

## IMPLEMENTASI DAN ANALISIS PERFORMANSI MOBICENTS SEBAGAI SERVER APLIKASI PADA ARSITEKTUR IP MULTIMEDIA SUBSISTEM UNTUK LAYANAN VIDEO CONFERENCE

<sup>1</sup>Boby Indradito Priaksa

<sup>2</sup> Dr. Rendy Munadi, Ir.,MT

<sup>3</sup>Iman Hedi Santoso, ST. MT.

<sup>1,2,3</sup>Fakultas Teknik Elektro, Universitas Telkom, Bandung

[bobyindradito@gmail.com](mailto:bobyindradito@gmail.com), [rnd@telkomuniversity.ac.id](mailto:rnd@telkomuniversity.ac.id), [ihs@telkomuniversity.ac.id](mailto:ihs@telkomuniversity.ac.id)

---

### Abstrak

Salah satu layanan telekomunikasi yang berkembang dan menjanjikan adalah komunikasi suara dan video. Kebutuhan pengguna (user) untuk komunikasi suara dan video yang tinggi merupakan sumbangan terbesar bagi perusahaan penyedia layanan telekomunikasi di Indonesia. Kedepannya, komunikasi mengalami kemajuan dengan mengarahkan semua teknologi agar berbasis IP (Internet Protocol). Teknologi ini akan membuat pengguna bisa berhubungan jarak jauh dengan kualitas yang sesuai standard dan harga yang relatif murah. Untuk itu komunikasi suara dan video perlu dikembangkan pada jaringan berbasis IP. Komunikasi berbasis IP sangat dipengaruhi oleh delay, packet loss, dan parameter lainnya. Di sisi lain komunikasi suara dan video harus realtime dan reliable. Hal ini merupakan tantangan bagi pengembangan layanan. Disini akan diuji kelayakan dari layanan suara dan video yang dilewatkan pada jaringan berbasis IP.

Tugas Akhir ini mengimplementasikan layanan video conference pada server OpenIMSCore dengan Mobicents sebagai server aplikasinya. Dari implementasi ini selanjutnya dianalisis dari aspek Quality of service-nya dengan parameter delay, jitter, throughput dan MOS. Analisis dilakukan berdasarkan hasil uji coba pengiriman suara dan video dari server menuju client dengan beberapa klien dan background traffic. Pada tugas akhir ini juga diuji tentang performansi server dengan media wired dan wireless.

Dari pengujian dan analisis diperoleh nilai maksimal one way delay sebesar 19.99367 ms, jitter 0.391403 ms, dan throughput 0.305433 Mbps untuk pengukuran dengan 3 klien. Kemudian untuk background traffic maksimal sebesar 80 Mbps nilai delay 20.00081 ms, jitter 0.185085 ms, dan throughput 0.2635 Mbps. Hasil yang diperoleh masih dibawah batas maksimum yang distandardkan ITU-T dan Cisco, maka disimpulkan sistem ini dapat berfungsi dengan baik. Serta menggunakan media wired akan memberikan hasil performansi lebih baik daripada menggunakan media wireless.

Kata Kunci: Video Conference, SIP, OpenIMSCore Server, Mobicents, Qos

---

### Abstract

*One of telecommunications services that growing and promising are voice and video communications. User requirements (user) for voice and video communications are high for the company is the largest donation telecommunications service provider in Indonesia. Going forward, with the progress of communication technology in order to direct all based on IP (Internet Protocol). This technology will make users can connect remotely to the appropriate quality standards and relatively low prices. For that voice and video communication needs to be developed on an IP-based network. IP-based communication is strongly influenced by the delay, packet loss, and other parameters. On the other hand voice and video communication should be realtime and reliable. It is a challenge for service developers. Here will be tested the feasibility of voice and video services on IP-based network passed.*

*This final project implementing video conferencing services on OpenIMSCore with Mobicents server as the application server. Of implementation is further analyzed from the aspect of its Quality-of-service parameters delay, jitter, throughput and MOS. The analysis was performed based on the results of testing the delivery of voice and video from the server to the client with multiple clients and background traffic. In this final performance was also tested on a server with wired and wireless media.*

*Obtained from the testing and analysis of one-way delay maximum value of 19.99367 ms, 0.391403 ms jitter, and throughput measurements 0.305433 Mbps to 3 clients. Then the background traffic for a maximum of 80 Mbps delay value 20.00081 ms, 0.185085 ms jitter, and throughput 0.2635 Mbps. The results obtained are still below the maximum limit of the standardized ITU-T and Cisco, it is concluded this system to function properly. As well as using a wired medium will give better performance results than using the wireless medium.*

*Keywords: Video Conference, SIP, OpenIMSCore server, Mobicents, QoS*

---

## 1. Pendahuluan

### 1.1 Latar Belakang

Teknologi komunikasi merupakan salah satu faktor terpenting dalam aspek kehidupan baik di bidang ilmu pengetahuan, bisnis, maupun ekonomi.

Kemajuan teknologi yang mendukung membuat komunikasi tidak cukup hanya dengan menggunakan kemampuan analog seperti surat, tetapi saat ini komunikasi sudah berkembang menjadi komunikasi digital. Pengguna (user) pun semakin kreatif dalam hal kebutuhan dan kepuasan untuk berkomunikasi. Pengguna tidak bisa puas hanya dengan bertukar voice atau messaging. Perkembangannya pun sudah sampai pada *video conference*. *Video Conference*, yang merupakan pengembangan dari VoIP yang memiliki kemampuan untuk mengirimkan suara (*voice*) dan gambar bergerak (*video*) secara bersamaan (Ahkam, 2006).

Di sisi lain, perkembangan ini menuntut kebutuhan infrastruktur jaringan yang sangat penting baik itu kebutuhan bandwidth untuk pengiriman suara dan gambar maupun kehandalan sistem dalam melakukan pelayanan untuk *user*. Di sini kita dapat melihat bahwa pelanggan mengharapkan dengan adanya perkembangan teknologi, kualitas layanan juga ikut meningkat. Untuk itulah diharapkan dengan menggunakan *server* aplikasi Mobicents, dapat melayani layanan yang diminta *client* dengan performansi lebih baik yaitu mampu mengurangi waktu yang dibutuhkan untuk eksekusi layanan tersebut (Tsietisi;Terzoli;Wells, 2006).

Selain itu, pengguna juga mengharapkan dapat melakukan komunikasi real time ini dengan gratis tanpa harus berhubungan dengan internet, melainkan intranet. Untuk itulah, pada tugas akhir ini, penulis merancang suatu topologi jaringan data yang terdiri dari perangkat jaringan Video Conference berbasis SIP (Session Initiation Protocol) menggunakan OpenIMSCore server dan Mobicents sebagai server aplikasinya yang ditanamkan pada arsitektur IP Multimedia Subsystem. Diharapkan dengan menggunakan Mobicents sebagai server aplikasinya, hasil analisa QoS yang didapatkan memenuhi rekomendasi dari ITU-T.

