *realtime* dan tidak mengurangi kuota pengguna aplikasi. Ketika pengguna membuka notifikasi, aplikasi akan menampilkan obrolan.

Kode Program 2 Membuat *Push notifications* Pada Aplikasi *Mobile*

```

01 private static void generateNotification(Context context, String message) {
02     int icon = R.drawable.ic_launcher;
03     long when = System.currentTimeMillis();
04     NotificationManager notificationManager =
05     (NotificationManager) context.
06     getSystemService(Context.NOTIFICATION_SERVICE);
07     @SuppressWarnings("deprecation")
08     Notification notification = new Notification(icon, message, when);
09     String title = context.getString(R.string.app_name);

```

```

10      Intent notificationIntent = new Intent(context, ViewMessage.class);
11      notificationIntent.putExtra("price", message);
12      notificationIntent.setAction(Intent.ACTION_MAIN);
13      notificationIntent.addCategory(Intent.CATEGORY_LAUNCHER);
14      final int soundResId = R.raw.sound;
15      final String packageName = context.getPackageName();
16      notification.sound = Uri.parse("android.resource://" + packageName
17          + "/" + soundResId);
18      PendingIntent intent = PendingIntent.getActivity(context, 0,
19          notificationIntent, 0);
20      notification.setLatestEventInfo(context, title, message, intent);
21      notification.flags |= Notification.FLAG_AUTO_CANCEL;
22      notificationManager.notify(0, notification);
23      notification.defaults |= Notification.DEFAULT_VIBRATE;
24      long[] vibrate = { 0, 100, 200, 300 };
25      notification.vibrate = vibrate;
26  }
27 }

```

Kode Program 2 merupakan potongan kode untuk membuat dan mengatur *push notifications* yang akan dihasilkan pada perangkat android.

Kode Program 3 Menerima Notifikasi Pada Aplikasi Mobile

```

01 private final BroadcastReceiver mHandleMessageReceiver = new
02 BroadcastReceiver() {
03     @Override
04     public void onReceive(Context context, Intent intent) {
05         String newMessage =
06             intent.getExtras().getString(EXTRA_MESSAGE);
07         WakeLocker.acquire(getApplicationContext());
08         Toast.makeText(getApplicationContext(), newMessage,
09             Toast.LENGTH_LONG).show();
10         WakeLocker.release();
11     }
12 };

```

Kode Program 3 merupakan potongan kode fungsi yang dibutuhkan untuk menerima notifikasi pada perangkat android.

Pengujian aplikasi dilakukan dengan menguji fungsi-fungsi dari aplikasi untuk mencari *bug*/kesalahan yang bertujuan agar aplikasi dapat berjalan sesuai dengan yang diharapkan dan memenuhi kebutuhan pengguna aplikasi. Dalam pengujian aplikasi menggunakan dua teknik yaitu *alpha testing* dan *beta testing*. Pada *alpha testing* dilakukan oleh pembuat aplikasi dan orang-orang yang terlibat dalam pembuatan aplikasi. Berdasarkan *alpha testing* yang dilakukan, dapat ditarik kesimpulan bahwa setiap fungsi yang dibuat sudah berjalan dengan baik dan sesuai yang diharapkan, setiap *input* dari pengguna akan menghasilkan *output* yang sesuai dengan kebutuhan sistem, dan fungsi yang berhubungan dengan pemanfaatan layanan GCM maupun fungsi yang memanfaatkan HTTP *request* dalam mengakses *database* dapat berjalan dengan baik.

Untuk mengetahui sejauh mana aplikasi bermanfaat untuk pengguna yaitu mahasiswa FTI-UKSW digunakan penyebaran kuisisioner sebagai media penilaian pada *beta testing*. Pada kuisisioner ini *sample* pengguna adalah mahasiswa FTI-UKSW yang bermasalah. Pada kuisisioner menggunakan 30 responden dengan variatif responden yaitu mahasiswa yang mengisi kuisisioner pra-penelitian dan mahasiswa yang tidak mengisi kuisisioner pra-penelitian. Responden dipilih secara acak yang mewakili dari setiap progdi pada FTI-UKSW. Hasil analisis data berdasarkan dari kuisisioner ada beberapa pendapat yang memiliki persamaan antara lain : Pertama, sebanyak 93,3% responden mengatakan bahwa aplikasi

pengajuan advokasi *online* ini mudah digunakan. Kedua, sebanyak 96,6% responden mengatakan bahwa tampilan huruf dari aplikasi dapat dilihat dengan jelas. Ketiga, sebanyak 90% responden mengatakan bahwa menu pada aplikasi mudah dipahami. Keempat, sebanyak 90% responden mengatakan bahwa aplikasi ini membantu mempermudah mahasiswa FTI-UKSW mengajukan advokasi, dan mendapatkan informasi tentang perkembangannya. Kelima, sebanyak 90% responden mengatakan bahwa aplikasi ini bermanfaat untuk digunakan mahasiswa dalam pengajuan advokasi.

