

# DJANGO FRAMEWORK AND PYTHON-GAMMU AS MIDDLEWARE SMS BROADCAST

Puji Winar Cahyo<sup>1</sup>, Arief Ikhwan Wicaksono<sup>2</sup>

Jurusan Teknik Informatika, Teknologi Informasi, Fakultas Teknik dan Teknologi Informasi  
Universitas Jenderal Achmad Yani Yogyakarta  
Jl. Siliwangi, Ringroad Barat, Banyuraden, Gamping, Sleman  
Daerah Istimewa Yogyakarta 55293  
Email : pwcahyo@unjaya.ac.id, ariefikhwan@unjaya.ac.id

## Abstract

*Because the growing of mobile technology in recently days, technology practitioners are required to be able to create a variety of devices and media that support the mobile technology. Application developers compete to create mobile applications that have a variety of features, including video call, voice sending, image sending or text messaging. Developing of mobile applications have an impact on using of Short Message Service, although using of Short Message Service began to shift, but the Short Message Service feature cannot be abandoned, because Short Message Service is a standard feature carried by mobile devices (mobile phones). For this reason, we need to develop Short Message Service Broadcast middleware as an intermediary for Short Message Service communication between web technology and mobile devices. This technology is using for disseminating information between systems that have been built with mobile devices via Short message service. Broadcast system will produce information about new student admissions, That will be received directly by Senior High School Students through their mobile phones. The Middleware can send message broadcast in real time to more than 500 cellular numbers automatically with an average time of 13 seconds for one message. The advantage of using this technology is direct configuration with Python Back-End. Meanwhile, Technology for message inbox still requires further configuration.*

**Keywords:** mobile technology, middleware, sms, broadcast, django.

## 1. Latar Belakang Masalah

Ragamnya komunikasi pada dekade ini mendorong praktisi bidang teknologi untuk menciptakan media yang sanggup menyediakan jalur komunikasi dengan dukungan waktu yang *realtime*. Banyak aplikasi *mobile* yang sudah dikembangkan, diantaranya: whatsapp, telegram, line ataupun komunikasi secara langsung melalui situs media sosial. Dengan dikembangkannya aplikasi *mobile* tersebut, penggunaan fitur SMS (*short message service*) menjadi mulai bergeser (semula merupakan media komunikasi primer, sekarang ini menjadi media komunikasi sekunder).

Meskipun penggunaan fitur SMS sudah mulai bergeser, namun fitur SMS tersebut belum bisa ditinggalkan dikarenakan teknologi SMS merupakan fitur standar yang dimiliki hampir semua tipe *handphone*. Selain itu SMS juga dapat dijadikan sebagai media komunikasi alternatif apabila komunikasi internet tidak tersedia, seperti penerapan *e-payment* di daerah terluar (daerah sulit terjangkau internet), hal tersebut memudahkan konsumen dalam melakukan proses pembayaran, pembelian barang, *top-up* dan transfer saldo [1], yang secara umum sering diterapkan pada transaksi pulsa menggunakan SMS Gateway [2].

Fitur layanan *broadcast* pada teknologi SMS masih sering digunakan, umumnya digunakan sebagai *reminder* atau notifikasi pendukung yang ada pada aplikasi *mobile*, seperti penerapan *middleware* SMS digunakan untuk pengiriman notifikasi pada perangkat *mobile* berupa informasi tingkat level arus pengisian air dan informasi mengenai indikator tangki persediaan air apabila sudah terisi penuh [3]. Penggunaan lain notifikasi SMS juga diterapkan pada pemantauan dan perbaikan jaringan komputer secara internal [4]. Tidak hanya digunakan sebagai *reminder service*, penggunaan *middleware* SMS seringkali diterapkan pada jenis komunikasi *bidirectional*. Seperti pada penggunaan *semi-automated bidirectional SMS platform* pada teknologi *mobile counseling* di Kenya, pengguna SMS dapat bercerita bahkan meminta saran melalui layanan SMS yang telah diterapkan [5].