## 1.2 Maksud dan Tujuan

Maksud dan tujuan yang ingin dicapai dalam tugas akhir ini adalah merancang dan mengimplementasikan Arsitektur IMS pada server OpenIMSCore dengan Mobicents sebagai server aplikasinya pada layanan video conference.

## 1.3 Rumusan Masalah

Permasalahan dari penelitian ini adalah :

1. Bagaimana cara mengimplementasikan layanan video conference menggunakan server OpenIMSCore?

2. Bagaimana cara mengimplementasikan Mobicents sebagai server aplikasi yang terintegrasi dengan server OpenIMSCore?
3. Bagaimana perbandingan performansi server OpenIMSCore tanpa Mobicents AS dan server OpenIMSCore terintegrasi Mobicents AS dengan jumlah klien berubah pada layanan video conference
4. Bagaimanakah performansi server OpenIMSCore terintegrasi Mobicents AS terhadap perubahan bandwidth dari background trafik pada layanan video conference?
5. Bagaimana performansi server OpenIMSCore terintegrasi Mobicents AS terhadap media wired dan wireless?
6. Bagaimana analisa dan penarikan kesimpulannya?

## 1.4 Batasan Masalah

Dalam penulisan tugas akhir ini, ada beberapa batasan untuk tema yang akan dibahas, yaitu :

1. Implementasi Open IMS dengan sistem operasi Linux Ubuntu 10.04.
2. Sistem tidak memperhitungkan aspek keamanan.
3. Implementasi diadakan di Lab. Switching.
4. Hanya membahas IPv4.
5. Menggunakan Mobicents sebagai server aplikasinya
6. Hanya membahas layanan *Video Conference*.
7. Menggunakan protokol pensinyalan SIP.
8. Performansi yang akan dianalisa adalah parameter-parameter yang menentukan QoS, yaitu: delay, jitter, throughput, dan MOS
9. Klien yang digunakan untuk komunikasi adalah softphone XLite

## 1.5 Metodologi Penelitian

Tugas Akhir ini menggunakan metode penelitian sebagai berikut :

1. Studi literatur  
Berisikan pembahasan teoritis melalui studi literatur dari buku atau jurnal ilmiah yang berkaitan dengan dasar sistem *Video Conference*, IMS, Mobicents AS, dan *software* Open IMS.
2. Implementasi  
Sistem dibuat dengan sistem operasi Linux. Kemudian diimplementasikan beberapa perangkat lunak yang mendukung Open IMS.
3. Analisa sistem  
Sistem diujicobakan pada *closed network*, kemudian dilakukan pengambilan data.

Selanjutnya dilakukan analisa mengenai kemampuan *server* dengan beban permintaan *user*.

**2 Dasar Teori**

**2.1 Konsep Dasar Jaringan NGN**

Jaringan *Internet Protocol* (IP) muncul dan mengalami perkembangan yang sangat cepat dalam dunia komunikasi generasi terkini. Munculnya teknologi jaringan masa depan atau *Next Generation Network* (NGN) merupakan awal dari pemikiran untuk bermigrasi dari teknologi jaringan konvensional yang berbasiskan jaringan *circuit switched* seperti PSTN menuju jaringan berbasiskan sistem *packet switched* dengan menggunakan jaringan IP. Teknologi *softswitch* pada dasarnya digunakan untuk melayani komunikasi *voice* dari jaringan IP atau non IP seperti PSTN, menuju jaringan IP atau sesama non IP dengan melewati suara pada jaringan berbasis *packet switched*. Salah satu aplikasi *softswitch* adalah teknologi VoIP, yang telah muncul pada awal pemikiran konsep NGN. Teknologi *softswitch* ini merupakan teknologi baru sebagai pengembangan VoIP yang dirancang untuk mampu berkembang menuju jaringan NGN dengan proses yang bertahap.

Konvergensi antara jaringan PSTN, PLMN dan jaringan data (khususnya IP) diharapkan dapat mempertemukan tiga kekuatan besar, yaitu layanan *voice* yang menjadi andalan PSTN, *mobility* dan kekayaan layanan yang dimiliki PLMN dan internet-based application (transfer informasi, dan transaksi) yang menjadi kekuatan jaringan IP.

**2.2 Overview IMS (IP Multimedia Subsystem)**

*IP Multimedia Subsystem* (IMS) berfungsi sebagai platform standard untuk layanan multimedia melalui IP/SIP protokol yang memungkinkan operator untuk menggunakan satu *platform* untuk beberapa layanan multimedia. IMS ini merupakan bagian dari standar arsitektur *Next Generation Network* (NGN). Beberapa jaringan (*fixed network, mobile network* atau *wireless network*), dapat dioperasikan layanannya melalui *platform* IMS tentu saja dengan layanan *IP-based* dan didukung protokol SIP.

**2.3 Overview Open IMS**

Open IMS merupakan software yang dibuat oleh FOKUS (sebuah institut di Jerman) pada December 2006. FOKUS mengimplementasikan komponen IMS seperti CSCFs, HSS, *Application Servers* dan lainnya yang terintegrasi dalam suatu sistem tunggal.

Makna kata “Open” berarti *open source* yang diperluas lebih jauh lagi untuk berbagai implementasi, termasuk implementasi jaringan eksisting.

**2.3 Video Conference Overview**

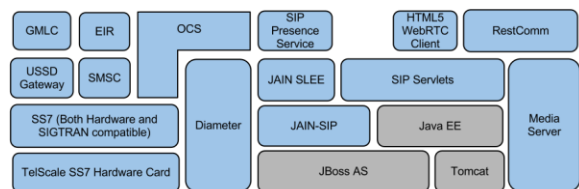
Video conference merupakan salah satu jenis aplikasi multimedia yang dapat menghubungkan beberapa titik secara simultan. Layanan video conference bersifat seketika dengan resolusi yang baik dan interaktif. Pada jaringan digital, pengiriman suara membutuhkan kecepatan sekitar 64 Kbps dan pengiriman video membutuhkan kecepatan 1,5-2 Mbps. Untuk layanan video conference secara keseluruhan akan dibutuhkan kecepatan pengiriman sekitar 9,2 Mbps.

**2.4 Application Server**

Server Aplikasi adalah aplikasi pada sistem komputer yang berfungsi melayani permintaan akses dari komputer pengguna atau klien.

**2.5 Mobicents**

Mobicents adalah sebuah Open Source platform VoIP yang ditulis di Java untuk membantu menciptakan, menyebarkan, mengelola layanan dan aplikasi yang mengintegrasikan suara, video dan data pada jaringan IP.



