



TUGAS AKHIR - KI091391

KOMPRESI VIDEO DINAMIS PADA SISTEM PEMBELAJARAN REAL TIME BERBASIS MOODLE DAN BIGBLUEBUTTON

**Resto Ajie Suyanto
NRP 5109100005**

**Dosen Pembimbing I
Ir. Muchammad Husni, M.Kom.**

**Dosen Pembimbing II
Henning Titi Ciptaningtyas, S.Kom.,
M.Kom.**

**JURUSAN TEKNIK INFORMATIKA
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya 2014**



UNDERGRADUATE THESES - KI091391

***DYNAMIC VIDEO COMPRESSION IN REAL
TIME LEARNING SYSTEM USING MOODLE
AND BIGBLUEBUTTON***

**Resto Ajie Suyanto
NRP 5109100005**

**First Advisor
Ir. Muchammad Husni, M.Kom.**

**Second Advisor
Henning Titi Ciptaningtyas, S.Kom.,
M.Kom**

**DEPARTMENT OF INFORMATICS
Faculty Of Information Technology
Sepuluh Nopember Institute Of Technology
Surabaya 2014**

KOMPRESI VIDEO DINAMIS PADA SISTEM PEMBELAJARAN REAL TIME BERBASIS MOODLE DAN BIGBLUEBUTTON

Nama Mahasiswa : Resto Ajie Suyanto
NRP : 510910005
Jurusan : Teknik Informatika FTIf-ITS
Dosen Pembimbing 1 : Ir. Muchammad Husni, M.Kom.
Dosen Pembimbing 2 : Henning Titi Ciptaningtyas, S.Kom.,
M.Kom.

Abstrak

Saat ini perkembangan teknologi informasi telah tumbuh sangat cepat dan perkembangannya telah memberikan peningkatan positif di seluruh sektor kehidupan, termasuk sektor pendidikan. Kendala dalam sistem pengajaran yang mengharuskan bertatap muka dalam pemberian materi dan penugasan saat ini juga telah dapat diatasi dengan semakin berkembangnya teknologi E-Learning dan streaming multimedia yang memungkinkan satu sama lain dapat bertatap muka melalui notebook atau komputer masing-masing dan berkomunikasi baik secara audio maupun visual dengan menggunakan VoIP. Bahkan teknologi-teknologi tersebut bersifat open source, dimana para pengguna dapat mengembangkan sendiri sesuai dengan kebutuhan.

Tugas akhir ini menggunakan dua perangkat lunak untuk menciptakan suatu sistem pembelajaran online dan real-time, yaitu Moodle dan BigBlueButton. Moodle merupakan sistem perangkat lunak pembelajaran online yang memiliki performa sangat baik untuk memfasilitasi pengajar dan siswa untuk dapat saling berbagi materi dan referensi, berdiskusi, dan melakukan aktivitas-aktivitas kelas secara online. Agar dapat memiliki nilai real-time, maka

Moodle akan diintegrasikan dengan BigBlueButton, yang merupakan perangkat lunak web conference yang memiliki fitur-fitur yang mendukung teknologi VoIP dan untuk menciptakan suatu kelas virtual, seperti berbagi presentasi dengan pengguna lain, kirim pesan antar pengguna, serta berbagi audio dan video dari mikrofon dan kamera web pengguna. Tugas akhir ini juga memfasilitasi pengguna dengan kecepatan akses internet yang rendah untuk tetap dapat berbagi video kamera web yang dimiliki dengan kualitas yang disesuaikan dengan kecepatan akses internet.

Hasil uji coba pada Tugas akhir ini memperlihatkan bahwa suatu pengguna dengan status pengajar dapat mengelola course yang diajar dan melakukan reservasi kelas virtual yang dapat diikuti oleh siswa untuk course yang bersangkutan. Kemudian pengajar atau siswa yang ingin berbagi video dari kamera web, nilai kompresi kamera web yang dibagikan akan menyesuaikan dengan kecepatan akses internet yang dimiliki oleh pengguna. Pada pengujian perubahan nilai kompresi kamera web, sistem berhasil mengubah nilai kompresi kamera web pengguna yang berada pada kecepatan 0-6 Mbps menjadi 50% dari kualitas semula, 6,1-7,5 Mbps menjadi 70% dari kualitas semula, dan lebih dari 7,5 Mbps menjadi 90% dari kualitas semula. Pada pengujian kemampuan server, sistem ini menggunakan 36% dari total memori untuk berjalan dan memacu memori server bekerja hingga 82% untuk menangani 20 pengguna. Sistem menggunakan bandwidth hingga 68 Mbps dalam menangani 20 pengguna untuk menggunakan fitur dalam kelas virtual secara optimal.

Kata kunci: E-Learning, Streaming Multimedia, Voice over IP, Kompresi Dinamis, Open source, Moodle, BigBlueButton

DYNAMIC VIDEO COMPRESSION IN REAL TIME LEARNING SYSTEM USING MOODLE AND BIGBLUEBUTTON

Student's Name : Resto Ajie Suyanto
Student's ID : 5109100005
Department : Informatics FTIf-ITS
First Advisor : Ir. Muchammad Husni, M.Kom.
Second Advisor : Henning Titi Ciptaningtyas, S.Kom.,
M.Kom.