Penelitian ini melakukan penerapan *middleware* SMS *broadcast* yang digunakan sebagai media penengah *broadcast* antara sistem yang sudah dibangun dengan media komunikasi melalui SMS. Layanan SMS *broadcast* yang dibangun diharapkan dapat mendukung penyebaran SMS ke banyak nomor *handphone* dengan waktu yang sesingkat mungkin. Kasus penerapan dan pengujian kali ini di terapkan pada penyebaran informasi mengenai promosi Penerimaan Mahasiswa Baru di Universitas Jenderal Achmad Yani Yogyakarta.

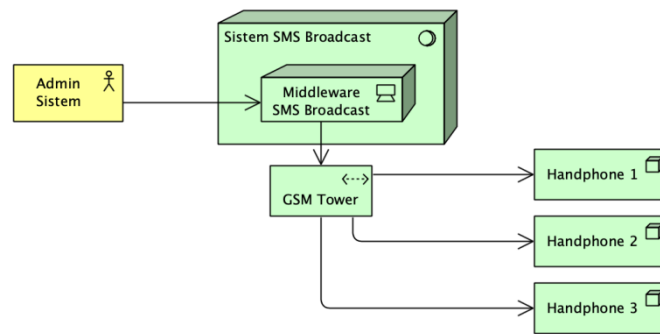
## 2. Metodologi Penelitian

Penelitian ini menggunakan metode riset pengembangan (*research and development* atau R & D) yaitu prosedur penelitian yang bertujuan untuk mengembangkan, menguji kemanfaatan dan efektivitas produk (model) yang dikembangkan, baik produk teknologi, material, organisasi, metode, alat-alat dan sebagainya [6]. Dalam penelitian ini melakukan riset pengembangan cara SMS *broadcast* ke beberapa nomor siswa sekolah tingkat menengah atas dengan menggunakan beberapa pilihan secara khusus, seperti pemilihan berdasarkan asal sekolah, jurusan dan tahun lulus. Hasil dari penelitian ini diharapkan dapat membantu pihak universitas khususnya Universitas Jenderal Achmad Yani Yogyakarta untuk dapat terlibat secara aktif dalam memberikan informasi terbaru kepada siswa sekolah.

### 2.1. Prinsip Kerja Sistem

Prinsip kerja sistem yang dibangun mengakomodasi cara kerja media digunakan sebagai alat komunikasi secara semi otomatis dengan pengguna akhir (*client*) melalui media komunikasi SMS, terlepas dari penyedia telekomunikasi GSM yang menyediakan layanan SMS [7]. *Middleware* yang dibangun tidak lepas dari peranan admin sistem yang terlibat secara aktif mengisi dan menyusun format informasi SMS *broadcast* yang akan dikirim, dalam tahap ini biro promosi fakultas dijadikan sebagai administrator sistem yang bertugas merencanakan dan menyusun format informasi penerimaan mahasiswa baru. Sedangkan *middleware* SMS *broadcast* ada pada sistem SMS *broadcast* yang telah dibangun.

Apabila format dan isian teks SMS sudah tersedia, maka sistem SMS *broadcast* akan melakukan kontak terhadap *middleware* SMS *broadcast* dengan melanjutkan *trigger* pengiriman SMS menuju nomor *handphone* milik siswa yang telah dimasukkan pada sistem. Data SMS yang berisikan informasi mengenai penerimaan mahasiswa baru akan dikirim secara *broadcast* sesuai dengan nomor *handphone* yang sudah dipilih melalui *middleware* SMS *broadcast*. Sehingga penerima akan menerima informasi penerimaan mahasiswa baru secara langsung melalui *handphone* individu masing – masing. Sehingga informasi yang diterima menjadi lebih *realtime* dan dapat segera dilakukan tindak lanjut. Rancangan prinsip kerja yang diterapkan pada *middleware* SMS *broadcast* ditunjukkan sesuai Gambar 1.