**Gambar 2.1** Mobicents Platform

**2.6 Jboss AS 7**

JBoss AS adalah server aplikasi yang ditulis oleh JBoss Corp, sekarang dikembangkan oleh Red Hat. JBoss AS ditulis dengan bahasa Java, dan mengimplementasikan pada Java Platform Enterprise Edition (Java EE). JBoss AS bersifat open source dan memenuhi persyaratan dari GNU Lesser General Public License (LGPL), versi 2.1.

JBoss AS 7 dikeluarkan 6 bulan setelah versi sebelumnya, JBoss AS 6. Tidak seperti versi sebelumnya, JBoss AS 7 mendukung spesifikasi Java EE rilis besar terakhir, yaitu Java EE6. Selain itu bersertifikat Web Profile. Perubahan besar terlihat oleh pengguna adalah ketidakmampuan untuk menentukan sumber daya seperti JMS tujuan dan datasources dalam arsip, cara datasources didefinisikan, ukuran yang lebih kecil (kurang dari setengah dari JBoss AS 6) dan penurunan 10 kali lipat dalam waktu startup.

**2.7 Quality of Service**

Parameter QoS menggolongkan kualitas transfer yang diberikan oleh suatu koneksi yang

diperoleh dengan membandingkan unit data pada sisi masukan dan keluaran interface. Parameter QoS adalah :

2.7.1 Delay

Delay adalah waktu tunda suatu paket yang diakibatkan oleh proses transmisi dari satu titik ke titik lain yang menjadi tujuannya. Delay di dalam jaringan dapat digolongkan diantaranya sebagai berikut:

Paketisasi delay, yaitu delay yang disebabkan oleh waktu yang diperlukan untuk proses pembentukan paket IP dari informasi user. Delay ini hanya terjadi sekali saja, yaitu di source informasi.

Delay propagasi, yaitu delay ini merupakan waktu proses perjalanan informasi selama di dalam media transmisi. Interarrival delay, yaitu waktu antar kedatangan paket di penerima. Disebut juga sebagai delta. End-to-end delay, yaitu waktu yang dibutuhkan sebuah paket untuk ditransmisikan dari pengirim ke penerima.

2.7.2 Jitter

Jitter merupakan variasi delay antar paket yang dikirim oleh sumber ke tujuan di dalam media jaringan. Besarnya nilai jitter sangat dipengaruhi besarnya tumbukan antar paket congestion yang ada pada jaringan IP. Semakin besar beban trafik pada jaringan akan menyebabkan semakin besar peluang terjadinya congestion dengan demikian jitter akan semakin besar. Jitter yang dibahas pada Tugas Akhir ini adalah interarrival jitter, yaitu jitter untuk interarrival delay.

2.7.3 Throughput

Dalam komunikasi jaringan, seperti ethernet atau packet radio, atau jaringan, throughput adalah kecepatan (rate) transfer data efektif. Throughput yang biasanya diukur dalam bit per detik (bit/s atau bps), dan kadang-kadang dalam paket data per detik atau data paket per waktu slot. Dengan perhitungan sebagai berikut:

$$\text{Throughput} = \frac{\text{Packet Receive} \times \text{ukuran paket} \times 8}{\text{Total waktu pengiriman}} \text{ (bps)}$$

2.8 Mean Opinion Score (MOS)

Metode ini merupakan metode yang digunakan untuk menentukan kualitas suara dalam jaringan IP berdasar kepada standart ITU-T P.800. Metode ini bersifat subjektif, karena berdasarkan pendapat pendengar. Untuk menentukan nilai MOS terdapat dua cara yaitu, conversation opinion test dan listening test. Rekomendasi nilai ITU-T P.800 untuk nilai MOS adalah sebagai berikut:

**Tabel 2.1** Rekomendasi ITU-T P.800 untuk nilai kualitas berdasarkan MOS

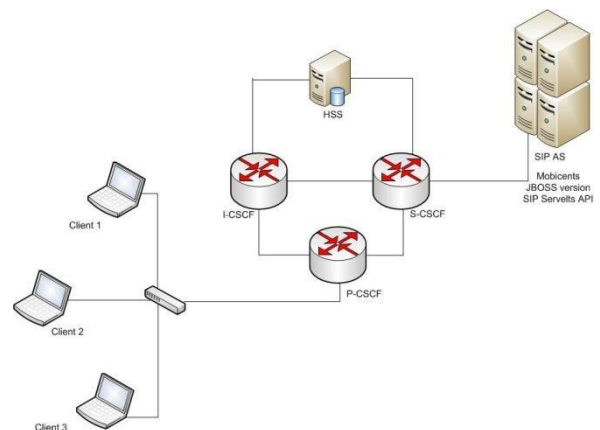
NILAI MOS	OPINI
-----------	-------

5	SANGAT BAIK
4	BAIK
3	CUKUP
2	KURANG
1	BURUK

3. Perancangan dan Implementasi Sistem

Dalam proses perancangan sebuah sistem, diperlukan sebuah skenario yang terstruktur dengan baik. Untuk memudahkan proses perancangan implementasi diperlukan *flowchart* yang membantu dalam memahami proses perancangan yang akan dibuat. Tujuannya adalah agar penelitian lebih terstruktur sehingga proses penelitian akan mengacu pada tujuan akhirnya

Dalam implementasi (di Lab. Switching), komponen IMS yang direalisasikan dibatasi hanya pada *fixed network* dengan 1 *server OpenIMSCore* berikut komponennya HSS dan CSCFs (*proxy, serving dan interrogating*) serta 1 *Mobicents Application Server*. *Switch* digunakan sebagai *switching network* pada jaringan LAN. *PC client* masing-masing memiliki *softphone X-Lite* yang digunakan sebagai *software video converence*.



**Gambar 3.1** Topologi Fisik Implementasi

4 Pengujian dan Analisis Hasil Implementasi

Pada bab ini dibahas mengenai pengujian dan analisis hasil implementasi yang telah dilakukan. Pengujian dan analisis ini bertujuan untuk mengetahui performansi dari server aplikasi Mobicents yang terintegrasi dengan server OpenIMSCore. Seperti yang telah dijelaskan pada bab sebelumnya, Tugas Akhir ini menganalisa faktor penambahan klien, bandwidth trafik, dan media wired wireless dari sistem yang telah dibangun. Beberapa *software* yang

digunakan untuk pengujian dan analisis yaitu wireshark dan iperf.