Abstract

Currently, the development of information technology has grown very rapidly . It provides positive improvement in all sectors, includes education . E-learning and multimedia streaming technologies which allow people to communicate each others through notebook or computer bu using VoIP, have been replaced the conservative teaching process, that requires face to face meeting. Moreover, these technologies are open source, where the users can develop according to their own needs.

Two open sources are used in this undergraduated thesis to create a real-time online learning system, such as Moodle and BigBlueButton. Moodle is an open source online learning system which has a very good performance to facilitate teachers and students to share materials and references, discussion, and conduct online classroom activities. In order to provide the real-time process, BigBlueButton is added and integrated into Moodle. BigBlueButton is an open source web conferencing system which supports VoIP technology and create a virtual classroom, such as sharing presentations with other users, send messages between users, as well as sharing audio and video from a microphone and web camera users. In addition, this undergraduated thesis also facilitates the users with low internet access to be able to run their

E-Learning process by automatically adjusting the web camera video quality.

Based on the experimental results, it shows that the teacher are able to manage the course and make reservations for virtual classroom that can be followed by the students who join the course . Then, the teacher or student who wants to share a video from a web camera , will get compression value of web camera based from their own speed of internet access. On the test of a web camera compression value, the system success to adjust the web camera compression value when located at a speed of 0-6 Mbps to 50% from its original quality, 6,1-7,5 Mbps to 70% from its original quality, and more than 7,5 Mbps to 90%from its original quality. In the test of the server capacity, the system uses 36 % of the total memory to run and push the memory of the server which runs up 82 % to handle 20 users. System uses the bandwidth up to 68 Mbps to handle 20 users when using the features of virtual class optimally.

Keywords: E-Learning, Multimedia Streaming, Voice over IP, Dynamic Compression, Open source, Moodle, BigBlueButton

LEMBAR PENGESAHAN

**KOMPRESI VIDEO DINAMIS PADA
SISTEM PEMBELAJARAN *REAL TIME*
BERBASIS MOODLE DAN
BIGBLUEBUTTON**

TUGAS AKHIR

Dizjukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada

Bidang Studi Komputasi Berbasis Jaringan
Program Studi S-1 Jurusan Teknik Informatika
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh

RESTO AJIE SUYANTO
NRP. 5109 100 005

Dibantu oleh Dosen Pembimbing Tugas akhir:

1. Ir. Muchammad Husni, M.Kom.

NP: 196002211984031001

(Pembimbing 1)

2. Henning Titi Ciptaningtyas, S.Kom.,

M.Kom.

NP: 198407082010122004

(Pembimbing 2)

SURABAYA
JUNI, 2014

Kata Pengantar

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Alhamdulillahirabbil'alamin, segala puji bagi Allah SWT, yang telah melimpahkan rahmat dan karunia-Nya sehingga akhirnya penulis dapat menyelesaikan tugas akhir yang berjudul “Kompresi Video Dinamis pada Sistem Pembelajaran Real Time Berbasis Moodle dan BigBlueButton”.

Pengerjaan tugas akhir ini merupakan suatu kesempatan yang sangat berharga bagi penulis, karena dengan pengerjaan tugas akhir ini, penulis bisa memperdalam, meningkatkan, serta mengimplementasikan apa yang telah didapatkan penulis selama menempuh perkuliahan di Teknik Informatika ITS.

Terselesainya buku tugas akhir ini, tidak terlepas dari bantuan dan dukungan semua pihak. Oleh karena itu, penulis ingin menyampaikan rasa terima kasih kepada:

1. Allah SWT atas limpahan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan tugas akhir ini dengan baik.
2. Kedua orang tua penulis, yang selalu memberikan dukungan dan apapun yang penulis butuhkan untuk menyelesaikan tugas akhir ini.
3. Bapak Ir. Muchammad Husni, M.Kom. selaku Dosen Pembimbing 1, yang telah memberikan ide dan topik kepada penulis sehingga lahirlah tugas akhir ini.
4. Ibu Henning Titi Ciptaningtyas, S.Kom.,M.Kom. selaku Dosen Pembimbing 2, yang telah dengan sabar dan penuh perhatian dalam membimbing dan memberikan berbagai macam solusi dalam pengerjaan hingga terselesainya tugas akhir ini.
5. Ibu Nanik Suciati, S.Kom., M.Kom., Dr.Eng. selaku ketua jurusan Teknik Informatika ITS, Ibu Isye Arishanti selaku dosen wali penulis, Bapak Abdul Munif, S.Kom, M.Sc selaku koordinator TA, Bapak Yudhi Purwananto, S.Kom, M.Kom. yang telah memberikan banyak ilmu tentang kehidupan, dan