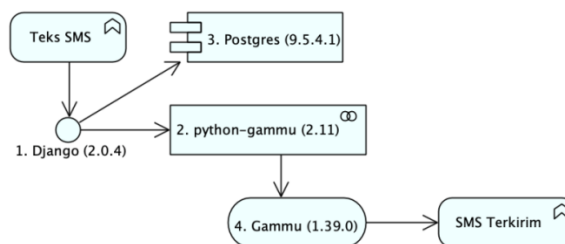


Gambar 1 Prinsip kerja system

## 2.2. Komponen Middleware SMS Broadcast

Middleware SMS *broadcast* terdiri dari 4 komponen *library* yang saling bekerjasama, setiap *library* mempunyai fungsi kerja yang berbeda, Gambar 2 merupakan tahapan bagaimana teks SMS dapat terkirim melalui *middleware SMS broadcast*. Dimulai dari penyiapan format teks SMS, format teks SMS tersebut akan dimasukan melalui sistem yang telah dibangun. Ke-empat *library* tersebut dapat disebutkan sebagai berikut:

1. Django versi 2.0.4, Django merupakan *framework back-end* Python untuk membangun sistem berbasis web. Fungsi *framework* Django pada tahap ini adalah menerima dan meneruskan data teks SMS dan seluruh kontak yang telah dipilih oleh admin menuju *python-gammu* dan Postgres.
2. Python-gammu versi 2.11, *python-gammu* merupakan *interface* yang menjembatani antara bahasa pemrograman Python dengan *Gammu engine service*. Pada tahap ini fungsi dari *python-gammu* adalah menghubungkan data yang diterima dari Django Framework menuju *Gammu engine service* untuk dilakukan penyebaran SMS.
3. Postgres versi 9.5.4.1, Postgres merupakan merupakan *Database Management System* yang dapat menyimpan data secara relasional, sehingga pada tahap ini memudahkan sistem untuk dapat melakukan penyimpanan *log-history* pesan yang pernah dikirim atau diproses melalui Django Framework.
4. Gammu versi 2.11, Gammu merupakan *engine service* yang dapat menangani pengiriman dan penerimaan data teks melalui SMS, pengiriman maupun penerimaan teks tersebut dapat bersifat *long-text*. Pada tahap ini Gammu digunakan untuk mengirimkan data pesan SMS secara *broadcast* sesuai dengan nomor yang telah diseleksi dan dikirimkan oleh *python-gammu*. Sehingga memproses pesan SMS menjadi SMS Terkirim



Gambar 2 Arsitektur Middleware SMS Broadcast

### 3. Hasil dan Pembahasan

Implementasi *middleware* SMS *broadcast* dibagi menjadi 4 tahapan, diantaranya adalah konfigurasi *gammu*, pengembangan kode Python pada Django Framework, aktivasi sistem SMS *broadcast*, pengujian sistem dan *middleware* SMS *broadcast*. Menuju langkah yang lebih detail pembahasan dari ke-empat tahapan tersebut dapat dibahas sebagai berikut.

#### 3.1. Konfigurasi Gammu

Konfigurasi Gammu diperlukan untuk menjamin *middleware* yang telah dibangun dapat terkoneksi dengan perangkat modem yang sudah disediakan, sehingga dapat menangani penyebaran SMS melalui jaringan seluler. Sedangkan nomor yang digunakan untuk *server* SMS *broadcast* ini adalah nomor seluler dengan tipe GSM (*Global System for Mobile Communications*) menggunakan *provider* XL Axiata.

Beberapa konfigurasi harus dilakukan pada Gammu, konfigurasi tersebut disesuaikan dengan modem dan *port* yang digunakan, pada *middleware* ini menggunakan Modem Huawei Model E1820 sehingga penyesuaian pengaturan dapat dilakukan sebagai berikut *device* = `/dev/tty.HUAWEIMobile-Modem` dikarenakan menggunakan modem dengan tipe HUAWEI. Sedangkan koneksi menggunakan *connection* = `at15200`, `at15200` merupakan tipe koneksi modem HUAWEI E1820. Masing – masing tipe koneksi tipe modem, lebih detailnya dapat diakses dihalaman <https://wammu.eu/phones/>.