Skenario pengujian yang digunakan dalam Tugas Akhir ini adalah sebagai berikut:

1. Pengukuran perbandingan performansi pada jumlah *client* berbeda  
Pengukuran ini dilakukan dengan membandingkan antara *server* OpenIMSCore tanpa Mobicents AS dan dengan Mobicents AS sebagai *server* aplikasinya pada jumlah *client* yang berbeda, yaitu 2 *client*, 3 *client*, dan 4 *client*
2. Pengukuran *Background Traffic*  
Pengukuran ini dilakukan pada *server* OpenIMSCore terintegrasi Mobicents AS dengan *background traffic* sebesar 20 Mbps, 40 Mbps, 60 Mbps, dan 80 Mbps
3. Pengukuran *Wired* dan *Wireless*  
Pengukuran ini dilakukan pada *server* OpenIMSCore terintegrasi Mobicents AS menggunakan dua media berbeda, yaitu *Wired* dan *Wireless*

#### 4.2.1 Pengukuran Jumlah Client

##### 4.2.1.1 Delay

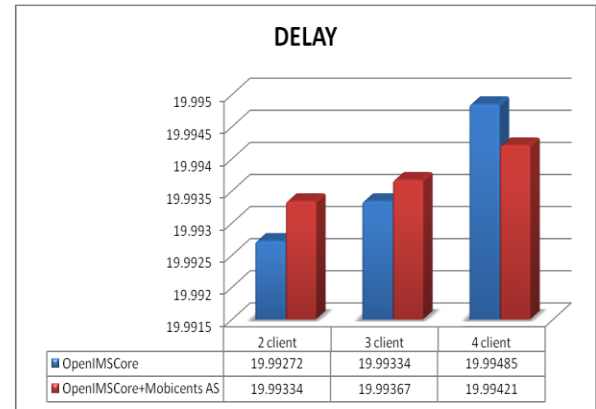
###### a. Tujuan pengukuran

Tujuan pengukuran *delay* adalah untuk mengetahui seberapa stabil jaringan yang dipakai dalam meneruskan paket dari pengirim ke penerima dengan melihat selisih waktu rata – rata paket yang diterima.

###### b. Sistematika Pengukuran

Pengukuran dilaksanakan dengan melakukan komunikasi *video conference* selama satu menit dengan melibatkan jumlah *client* yang berbeda yaitu 2 *client*, 3 *client* dan 4 yang dilakukan 30 kali pengambilan data yang masing-masing selama 1 menit yang kemudian Hasilnya akan dirata – rata. *Delay* yang dibandingkan adalah *delay* pada *server* OpenIMSCore tanpa Mobicents AS dan dengan Mobicents AS

###### c. Hasil Pengukuran



Gambar 4.1 Grafik *Delay* Skenario 1

###### d. Analisis Hasil Pengukuran

Dari hasil pengukuran yang dilakukan pada layanan *video conference* dapat dilihat bahwa ada pengaruh perubahan jumlah *client* baik pada *server* OpenIMSCore tanpa Mobicents AS maupun dengan Mobicents AS. Dari Gambar 4.1 dapat diamati semakin banyak *client* yang mengikuti *conference* maka nilai *one way delay* akan semakin besar.

Pada skenario diatas dapat dilihat bahwa nilai *one way delay* yang didapatkan ketika *client* bertambah, dibandingkan *server* OpenIMSCore tanpa Mobicents AS, *server* OpenIMSCore terintegrasi Mobicents AS memiliki nilai *one way delay* yang meningkat tidak terlalu tajam. Terlihat pada saat 2 *client* nilai *one way delay* sebesar 19.99334 ms dan pada saat 4 *client* 19.99421 ms, selisih 0.00087 ms dibandingkan dengan *server* OpenIMSCore tanpa Mobicents AS saat 2 *client* 19.99272 ms dan pada saat 4 *client* 19.99485 ms, selisih 0.00213 ms. Akan tetapi secara keseluruhan *one way delay* pada kedua *server* menunjukkan hasil yang baik karena masih berada dalam batas yang bisa ditolerir, yaitu tidak lebih dari 150 ms.

##### 4.2.1.2 Jitter

###### a. Tujuan Pengukuran

Tujuan pengukuran adalah untuk mengetahui kestabilan *forwarding* data dalam suatu jaringan. Kestabilan bisa dilihat dari banyaknya variasi *delay* yang terjadi selama waktu komunikasi di jaringan.

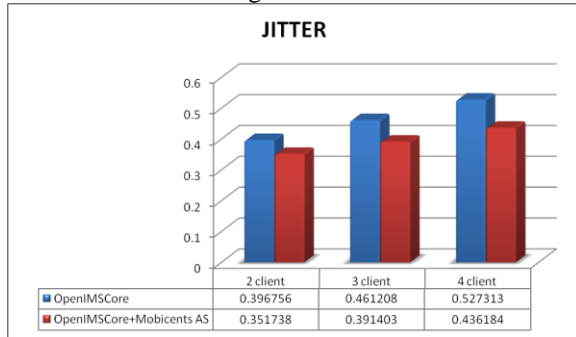
###### b. Sistematika Pengukuran

Pengukuran dilaksanakan dengan melakukan komunikasi *video conference* selama satu menit dengan melibatkan jumlah *client* yang berbeda yaitu 2 *client*, 3 *client* dan 4 *client* yang dilakukan 30 kali pengambilan data yang masing-masing selama 1 menit yang kemudian Hasilnya akan dirata – rata. *jitter* yang dibandingkan



adalah *jitter* pada *server* OpenIMScore tanpa Mobicents AS dan dengan Mobicents AS

c. Hasil Pengukuran



Gambar 4.2 Grafik *Jitter* Skenario 1

d. Analisa Hasil Pengukuran *Jitter*

Dari hasil pengukuran *jitter* yang dilakukan pada layanan *video conference* dapat dilihat bahwa ada pengaruh perubahan jumlah *client* baik pada *server* OpenIMScore tanpa Mobicents AS maupun dengan Mobicents AS yaitu peningkatan nilai *jitter*.

Pada skenario diatas dapat dilihat bahwa *jitter* yang didapatkan ketika *client* bertambah, pada *server* OpenIMScore tanpa Mobicents AS terjadi kenaikan yaitu dari 2 *client* sebesar 0.396756 ms menjadi 4 *client* sebesar 0.527313 ms. Serta pada *server* OpenIMScore terintegrasi Mobicents AS, dari 2 *client* sebesar 0.351738 ms menjadi 4 *client* sebesar 0.436184 ms.

Peningkatan *jitter* yang besar artinya banyaknya variasi *delay* yang terjadi. Semakin banyak variasi *delay* maka keterlambatan kedatangan paket banyak yang berubah-ubah. Untuk itulah akan lebih baik jika memilih nilai *jitter* yang kecil. Dapat dilihat pada Gambar 4.2 nilai *jitter* pada *server* OpenIMScore tanpa Mobicents AS minimalnya sebesar 0.396756 ms dan pada *server* OpenIMScore terintegrasi Mobicents AS sebesar 0.351738 ms.