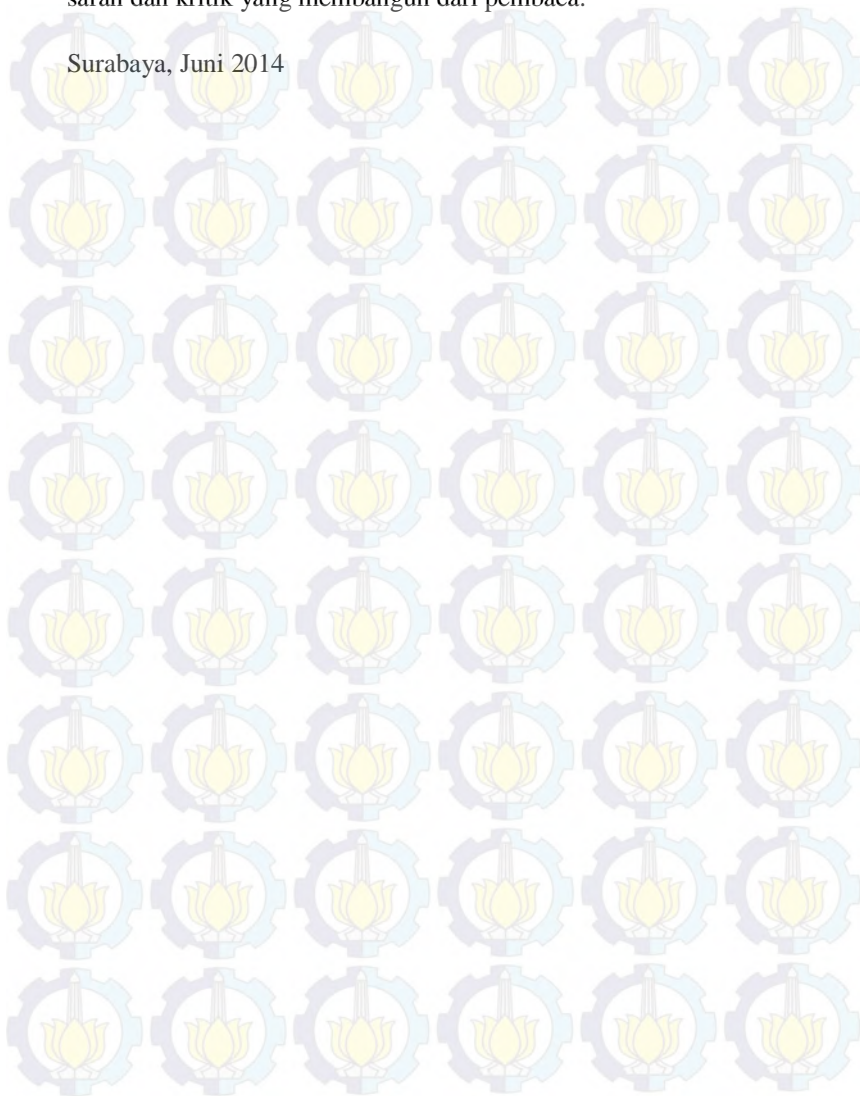
segenap dosen Teknik Informatika yang telah memberikan ilmunya.

6. Pak Yudi, Pak Sugeng, Bu Wartani, Mbak Fatin, Mas Jumali, Mas Murdiono dan segenap staff Tata Usaha yang telah memberikan segala bantuan dan kemudahan kepada penulis selama menjalani kuliah di Teknik Informatika ITS.
7. Nilam Sari Octaviani, yang telah menjadi obat hati dan mengalihkan perhatian penulis dari segala permasalahan hidup untuk dapat tetap fokus menyelesaikan tugas akhir ini.
8. Arya Sinaga dan Rachmat Nafisholeh yang memberikan banyak sekali bantuan dan dukungan kepada penulis dalam proses pengerjaan tugas akhir.
9. Mantan PH Kabinet Perjuangan, Dedy, Fitri, Ceria, Tius, Meta, Om Rizky, Ajeng, Andre, Eka Ayu, Weny, Nur, Rian, Dewi, Bayu, Agnes, Sindung, Rani, Gigih, dan Nurul yang telah berkontribusi dan berjuang bersama penulis di himpunan, perkuliahan serta penyelesaian Tugas akhir.
10. Tim Pemandu Patriot LKMM TM ITS yang luar biasa, Fatma, Gesty, Sani, Rizal, Hanif, Sita, Dewi, Gunadhi, Agus, Sony, Deno, Riza yang telah memberikan semangat dan menjadi inspirasi penulis untuk semakin mengembangkan diri.
11. Anak-anakku yang menginspirasi, forkom Samurai Biru LKMM TM VI FTK, yang selalu mendukung penulis dengan canda dan tawa untuk segera menyelesaikan tugas akhir penulis.
12. Adik-adikku, Farroh, Amik, dan Aik yang mendukung dan memberikan semangat serta keceriaan kepada penulis.
13. Teman-teman angkatan C19 dan 2009 yang sudah bersama-sama menjalani kehidupan kuliah di kampus sejak maba hingga akhir kuliah.
14. Juga tidak lupa kepada semua pihak yang belum sempat disebutkan satu per satu disini yang telah membantu terselesaikannya tugas akhir ini.

Sebagai manusia biasa, penulis menyadari bahwa Tugas akhir ini masih jauh dari kesempurnaan dan memiliki banyak kekurangan.

Sehingga dengan segala kerendahan hati, penulis mengharapkan saran dan kritik yang membangun dari pembaca.