Selain itu pengaturan pada konfigurasi database juga diperlukan, untuk pengaturan database yang digunakan tersedia pada kolom *smsd service*, karena menggunakan database postgres maka pengaturan dapat dilakukan sebagai berikut *service* = `sql`, *driver* = `native_psql`, Keseluruhan konfigurasi yang dilakukan berada pada file `.gammurc` sehingga dapat dilihat seperti berikut.

```
[gammu]
device = /dev/tty.HUAWEIMobile-Modem
connection = at115200
model = at
synchronizetime = yes
logfile = /Users/pwcahyo/.gammu/log
logformat = errorsdate
gammucoding = utf8

[smsd]
Service = sql
driver = native_psql
PIN = 1234
LogFile = syslog
User = postgres
Password =
PC = localhost
Database = smsd
```

#### 3.2. Pengembangan Kode Python pada Django Framework

Kode Python digunakan untuk mengirimkan teks SMS secara *broadcast (multiple number)*. Kode tersebut disusun didalam sistem yang dibangun menggunakan Django

Framework, dimulai dengan *installing library* python-gammu kemudian dilakukan penyusunan kode sesuai potongan kode berikut.

```

encoded = gammu.EncodeSMS(smsinfo)

for message in encoded:
    # Fill in numbers
    message['SMSC'] = {'Location': 1}
    message['Number'] = data_person[2]

    try:
        # Send SMS if all is OK
        state_machine.SendSMS(message)
        print('{} Success, SMS was Sent'.format(data_person[0]))
        status = MessageStatus.objects.get(id=2) # 2 : Terkirim

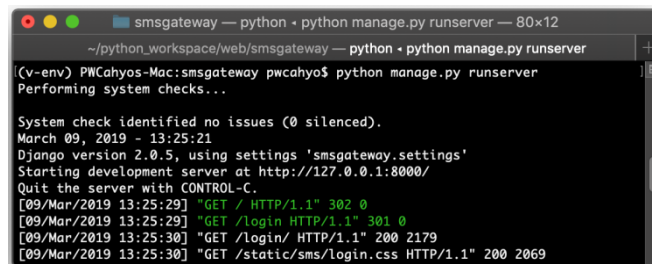
    except gammu.GSMError:
        # Show error if message not sent
        print('{} Error, SMS not Sent'.format(data_person[0]))
        status = MessageStatus.objects.get(id=1) # 1 : Tidak Terkirim

```

Sesuai dengan potongan kode Python tersebut, maka dapat dijelaskan bahwa `gammu.EncodeSMS(smsinfo)` merupakan pemformatan *encoding* yang dilakukan pada *info structure* SMS yang akan dikirim. Sedangkan `message['SMSC'] = {'Location':1}` merupakan *binding SMS Center* dari *simcard* yang akan digunakan untuk pengiriman SMS memanfaatkan *SMS Center* yang ada pada urutan pertama. Kode `message['Number'] = data_person[2]` merupakan isian untuk nomor penerima SMS. Sedangkan proses pengiriman SMS dilakukan didalam kode *try*, apabila kode tersebut berjalan maka dipastikan SMS akan terkirim. Begitu sebaliknya apabila baris kode yang dijalankan adalah baris kode *except*, maka dapat dipastikan proses pengiriman SMS tidak berhasil atau tidak terkirim.

### 3.3. Aktifasi Sistem SMS Broadcast

Aktifasi sistem dilakukan agar *middleware SMS broadcast* dapat berjalan, aktifasi tersebut berpusat pada Django Framework, sehingga untuk aktifasi memerlukan Django *scripting* sebagai berikut: `python manage.py runserver`, kode tersebut dijalankan pada *terminal* atau *console* apabila muncul peringatan *Starting development server at http://127.0.0.1:8000/* sesuai pada Gambar 3 dapat diartikan sistem sudah berjalan secara normal, dilanjutkan dengan mengetikkan alamat url (*Uniform Resource Locator*) `127.0.0.1:8000` pada *web browser* maka sistem siap untuk digunakan.



```

msggateway — python · python manage.py runserver — 80x12
~/python_workspace/web/msggateway — python · python manage.py runserver
(v-env) PwCahyo-Mac:msggateway pwcahyo$ python manage.py runserver
Performing system checks...

System check identified no issues (0 silenced).
March 09, 2019 - 13:25:21
Django version 2.0.5, using settings 'msggateway.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
[09/Mar/2019 13:25:29] "GET / HTTP/1.1" 302 0
[09/Mar/2019 13:25:29] "GET /Login HTTP/1.1" 301 0
[09/Mar/2019 13:25:30] "GET /Login/ HTTP/1.1" 200 2179
[09/Mar/2019 13:25:30] "GET /static/sms/login.css HTTP/1.1" 200 2069