4.2.1.3 *Throughput*

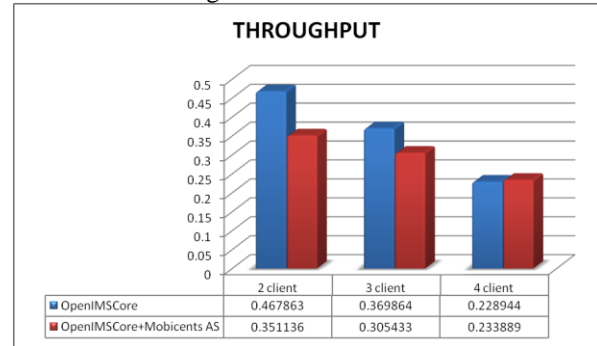
a. Tujuan Pengukuran

Tujuan pengukuran untuk mengetahui kehandalah suatu jaringan dalam melakukan *forwarding* paket data.

b. Sistematika Pengukuran

Pengukuran dilaksanakan dengan melakukan komunikasi *video conference* selama satu menit dengan melibatkan jumlah *client* yang berbeda yaitu 2 *client*, 3 *client* dan 4 yang dilakukan 30 kali pengambilan data yang masing-masing selama 1 menit yang kemudian hasilnya akan dirata – rata. *Delay* yang dibandingkan adalah *delay* pada *server* OpenIMScore tanpa Mobicents AS dan dengan Mobicents AS

c. Hasil Pengukuran



Gambar 4.3 Grafik *Throughput* Skenario 1

d. Analisa Hasil Pengukuran *Throughput*

Dari hasil pengukuran dapat diketahui bahwa baik pada *server* OpenIMScore terintegrasi Mobicents AS maupun tidak terintegrasi Mobicents AS terjadi penurunan nilai *throughput*.

Pada Gambar 4.3 dapat dilihat untuk *server* OpenIMScore tanpa Mobicents AS dengan 2 *client* memiliki nilai *throughput* sebesar 0.467863 Mbps dan ketika dengan 4 *client* menurun menjadi 0.228944 Mbps, begitu pula dengan yang terintegrasi Mobicents AS saat 2 *client* sebesar 0.351136 ms dan saat 4 *client* menjadi 0.233889 ms. Meskipun keduanya sama-sama mengalami penurunan, namun selisih nilai *throughput* pada *server* OpenIMScore terintegrasi Mobicents AS lebih kecil daripada yang tidak terintegrasi Mobicents AS, yaitu selisih 0.11 ms dibanding 0.23 ms dari 2 *client* ke 4 *client*.

*Throughput* merupakan *bandwidth* aktual yang digunakan. Di sisi *client* semakin besar nilai *throughput* semakin banyak data yang sukses terkirim. Untuk itu, nilai *throughput* yang besar akan lebih baik. Dapat dilihat dari hasil pengukuran bahwa *server* OpenIMScore terintegrasi Mobicents AS memiliki nilai selisih yang lebih kecil seiring penambahan jumlah *client*, untuk itulah untuk jangka panjang akan lebih baik menggunakan *server* OpenIMScore terintegrasi Mobicents AS.

4.2.2 Pengukuran *background traffic*

4.2.2.1 *Delay*

a. Tujuan Pengukuran

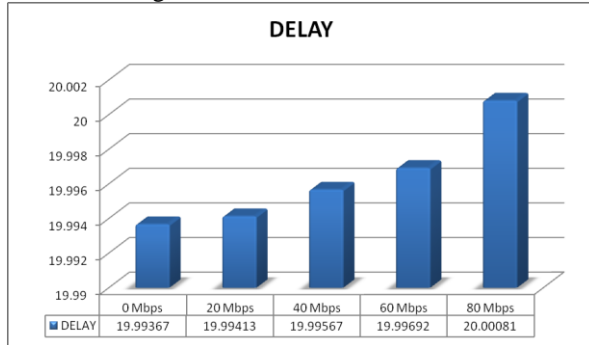
Tujuan pengukuran *delay* adalah untuk mengetahui seberapa stabil jaringan yang dipakai dalam meneruskan paket dari pengirim ke penerima dengan melihat selisih waktu rata – rata paket yang diterima.

b. Sistematika Pengukuran

Pengukuran dilaksanakan dengan melakukan komunikasi *video conference* sebanyak 30 kali

yang masing-masing selama satu menit pada server OpenIMSCore terintegrasi Mobicents AS dengan melibatkan *background traffic* sebesar 0 Mbps, 20 Mbps, 40 Mbps, 60 Mbps, 80 Mbps yang kemudian hasilnya akan dirata – rata.

c. Hasil Pengukuran



Gambar 4.4 Grafik Delay Skenario 2

d. Analisa Hasil Pengukuran Delay

Dari hasil pengukuran yang dilakukan pada layanan *video conference* dapat dilihat bahwa ada pengaruh perubahan besar nilai *bandwidth* dari *background traffic* pada nilai *one way delay* terhadap server OpenIMSCore terintegrasi Mobicents AS. Dapat dilihat dari Gambar 4.4 bahwa semakin besar nilai *bandwidth* yang diberikan, maka nilai *one way delay* akan semakin besar juga.

Pada skenario diatas dapat dilihat bahwa nilai *one way delay* yang didapatkan ketika trafik dipadatkan sebesar 20 Mbps adalah terjadi peningkatan, yaitu dari tanpa *background* trafik sebesar 19.99367 ms menjadi 19.99413 ms. Akan tetapi secara keseluruhan *one way delay* pada server OpenIMSCore terintegrasi Mobicents AS menunjukkan hasil yang baik karena masih berada dalam batas yang bisa ditolerir, yaitu tidak lebih dari 150 ms, yaitu dengan nilai maksimal *one way delay* sebesar 20.00081 ms. Nilai ini diperoleh pada saat nilai *bandwidth* diberikan sebesar 80 Mbps.