Surabaya, Juni 2014



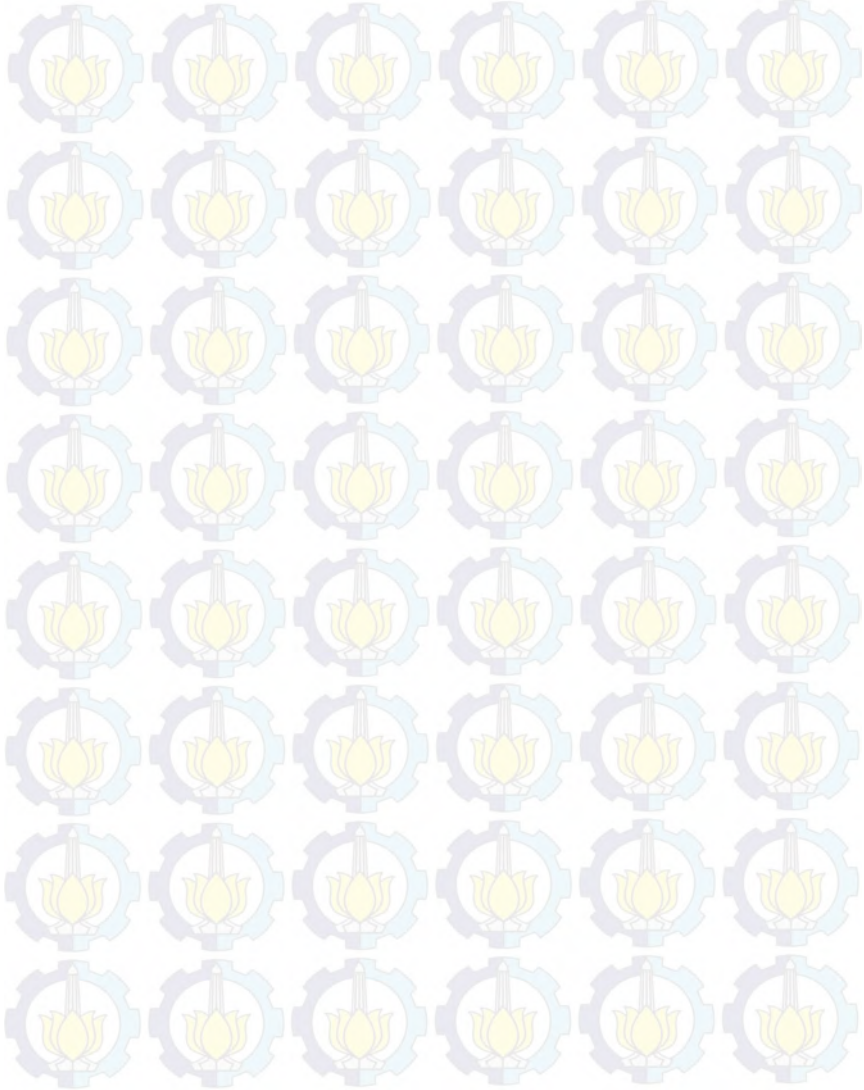
Daftar Isi

LEMBAR PENGESAHAN	v
Abstrak	vii
Abstract	ix
Kata Pengantar	xi
Daftar Isi	xv
Daftar Gambar	xix
Daftar Tabel	xxi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	3
1.4 Tujuan	3
1.5 Metodologi	3
1.6 Sistematika Penulisan	4
2 BAB II	7
DASAR TEORI	7
2.1 LMS (<i>Learning Management System</i>)	7
2.2 <i>Multimedia Streaming</i>	8
2.2.1 Metode Pengiriman dalam <i>Multimedia Streaming</i>	10
2.2.2 Media dalam <i>Streaming</i>	11
2.2.3 <i>Real Time Messanging Protocol (RTMP)</i>	12
2.3 Kompresi <i>Lossless</i>	13
2.3.1 Metode dalam Kompresi <i>Lossless</i>	14
2.3.2 Algoritma Kompresi <i>DEFLATE</i>	15
2.4 <i>Voice over IP (VoIP)</i>	16
2.4.1 Fungsionalitas pada VoIP	17
2.4.2 <i>Signalling Protocol</i> pada VoIP	18
2.5 <i>Asynchronous Javascript and XML (AJAX)</i>	22
2.6 Moodle	23
2.6.1 Konten pada Moodle	26
2.7 BigBlueButton	27
2.7.1 Komponen <i>Open source</i> pada BigBlueButton	28
2.7.2 Arsitektur BigBlueButton	30
3 BAB III	32
PERANCANGAN PERANGKAT LUNAK	33
3.1 Deskripsi Umum Perangkat Lunak	33

3.2	Arsitektur Umum Perangkat Lunak	34
3.3	Diagram <i>Use case</i> Perangkat Lunak	36
3.4	Alur Kerja dan Diagram Alir Perangkat Lunak	37
3.4.1	Alur Kerja dan Diagram Alir Penggunaan Sistem	37
3.4.2	Alur Kerja dan Diagram Alir Permintaan kepada Server BigBlueButton	39
3.4.3	Alur Kerja dan Diagram Alir Perubahan Nilai Kompresi pada BigBlueButton	39
3.5	Perancangan Antar Muka Perangkat Lunak	41
4	BAB IV	43
	IMPLEMENTASI PERANGKAT LUNAK	43
4.1	Lingkungan Implementasi	43
4.1.1	Lingkungan Perangkat Lunak	43
4.1.2	Lingkungan Perangkat Keras	43
4.2	Implementasi Server Moodle	44
4.3	Implementasi Server BigBlueButton	44
4.3.1	Konfigurasi pada Nginx	44
4.3.2	Konfigurasi pada Red5	45
4.3.3	Konfigurasi pada Tomcat6	45
4.3.4	Konfigurasi pada FreeSwitch	45
4.3.5	Konfigurasi Klien BigBlueButton	45
4.4	Implementasi Pengintegrasian Moodle dan BigBlueButton	46
4.5	Implementasi Fitur Perubahan Nilai Kompresi pada Kamera <i>Web</i> BigBlueButton	46
4.5.1	Membaca Kecepatan Akses Pengguna	46
4.5.2	Konversi Nilai Kecepatan Pengguna dengan Nilai Kompresi	47
4.5.3	Pengiriman Nilai Kompresi Menggunakan AJAX	48
4.5.4	Perubahan Nilai Kompresi Menggunakan PHP XML DOM	49
4.6	Implementasi Rancangan Antarmuka	49
5	BAB V	51
	UJI COBA DAN EVALUASI	51
5.1	Lingkungan Uji Coba	51
5.1.1	Lingkungan Perangkat Lunak	51
5.1.2	Lingkungan Perangkat Keras	51
5.2	Uji Coba Fungsionalitas	52
5.2.1	Uji Coba Autentikasi pada Moodle	52