5. Simpulan

Penelitian ini menghasilkan aplikasi pengajuan advokasi *online* FTI-UKSW memanfaatkan Layanan *Google Cloud Messaging* pada Platform Android yang dapat menyelesaikan faktor penghambat yaitu belum terdapat sistem untuk melayani layanan pengajuan advokasi pada BPMF LK-FTI UKSW. Sistem pengajuan advokasi saat ini masih dilakukan secara manual. Sehingga waktu yang dibutuhkan lebih lama dalam pengolahan data jika dibandingkan dengan menggunakan teknologi serta pertukaran informasi yang terhambat. Dengan menggunakan aplikasi ini dapat menjadikan pengolahan data lebih cepat dan pertukaran informasi menjadi tidak terhambat sehingga layanan advokasi yang bermanfaat dalam menyelesaikan masalah mahasiswa dapat berjalan dengan baik. *Google Cloud Messaging* (GCM) pada penelitian ini digunakan agar mahasiswa FTI-UKSW dan fungsionaris BPMF LK-FTI UKSW bidang advokasi untuk menerima *push* notifikasi pada pertukaran informasi yang berguna ketika setiap terdapat pesan baru. Untuk pengembangan maupun penelitian selanjutnya, ada beberapa saran yang dapat dipertimbangkan yaitu Aplikasi Pengajuan Advokasi *Online* dibangun dapat dikembangkan menjadi *multiplatform*, agar dapat digunakan oleh perangkat lainnya contohnya *Blackberry* dan *i-phone*, untuk fitur obrolan atau berbalas pesan dengan admin pengguna dapat mengirim bukti masalah yang dia alami, Aplikasi *web* yang dipegang oleh admin dalam aplikasi pengajuan advokasi *online* ini dapat dikembangkan dalam bentuk *mobile web*.

6. Pustaka

- [1] Premire. 2009. Short Message Service, <https://premiere.wordpress.com/2009/07/19/sms-gateway/>. Diakses Tanggal 29 April 2015
- [2] Febrian, Chandra. 2013. Penerapan *Google Cloud Messaging Service* Sebagai Pengiriman Pesan Singkat pada Aplikasi Penjualan Batik. Salatiga: Universitas Kristen Satya Wacana.
- [3] Victor Utomo, 2013, Personnel Schedule Distribution Using Google Cloud Messaging, *Teknologi dan Informasi* 4:1-8, <http://www.provisi.ac.id/ejurnal/index.php/JTIKP/article/view/4/1>. Diakses tanggal 3 Maret 2015.

- [4] Google Inc. 2013. *Google Cloud Messaging Overview*. <http://developer.android.com/google/gcm/gcm.html>. Diakses tanggal 28 Februari 2015).
- [5] Gartias, Zapr. 2013. Apa Itu JSON ? Mengenal JSON. <http://www.tuxcorner.net/apa-itu-json-mengenal-json/>. Diakses tanggal 9 Maret 2015).
- [6] Suprianto, Dodit& Agustina, Rini. 2012. *Pemrograman Aplikasi Android*. Malang: MediaKom.
- [7] Muhamad Fardan, 2012, PHP adalah-Hypertext Preprocessor, <http://agiptek.com/index.php/php/101-php.html>. Diakses tanggal 5 Maret 2015.
- [8] Comput-techno.blogspot.com, 2013, Definisi dan Fungsi Mysql, Apache dan PHP, <http://comput-techno.blogspot.com/2013/01/definisi-dan-fungsi-mysql-apache-dan-php.html>. Diakses tanggal 5 Maret 2015.
- [9] Huda, Mifthakul, dan Bunafit Komputer, 2010, *Membuat Aplikasi Database dengan Java MySQL dan Netbeans*, Jakarta : Elex Media Komputindo.
- [10] LK FTI, 2014, Badan Perwakilan Mahasiswa Fakultas. <http://lk-ftiuksw.net/index.php/lk/badan-perwakilan-mahasiswa-fakultas>. Diakses tanggal 6 April 2015.
- [11] Hasibuan, Zainal A. 2007. *Metodologi Penelitian Pada Bidang Ilmu Komputer dan Teknologi Informasi : Konsep, Teknik, dan Aplikasi*. Jakarta : ilmu komputer Univesitas Indonesia.
- [12] Christine, 2014, Mengenal Use Case Diagram. http://www.academia.edu/5295802/Mengenal_Use_Case_Diagram. Diakses tanggal 27 Februari 2015.
- [13] Nugroho, Adi. 2005. *Rational Rose Untuk Pemodelan Berorientasi Objek*. Bandung: Informatika.
- [14] Joko, 2012, Fungsi Dan Pengertian UML http://www.academia.edu/4887559/Fungsi_Dan_Pengertian_UML. Diakses tanggal 27 Mei 2015.