4.2.2.2 Jitter

a. Tujuan Pengukuran

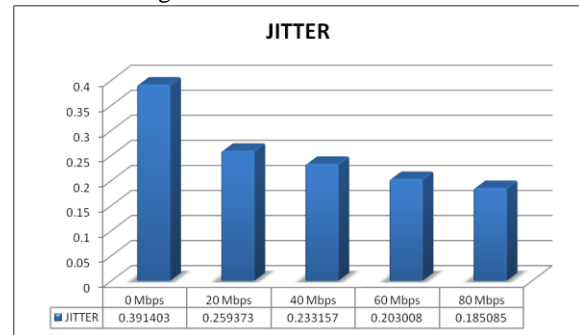
Tujuan pengukuran adalah untuk mengetahui kestabilan *forwarding* data dalam suatu jaringan. Kestabilan bisa dilihat dari banyaknya variasi *delay* yang terjadi selama waktu komunikasi di jaringan

b. Sistematika Pengukuran

Pengukuran dilaksanakan dengan melakukan komunikasi *video conference* sebanyak 30 kali yang masing-masing selama satu menit pada server OpenIMSCore terintegrasi Mobicents AS dengan melibatkan *background traffic* sebesar 0

Mbps, 20 Mbps, 40 Mbps, 60 Mbps, 80 Mbps yang kemudian hasilnya akan dirata – rata.

c. Hasil Pengukuran



Gambar 4.5 Grafik Jitter Skenario 2

d. Analisa Hasil Pengukuran Jitter

Dari hasil pengukuran yang dilakukan pada layanan *video conference* skenario dua, dapat dilihat bahwa ada pengaruh perubahan nilai *bandwidth* trafik pada server OpenIMSCore terintegrasi Mobicents AS. Dari Gambar 4.5 dapat diamati semakin besar nilai *bandwidth*, maka semakin kecil nilai *jitter*-nya.

Pada skenario diatas dapat dilihat bahwa *jitter* yang didapatkan ketika nilai *bandwidth* 0 Mbps berubah menjadi 20 Mbps terjadi penurunan, yaitu dari *jitter* 0.391403 ms menjadi 0.259373 ms.

Penurunan nilai *jitter* artinya sedikitnya variasi *delay* yang terjadi. Semakin sedikit variasi *delay* terjadi maka kedatangan paket semakin sedikit yang berubah-ubah. Server OpenIMSCore terintegrasi Mobicents AS meskipun diberikan trafik, justru memberikan nilai *jitter* yang kecil, sehingga dapat dikatakan pada parameter *jitter* yang dibanjiri trafik, server aplikasi bekerja dengan baik.

4.2.2.3 Throughput

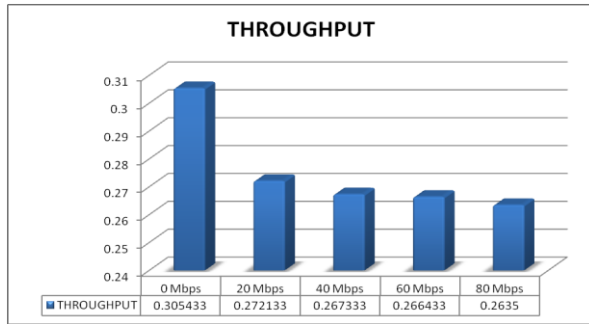
a. Tujuan Pengukuran

Tujuan pengukuran untuk mengetahui kehandalah suatu jaringan dalam melakukan *forwarding* paket data.

b. Sistematika Pengukuran

Pengukuran dilaksanakan dengan melakukan komunikasi *video conference* sebanyak 30 kali yang masing-masing selama satu menit pada server OpenIMSCore terintegrasi Mobicents AS dengan melibatkan *background traffic* sebesar 0 Mbps, 20 Mbps, 40 Mbps, 60 Mbps, dan 80 Mbps yang kemudian hasilnya akan dirata – rata.

c. Hasil Pengukuran



Gambar 4.6 Grafik *Throughput* Skenario 2

d. Analisa Hasil Pengukuran *Throughput*

Dari hasil pengukuran dapat diketahui bahwa nilai *throughput* pada server OpenIMSCore terintegrasi Mobicents AS akan semakin menurun seiring bertambahnya nilai *bandwidth traffic*.

Dari Gambar 4.6 dapat dilihat, bahwa ketika dibebani trafik, nilai *throughput* akan menurun, dalam hal ini saat tanpa dibebani trafik, nilai *throughput* sebesar 0.305433 Mbps dan setelah dibebani trafik sebesar 20 Mbps maka nilai *throughput* menjadi 0.272133 Mbps hingga saat dibebani trafik sebesar 80 Mbps, nilai *throughput* menjadi 0.2635 Mbps.

Dari hasil tersebut dapat dikatakan bahwa semakin besar trafik yang terjadi, maka nilai *throughput* akan semakin menurun yang artinya pada sisi server semakin tidak terbebani karena *bandwidth* aktual yang digunakan tidak semakin besar. Namun pada sisi *client* memiliki kecepatan pengiriman yang kecil.

4.2.3 Pengukuran *Wired* dan *Wireless*

4.2.3.1 *Delay*

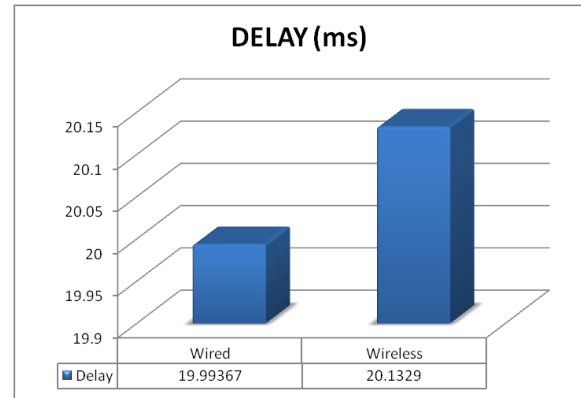
a. Tujuan Pengukuran

Tujuan pengukuran *delay* adalah untuk mengetahui seberapa stabil jaringan yang dipakai dalam meneruskan paket dari pengirim ke penerima dengan melihat selisih waktu rata – rata paket yang diterima

b. Sistematika Pengukuran

Pengukuran dilaksanakan dengan melakukan komunikasi *video conference* sebanyak 30 kali yang masing-masing selama satu menit pada server OpenIMSCore terintegrasi Mobicents AS dengan membandingkan media yang berbeda, yaitu *Wired* dan *Wireless*

c. Hasil Pengukuran



Gambar 4.7 Grafik *Delay* Skenario 3

d. Analisa Hasil Pengukuran *Delay*

Dari hasil pengukuran yang dilakukan pada layanan *video conference* dapat dilihat bahwa ada pengaruh perubahan media dari *wired* ke *wireless* terhadap nilai *one way delay* terhadap server OpenIMSCore terintegrasi Mobicents AS. Dapat dilihat dari Gambar 4.7 bahwa nilai *one way delay* pada media *wired* lebih kecil dibandingkan menggunakan media *wireless*.

Dari skenario diatas dapat dikatakan bahwa nilai *one way delay* yang lebih baik adalah yang menggunakan media *wired*, karena ketika nilai *delay* kecil artinya kedatangan paket memiliki nilai keterlambatan kecil. Dalam hal ini media *wired* memberikan nilai *delay* sebesar 19.99367 ms yang jelas lebih kecil dibandingkan media *wireless* dengan *delay* sebesar 20.1329 ms. Akan tetapi secara keseluruhan *one way delay* pada server OpenIMSCore terintegrasi Mobicents AS menunjukkan hasil yang baik karena masih berada dalam batas yang bisa ditolerir, yaitu tidak lebih dari 150 ms.