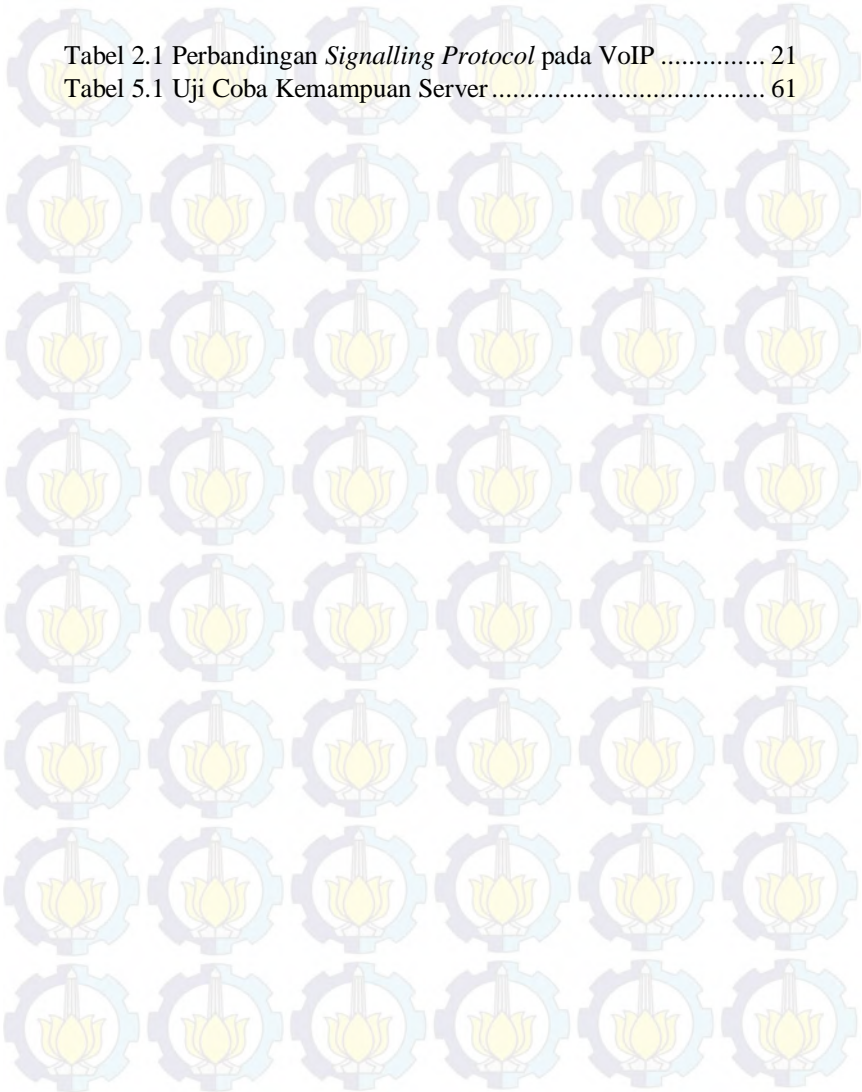
5.2.2	Uji Coba Mengunggah Berkas Pendukung pada Moodle	52
5.2.3	Uji Coba Melakukan Reservasi Kelas Virtual BigBlueButton pada Moodle	53
5.2.4	Uji Coba Mengunggah Berkas Presentasi pada BigBlueButton	54
5.2.5	Uji Coba <i>Editing</i> Berkas Presentasi	55
5.2.6	Uji Coba Menggunakan Fitur Chat BigBlueButton	55
5.2.7	Uji Coba Bergabung Dalam Konferensi Suara	56
5.2.8	Uji Coba Berbagi Kamera <i>Web</i>	57
5.3	Uji Coba Performa	57
5.3.1	Uji Coba Kompresi Dinamis Kamera <i>Web</i>	58
5.3.1.1	Uji Coba Nilai Kompresi Pada Kecepatan Akses Internet 0-6 Mbps	57
5.3.1.2	Uji Coba Nilai Kompresi Pada Kecepatan Akses Internet 6,1-7,5 Mbps	57
5.3.1.3	Uji Coba Nilai Kompresi Pada Kecepatan Akses Internet > 7,5 Mbps	57
5.3.2	Uji Coba Kemampuan Server dalam Menangani Jumlah Pengguna Kelas Virtual	60
5.3.2.1	Uji Coba Kemampuan Server dengan 5 Pengguna	57
5.3.2.2	Uji Coba Kemampuan Server dengan 10 Pengguna	57
5.3.2.3	Uji Coba Kemampuan Server dengan 20 Pengguna	57
5.3.3	Uji Coba Besar Penggunaan Bandwidth pada Kelas Virtual	62
5.3.2.1	Uji Coba Penggunaan Bandwidth dengan 5 Pengguna	61
5.3.2.2	Uji Coba Penggunaan Bandwidth dengan 10 Pengguna	61
5.3.2.3	Uji Coba Penggunaan Bandwidth dengan 20 Pengguna	62
6	BAB VI PENUTUP	65
6.1	Kesimpulan	65
6.2	Saran	66
	DAFTAR PUSTAKA	67