```

Gambar 3 Aktifasi sistem SMS Broadcast

Apabila sistem berhasil diakses maka tampilan *dashboard* seperti ditunjukkan pada Gambar 4, sistem SMS *broadcast* tersebut dapat melayani pengiriman SMS secara *broadcast* berdasarkan biodata siswa SMA (Sekolah Menengah Atas) sebagai berikut: *broadcast* berdasarkan nama sekolah, *broadcast* berdasarkan jurusan, *broadcast* berdasarkan tahun lulus, *broadcast* berdasarkan nama sekolah dan jurusan, *broadcast* berdasarkan nama sekolah dan tahun lulus, *broadcast* berdasarkan jurusan dan tahun lulus dan *broadcast* berdasarkan nama sekolah, jurusan dan tahun lulus.

Gambar 4 Tampilan Sistem SMS Broadcast

### 3.4. Pengujian Sistem dan Middleware SMS Broadcast

Pengujian dilakukan dengan cara *black-box testing*, yaitu berfokus pada fungsionalitas sistem yang tersedia tanpa harus mengetahui secara mendalam alur dari proses fitur yang berjalan. Pengujian *black-box* disempangi pengujian pengembangan perangkat lunak diantaranya dilakukan pada *regression testing*, *acceptance testing*, *unit testing*, *integration testing* dan *system testing stage* [8]. Jenis pengujian yang dilakukan pada tahap ini berfokus pada pengujian *integration testing* yaitu pada integrasi antara sistem dengan *middleware* yang telah dibangun, pengujian ini dilakukan untuk memastikan bahwa integrasi *middleware* berhasil dan dapat dijadikan sebagai sebagai perantara penyebaran SMS.

Dengan melakukan pengiriman sebanyak 689 nomor, *middleware SMS broadcast* dapat mengirimkan 665 SMS terkirim dan 24 nomor tidak terkirim. SMS Terkirim tersebut dapat diartikan bahwa proses pengiriman SMS sudah sampai pada nomor tujuan yang dimaksud, sedangkan tidak terkirim dikarenakan nomor tujuan pengiriman yang salah (penulisan nomor yang tidak benar) atau nomor penerima tidak aktif. Gambar 5 Menunjukkan *history* pengiriman SMS mengenai informasi penerimaan mahasiswa baru di UNJANI (Universitas Jenderal Achmad Yani) kepada siswa SMA yang telah dinyatakan diterima. Dengan dilakukannya intergrasi sistem dan *middleware SMS broadcast* tentunya diharapkan dapat membantu Universitas dalam menyampaikan informasi penerimaan mahasiswa baru dan memudahkan siswa dalam menerima informasi pemberitahuan.

#	Nama / No Handphone	Pesan	Waktu Pengiriman	Status	Sekolah	Jurusan	Tahun Lulus	Aksi
733	085729002401 / Anggie Rezza Julyawati	Selamat! Anggie Rezza Julyawati telah DITERIMA menjadi calon mahasiswa baru Universitas Jenderal Achmad Yani Yogyakarta Info lebih lanjut silahkan bergabung dengan grup chat <a href="https://goo.gl/9Vo9Np">https://goo.gl/9Vo9Np</a> atau menghubungi hotline 085101861996 atau akses langsung ke laman <a href="https://unjaya.ac.id">https://unjaya.ac.id</a>	09 Jun 2018 11:25:07	Terkirim	None	None	2018-7	Delete
732	089675015734 / Rizky Fajar Ramadhan	Selamat! Rizky Fajar Ramadhan telah DITERIMA menjadi calon mahasiswa baru Universitas Jenderal Achmad Yani Yogyakarta Info lebih lanjut silahkan bergabung dengan grup chat <a href="https://goo.gl/9Vo9Np">https://goo.gl/9Vo9Np</a> atau menghubungi hotline 085101861996 atau akses langsung ke laman <a href="https://unjaya.ac.id">https://unjaya.ac.id</a>	09 Jun 2018 11:24:54	Terkirim	None	None	2018-7	Delete
731	085879153354 / Zuhriyi Sabilla Rosyad	Selamat! Zuhriyi Sabilla Rosyad telah DITERIMA menjadi calon mahasiswa baru Universitas Jenderal Achmad Yani Yogyakarta Info lebih lanjut silahkan	09 Jun 2018 11:24:41	Terkirim	None	None	2018-7	Delete