4.2.3.2 *Jitter*

a. Tujuan Pengukuran

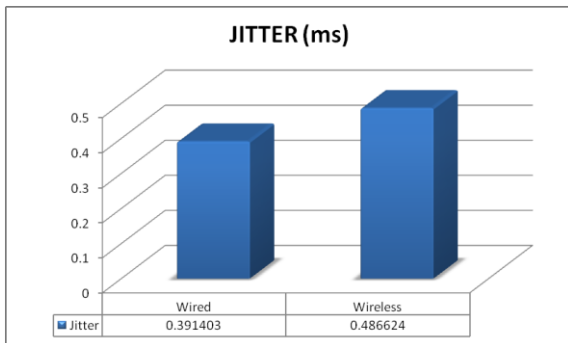
Tujuan pengukuran adalah untuk mengetahui kestabilan *forwarding* data dalam suatu jaringan. Kestabilan bisa dilihat dari banyaknya variasi *delay* yang terjadi selama waktu komunikasi di jaringan

b. Sistematika Pengukuran

Pengukuran dilaksanakan dengan melakukan komunikasi video conference sebanyak 30 kali yang masing-masing selama satu menit pada server OpenIMSCore terintegrasi Mobicents AS dengan membandingkan media yang berbeda, yaitu *Wired* dan *Wireless*

c. Hasil Pengukuran





Gambar 4.8 Grafik *Jitter* Skenario 3

d. Analisa Hasil Pengukuran *Jitter*

Dari hasil pengukuran yang dilakukan pada layanan *video conference* dapat dilihat bahwa ada pengaruh perubahan media dari *wired* ke *wireless* pada *server* OpenIMSCore terintegrasi Mobicents AS. Dari Gambar 4.8 dapat diamati nilai *jitter* pada media *wired* lebih kecil dibandingkan menggunakan media *wireless*.

Dari skenario diatas dapat dikatakan bahwa nilai *jitter* yang lebih baik adalah ketika menggunakan media *wired*, dikarenakan variasi keterlambatan kedatangan paket tidak banyak yang berubah-ubah sehingga nilai *jitter*nya kecil. Untuk itulah akan lebih baik jika memilih nilai *jitter* yang kecil. Pada Gambar 4.8, yaitu pada media *wired* dengan nilai 0.391403 ms daripada media *wireless* dengan nilai 0.486624 ms.

4.2.3.3 *Throughput*

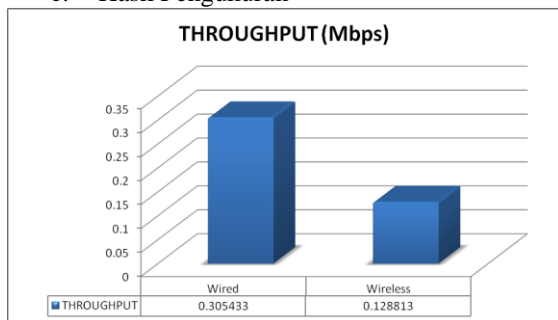
a. Tujuan Pengukuran

Tujuan pengukuran untuk mengetahui kehandalah suatu jaringan dalam melakukan *forwarding* paket data

b. Sistematika Pengukuran

Pengukuran dilaksanakan dengan melakukan komunikasi *video conference* sebanyak 30 kali yang masing-masing selama satu menit pada *server* OpenIMSCore terintegrasi Mobicents AS dengan membandingkan media yang berbeda, yaitu *Wired* dan *Wireless*

c. Hasil Pengukuran



Gambar 4.9 Grafik *Throughput* Skenario 3

d. Analisa Hasil Pengukuran *Throughput*

Dari hasil pengukuran dapat diketahui bahwa nilai *throughput* pada *server* OpenIMSCore terintegrasi Mobicents AS akan menurun jika menggunakan media *wireless* daripada media *wired*.

Dari Gambar 4.9 dapat dikatakan bahwa nilai *throughput* pada media *wired* lebih baik daripada media *wireless*, yaitu pada media *wired* sebesar 0.305433 Mbps dan pada media *wireless* sebesar 0.128813 Mbps. Dikatakan media *wired* lebih baik karena memiliki nilai *throughput* yang lebih tinggi yang artinya adalah pengiriman data yang dilakukan menggunakan *bandwidth* besar sehingga akan cepat sampai tujuan. Untuk itulah akan lebih baik memilih media dengan nilai *throughput* tinggi yaitu media *wired* sebesar 0.305433 Mbps.

4.3 Pengukuran dan Analisis MOS

Metode ini merupakan metode yang digunakan untuk menentukan kualitas suara dan video dalam jaringan IP berdasarkan standart ITU-T P.800.

Metode ini bersifat subyektif, karena berdasarkan pendapat seseorang. Analisis performansi yang dilakukan di sini menggunakan analisis kualitatif, yaitu dengan mendatangkan responden untuk menilai performansi sistem yang telah dibuat dalam hal ini adalah *server* OpenIMSCore terintegrasi Mobicents sebagai *server* aplikasinya. Cara pengukurannya setiap responden melakukan percobaan sebagai *client* dengan menjalankan layanan *video conference*. Sehingga responden dapat mengetahui kekurangan dan kelebihan dari sistem tersebut. Aspek yang dianalisa dalam sistem adalah :

1. Media *Video conference* sebagai *Interface User*

- a. Tampilan media *softphone*, penilaian ini dilakukan agar responden dapat memberikan penilaian terhadap tampilan *softphone xlite* yang menjadi media untuk layanan *video conference*.
- b. Kemudahan menggunakan, hal ini dilakukan untuk mengetahui pendapat user tentang kemudahan untuk menggunakan media *softphone* untuk mengakses layanan *video conference*. Selain itu juga untuk mengetahui seberapa besar tingkat kemudahan untuk mengakses menu-menu yang ditawarkan pada media.

2. Performansi Sistem

- c. Kualitas Audio, untuk mengetahui Kualitas suara yang diHasilkan sistem berdasarkan penilaian responden pada saat konferensi berlangsung.
- d. Kualitas Video, untuk mengetahui Kualitas gambar yang diHasilkan sistem berdasarkan

penilaian responden pada saat pengiriman video berlangsung.

- e. Kualitas Secara Keseluruhan, untuk mengetahui Kualitas sistem secara keseluruhan saat berlangsungnya proses pengiriman layanan *video conference*.
- f. Hasil dari responden tugas akhir ini dapat dilihat pada tabel di bawah.