LAMPIRAN	69
BIODATA PENULIS	85



Daftar Tabel

Tabel 2.1 Perbandingan <i>Signalling Protocol</i> pada VoIP	21
Tabel 5.1 Uji Coba Kemampuan Server	61

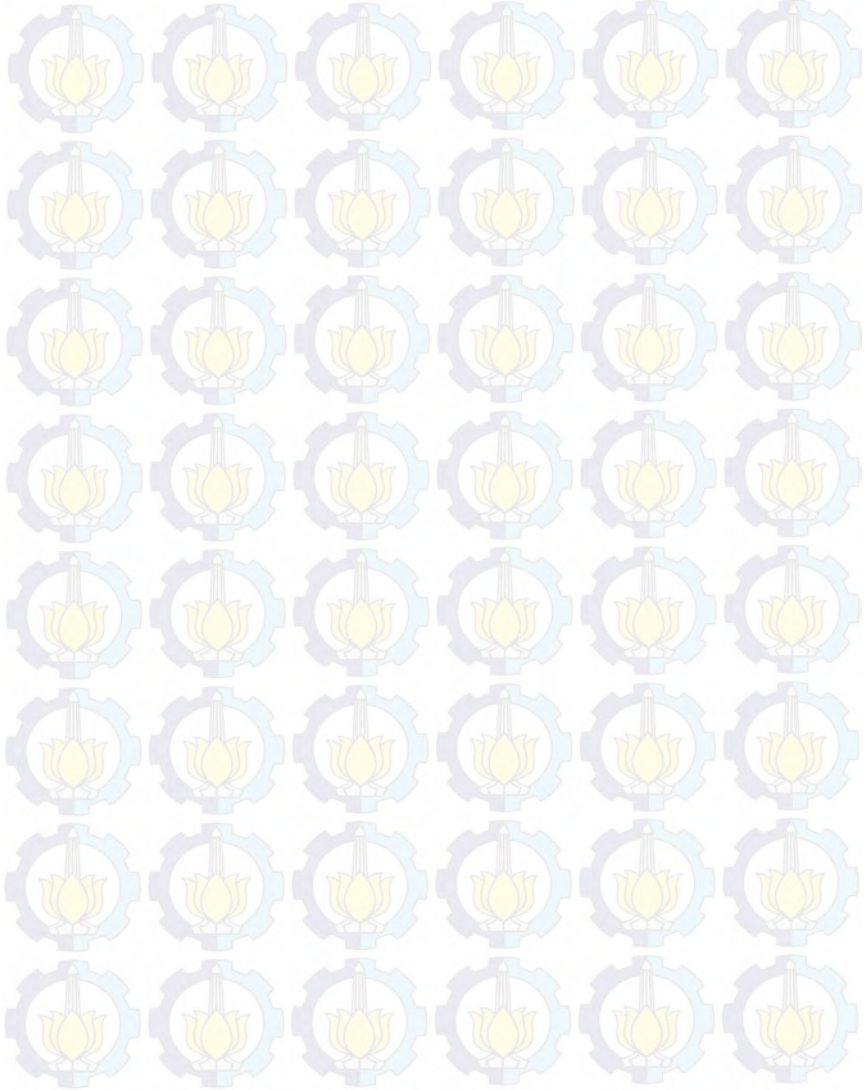
A decorative background pattern consisting of a grid of light blue gears, each containing a yellow lotus flower. The pattern is repeated across the page, with the text of the table of contents overlaid on the top row.