Gambar 5 History pengiriman SMS

#### 4. Kesimpulan

Setelah dilakukan implementasi dan pengujian menggunakan *integration testing*, maka dapat diambil beberapa kesimpulan dari hasil analisis mengenai integrasi sistem dengan *middleware SMS broadcast*. Sistem dengan *middleware SMS broadcast* dapat digunakan dengan baik sebagai media penyebaran SMS. Melalui sistem yang sudah terintegrasi maka dapat diketahui pesan SMS terkirim ataupun tidak terkirim. Pesan terkirim adalah pesan yang berhasil sampai pada nomor *handphone* penerima, sedangkan pesan tidak terkirim teridentifikasi dari kesalahan pengetikan nomor atau nomor penerima sudah tidak aktif. Selain itu, dengan dilakukannya pengujian pada kemampuan pengiriman SMS, *middleware SMS broadcast* dapat mengirimkan SMS lebih dari 500 nomor dengan waktu rata – rata pengiriman 13 detik untuk satu kali SMS, angka tersebut dihitung dari total waktu yang diperlukan dari keseluruhan pengiriman SMS dibagi dengan banyaknya SMS yang terkirim. Dengan memanfaatkan Python-Gammu maka konfigurasi antara *middleware* dengan sistem berbahasa pemrograman Python dapat dijalankan secara langsung. Sementara itu *middleware* yang dibangun hanya melayani SMS *broadcast*, mengenai SMS *inbox* dibutuhkan konfigurasi lebih lanjut.

#### Daftar Pustaka

- [1] Husni, E., & Hidayat, M. A. (2018, July). E-Payment System Using SMS Gateway and Line Application. In *2018 International Conference on Information and Communication Technology for the Muslim World (ICT4M)* (pp. 173-178). IEEE.
- [2] Honggowibowo, A. S. (2013, December). Sistem Informasi Interaktif Pada Transaksi Pulsa Dengan Menggunakan Sms Gateway. In *Conference SENATIK STT Adisutjipto Yogyakarta* (Vol. 1, pp. 120-131).
- [3] Krishna, M. B., & Goyal, D. (2015, February). Mobile middleware application and services for real-time water pump system. In *2015 2nd International Conference on Signal Processing and Integrated Networks (SPIN)* (pp. 411-417). IEEE.
- [4] Hidayatullah, A., & Wintolo, H. (2014). Management Server Berbasis SMS Gateway Studi Kasus pada Gcm Ednovation YOGYAKARTA. *Compiler*, 3(2).
- [5] Perrier, T., Harrington, E. K., Ronen, K., Matemo, D., Kinuthia, J., John-Stewart, G., ... & Unger, J. A. (2018, June). Male Partner Engagement in Family Planning SMS

- Conversations at Kenyan Health Clinics. In *Proceedings of the 1st ACM SIGCAS Conference on Computing and Sustainable Societies* (p. 3). ACM.
- [6] Jaedun, A. (2011). Metodologi Penelitian Eksperimen. *Fakultas Teknik UNY*.
- [7] Taddia, C., & Mazzini, G. (2015, September). Architectures for an efficient SMS Gateway service. In *2015 23rd International Conference on Software, Telecommunications and Computer Networks (SoftCOM)* (pp. 254-258). IEEE.
- [8] Khan, M. E. (2011). Different approaches to white box testing technique for finding errors. *International Journal of Software Engineering and Its Applications*, 5(3), 1-14.