Tabel 4.3 Nilai masing-masing Aspek

Aspek Penilaian	Nilai					Total	Nilai Rata-rata
	1XN	2XN	3XN	4XN	5XN		
	Buruk	Kurang	Cukup	Baik	Sangat Baik		
Tampilan Media Softphone	0	6	69	16	0	91	3.03
Kemudahan Menggunakan	0	8	66	16	0	90	3.00
Kualitas Audio	0	6	60	28	0	94	3.13
Kualitas Video	0	6	69	16	0	91	3.03
Kualitas Secara Keseluruhan	0	4	72	16	0	92	3.06
Rata-rata Total							3.05

Dari tabel di atas kita dapat mengetahui bahwa beberapa aspek yang dinilai diatas telah cukup memenuhi standart Kualitas yang diharapkan. Semua komponen bernilai lebih dari 3.00 dan rata-rata dari keseluruhan nilai masing-masing aspek adalah 3.05 yang berarti cukup baik menurut standart ITU-T P.800 untuk *MOS*.

**5 Kesimpulan Dan Saran**

**5.1 Kesimpulan**

Berdasarkan dari hasil proses implementasi, pengujian, dan analisis maka dapat ditarik kesimpulan berikut.

1. Implementasi Mobicents sebagai *server* aplikasi pada *server* OpenIMSCore berhasil dilakukan
2. *Server* OpenIMSCore terintegrasi Mobicents AS dibandingkan dengan *server* OpenIMSCore tanpa Mobicents AS dengan penambahan jumlah *client* terbukti memiliki nilai *one way delay* yang sama-sama meningkat, yaitu dari 19.99334 ms ke 19.99421 ms disaat yang tanpa Mobicents AS sebesar 19.99272 ms ke 19.99485 ms, nilai *jitter* juga sama-sama meningkat, dari 0.351738 ms ke 0.436184 ms, disaat yang tanpa Mobicents AS sebesar 0.396756 ms ke 0.527313 ms, dan nilai *throughput* yang sama-sama menurun dari 0.351136 Mbps ke 0.233889 Mbps disaat yang tanpa Mobicents AS sebesar 0.467863 Mbps ke 0.228944 Mbps
3. Menggunakan *background traffic* dengan nilai *bandwidth* dari 20 Mbps ke 80 Mbps, *server* OpenIMSCore terintegrasi Mobicents AS

memiliki *delay* meningkat dari 19.99367 ms ke 20.00081 ms, nilai *jitter* menurun dari 0.341403 ms ke 0.185085 ms, dan nilai *throughput* menurun dari 0.305433 Mbps ke 0.2635 Mbps

4. Menggunakan media *wired*, kualitas komunikasi lebih baik daripada menggunakan media wireless. *Statement* ini dibuktikan dengan pengukuran parameter dari *wired* ke *wireless* yaitu nilai *delay* meningkat dari 19.99367 ms ke 20.1329 ms, nilai *jitter* meningkat dari 0.391403 ms ke 0.486624 ms, dan nilai *throughput* menurun dari 0.305433 Mbps ke 0.2635 Mbps
5. Pada pengukuran kualitas sistem layanan *video conference* dengan menggunakan metode Mean Opinion Score (MOS), didapatkan nilai rata-rata dari setiap aspek penilaian sebesar 3.05 itu cukup baik/fair berdasarkan rekomendasi ITU-T P.800

**5.2 Saran**

Saran yang dapat diajukan untuk penelitian lebih lanjut mengenai topik ini adalah:

1. Perlu diimplementasikan layanan *video conference* yang berbasis IP Multimedia Subsystem menggunakan jaringan akses *mobile IPv6*.
2. Perlu dilakukan penelitian terhadap metode *video conference* lain seperti WebRTC atau Click2Call yang disediakan Mobicents sebagai *Application Server*.
3. Perlu dilakukan percobaan mengintegrasikan Mobicents AS pada arsitektur jaringan lainnya.

**DAFTAR PUSTAKA**

- [1] Ahkam, M. (2006). Analisis Implementation Of IMS Architecture Using Open IMS for Video Conference Service, 3.
- [2] Tsietzi, M., Terzoli, A., & Wells, G. (2006). Mobicents as a Service Creation and Deployment Environment for the Open IMS Core.
- [3] Spiers, R., & Ventura, N. (n.d.). An Evaluation of Architectures for IMS Based Video Conferencing.
- [4] InstallOpenIMSCoreandMobicentsAS - multi-p2p - Multimedia Multicasting over P2P - Google Project Hosting. (n.d.). <https://code.google.com/p/multi-p2p/wiki/InstallOpenIMSCoreandMobicentsAS>
- [5] LLC, R. H. M. (2008). Mobicents Sip Servlets -. Retrieved from [http://www.mobicents.org/mss-diameter\\_sh.html](http://www.mobicents.org/mss-diameter_sh.html)
- [6] Designer, E. S. (n.d.). IMS Conference ( IMS Conference Call ) Calling UE Caller U

- ser IMS 1 Equipment Conference Initiating Initiating Initiator IMS Network IMS 1 MRF MRFP Participating IMS 2 Participating Participating Called UE Called User Equipment Conference Participant This.
- [7] Wikipedia. (n.d.). Mean opinion score. Retrieved from [http://en.wikipedia.org/wiki/Mean\\_opinion\\_score](http://en.wikipedia.org/wiki/Mean_opinion_score)
- [8] L. T. Switching, "MODUL PRAKTIKUM TEKNIK SWITCHING LABORATORIUM TEKNIK SWITCHING," pp. 1–55, 2013.
- [9] Wikipedia, "Mobicents - Wikipedia, the free encyclopedia." [Online]. Available: <http://en.wikipedia.org/wiki/Mobicents>.
- [10] Telestax, "SIP Servlets Server User Guide." [Online]. Available: [https://mobicents.ci.cloudbees.com/job/Mobicents-SipServlets-Release/lastSuccessfulBuild/artifact/documentation/html\\_single/index.html#sfss-The\\_Call-Forwarding\\_Service](https://mobicents.ci.cloudbees.com/job/Mobicents-SipServlets-Release/lastSuccessfulBuild/artifact/documentation/html_single/index.html#sfss-The_Call-Forwarding_Service).
- [11] Jenkins, "Mobicents-SipServlets-Release #547 \_ [Jenkins]." [Online]. Available: <https://mobicents.ci.cloudbees.com/job/Mobicents-SipServlets-Release/lastSuccessfulBuild/artifact/>.
- [12] A. Bayu, "Apa ya Aplikasi Server \_ andibayu." [Online]. Available: <http://andibayu13.blogspot.com/2012/08/apa-ya-aplikasi-server.html>.