Daftar Gambar

Gambar 2.1 <i>Signalling</i> SIP menggunakan SIP Proxy[6]	20
Gambar 2.2. Alur Kerja AJAX.....	23
Gambar 2.3. Arsitektur BigBlueButton[16].....	30
Gambar 3.1. Arsitektur Umum Perangkat Lunak	35
Gambar 3.2. Diagram <i>Use case</i> Perangkat Lunak.....	36
Gambar 3.3. Diagram Alir Penggunaan Sistem	38
Gambar 3.4 Diagram Alir Permintaan kepada Server BigBlueButton	40
Gambar 3.5. Diagram Alir Perubahan Nilai Kompresi	41
Gambar 3.6 Rancangan Tampilan Antar Muka Reservasi Kelas Virtual.....	42
Gambar 4.1 Fungsi Membaca Kecepatan Akses Pengguna	47
Gambar 4.2 Fungsi Konversi Nilai Kecepatan dengan Nilai Kompresi	47
Gambar 4.3 Grafik Perbandingan Nilai Kompresi dengan Jumlah Penggunaan Bandwidth	48
Gambar 4.4 Fungsi Mengirim Nilai Kompresi dengan AJAX.....	48
Gambar 4.5 Fungsi Perubahan Nilai Kompresi pada Berkas XML	49
Gambar 4.6 Implementasi Rancangan Antarmuka.....	50
Gambar 5.1 Uji Coba Autentikasi Pengguna	52
Gambar 5.2 Uji Coba Mengunggah Berkas	53
Gambar 5.3 Uji Coba Reservasi Kelas Virtual.....	53
Gambar 5.4 Uji Coba Unggah Presentasi	54
Gambar 5.5 Uji Coba Editing pada Berkas Presentasi.....	55
Gambar 5.6 Uji Coba Chat antara Dua Pengguna: (a) Pengguna 1, (b) Pengguna 2	56
Gambar 5.7 Uji Coba Bergabung dengan Konferensi Suara.....	56
Gambar 5.8 Uji Coba Membagikan Kamera <i>Web</i>	57
Gambar 5.9 Uji Coba Kompresi Kamera Web dengan Kecepatan Akses 0-6 Mbps.....	59
Gambar 5.10 Uji Coba Kompresi Kamera Web dengan Kecepatan Akses 6,1-7,5 Mbps.....	59
Gambar 5.11 Uji Coba Kompresi Kamera <i>Web</i> dengan Kecepatan Akses > 7,5 Mbps.....	60

Gambar 5.12 Uji Coba Besar Penggunaan Bandwidth Kelas Virtual

.....63



BAB I

PENDAHULUAN

Pada bab pendahuluan ini, akan dijelaskan tentang hal-hal dasar mengenai tugas akhir. Hal-hal dasar tersebut meliputi latar belakang, rumusan masalah, batasan masalah, tujuan pembuatan Tugas akhir serta metodologi dan sistematika penulisan Tugas akhir. Dengan penjelasan pada bab pendahuluan ini, diharapkan gambaran Tugas akhir secara keseluruhan dapat dimengerti dengan baik.

1.1 Latar Belakang

Tidak dipungkiri lagi, bahwa saat ini kebutuhan akan pendidikan menjadi salah satu kebutuhan pokok bagi masyarakat, tak terkecuali masyarakat Indonesia. Pemerintah Republik Indonesia sendiri telah menerapkan program wajib belajar sembilan tahun untuk memenuhi kebutuhan akan pendidikan tersebut kepada masyarakat Indonesia. Bahkan pemerintah telah menetapkan jenjang-jenjang dalam pendidikan, yakni pendidikan usia dini, dasar, menengah dan tinggi. Dalam penerapannya, metode yang digunakan dalam kurikulum belajar mengajar di semua jenjang menggunakan metode tatap muka karena merupakan metode yang efektif dalam pengajaran. Dalam jenjang pendidikan tinggi, terkadang muncul permasalahan mengenai metode ini, dimana mahasiswa memiliki sesi tatap muka yang tidak sebanyak jenjang pendidikan dibawahnya, sehingga untuk pemahaman terkait materi butuh usaha lebih dan kemandirian dari mahasiswa. Untuk mengatasi hal tersebut, solusi untuk asistensi atau responsi dengan dosen atau asisten dosen yang terkait sering dilakukan, namun penyesuaian jadwal untuk dapat kehadiran sering kali menjadi masalah.

Melihat perkembangan teknologi informasi saat ini yang terus berinovasi, semakin memudahkan antar individu untuk berkomunikasi. Kemudahan dan kecepatan yang disuguhkan oleh teknologi informasi sangat membantu kegiatan masyarakat dalam

berbagai macam sektor kehidupan. Munculnya teknologi *Voice over IP* (VoIP) menjadi salah satu era baru berkomunikasi, yakni bertatap muka dengan posisi antar individu yang berjauhan dalam satu waktu.

Melihat dari potensi ini, tugas akhir ini mencoba memulai suatu gagasan mengenai sistem pembelajaran *online* yang dapat memberikan bahan-bahan belajar kepada mahasiswa, memberikan penugasan kepada mahasiswa, sekaligus melakukan sesi tatap muka, tanpa harus berada dalam kelas pengajaran, namun tetap dapat mawadahi semua mahasiswa, baik yang memiliki kecepatan akses internet yang tinggi maupun yang rendah.

Pada tugas akhir ini, akan digunakan dua macam *open source*, yaitu Moodle sebagai portal untuk masuk ke dalam sistem pembelajaran *online* dan BigBlueButton sebagai penyedia kelas virtual untuk melakukan tatap muka secara online dalam suatu forum. Kedua *open source* ini akan diintegrasikan dan untuk fitur tatap muka dalam kelas virtual akan disesuaikan dengan kecepatan akses internet dari pengguna atau mahasiswa.

1.2 Rumusan Masalah

Rumusan masalah pada tugas akhir ini adalah:

- Bagaimana mengintegrasikan antara Moodle dan BigBlueButton sehingga fitur-fitur pada BigBlueButton dapat digunakan oleh Moodle ?
- Bagaimana membuat fitur tatap muka pada kelas virtual BigBlueButton dapat melakukan kompresi secara dinamis berdasarkan *bandwidth* atau kecepatan akses internet ?

1.3 Batasan Masalah

Dalam pengimplementasian tugas akhir ini, terdapat beberapa batasan dalam pengimplementasiannya, yaitu:

- Menggunakan Moodle minimal versi 2.4.
- Menggunakan BigBlueButton minimal versi 0.81.
- Menggunakan bandwidth minimal 100 Mbps.
- Menggunakan jaringan LAN (*Local Area Network*).
- Pengubahan nilai kompresi hanya pada kamera web untuk streaming video.

1.4 Tujuan

Tugas akhir ini memiliki tujuan dan manfaat yang tertera sebagai berikut:

- Mengintegrasikan antara aplikasi *web* LMS Moodle dengan *open source web conference* BigBlueButton.
- Memberikan fasilitas tatap muka yang fleksibel berdasarkan kecepatan akses internet pengguna.

1.5 Metodologi

Pembuatan tugas akhir ini dilakukan dengan menggunakan metodologi sebagai berikut:

1. Studi literatur
Tahap ini merupakan tahap pemahaman dan pengumpulan informasi dari literatur yang berhubungan dengan penyelesaian tugas akhir. Literatur yang dipelajari dan digunakan meliputi buku dan paper referensi serta dokumentasi internet. Informasi-informasi tersebut nantinya akan digunakan untuk merancang skema sistem secara keseluruhan.

2. Perancangan Sistem

Tahap ini meliputi perancangan sistem berdasarkan studi literatur dan pembelajaran konsep teknologi dari perangkat lunak yang ada. Tahap ini merupakan tahap yang paling penting dimana bentuk awal aplikasi yang akan diimplementasikan didefinisikan. Pada tahapan ini dilakukan perancangan tentang arsitektur sistem, alur kerja sistem, serta tampilan antarmuka untuk implementasi sistem.

3. Implementasi dan pembuatan sistem

Tahap ini merupakan tahap implementasi dari rancangan sistem yang sebelumnya telah dibuat. Tahapan ini membangun server untuk Moodle dan BigBlueButton, lalu mengintegrasikan kedua *open source* tersebut. Setelah itu melakukan modifikasi pada BigBlueButton agar dapat melakukan kompresi secara dinamis berdasarkan pada kecepatan akses internet pengguna.

4. Uji coba dan evaluasi

Pada tahap ini dilakukan uji coba terhadap aplikasi yang telah dibuat. Sistem akan diuji mengenai fungsionalitas dari masing-masing fitur dalam kelas virtual BigBlueButton. Setelah itu akan diuji mengenai performa server kelas virtual BigBlueButton dalam menangani jumlah pengguna.

5. Penyusunan laporan Tugas akhir

Tahap terakhir merupakan penyusunan laporan yang memuat dokumentasi mengenai konsep, dasar teori, hasil implementasi serta uji coba dan evaluasi sistem yang telah dibuat.

1.6 Sistematika Penulisan

1. Bab I. Pendahuluan

Bab ini berisi latar belakang masalah, tujuan dan manfaat pembuatan tugas akhir, permasalahan, batasan masalah, metodologi yang digunakan, dan sistematika penyusunan tugas akhir.

2. Bab II. Tinjauan Pustaka dan Dasar Teori
Bab ini membahas beberapa teori penunjang yang berhubungan dengan pokok pembahasan dan mendasari pembuatan tugas akhir ini.
3. Bab III. Perancangan Perangkat Lunak
Bab ini membahas desain dari sistem yang akan dibuat meliputi : arsitektur, proses dan antarmuka perangkat lunak
4. Bab IV. Implementasi Perangkat Lunak
Bab ini membahas implementasi dari desain sistem yang dilakukan pada tahap desain, disertai dengan potongan kode program yang penting dalam perangkat lunak.
5. Bab V. Uji Coba dan Evaluasi
Bab ini membahas uji coba dari perangkat lunak yang dibuat dengan melihat keluaran yang dihasilkan oleh perangkat lunak dan evaluasi untuk mengetahui kemampuan perangkat lunak.
6. Bab VI. Penutup
Bab ini berisi kesimpulan dari hasil uji coba yang dilakukan serta saran untuk pengembangan perangkat lunak selanjutnya.

BAB II

DASAR TEORI

Pada bab ini akan dibahas mengenai dasar teori yang mendukung penyusunan tugas akhir. Bab ini menjelaskan mengenai *Learning Management System (LMS)*, *Mulimedia Streaming*, Algoritma Kompresi *Lossless*, *Voice over IP (VoIP)*, *AJAX (Asynchronous JavaScript and XML)*, Moodle dan BigBlueButton. Hal ini ditujukan untuk memberikan gambaran secara umum terhadap aplikasi yang akan dibuat, juga berguna untuk menunjang perancangan dan pembuatan aplikasi sehingga sesuai dengan teori yang sudah ada dan berlaku.

2.1 LMS (*Learning Management System*)

Learning Management System (LMS) merupakan sebuah aplikasi perangkat lunak atau teknologi berbasis *web* yang dapat digunakan untuk merencanakan, melaksanakan, dan menilai suatu proses belajar. LMS menyediakan fasilitas kepada para pengajar atau instruktur untuk membuat dan menyebarkan konten pembelajaran kepada siswa, melakukan pemantauan terhadap partisipasi siswa, dan juga menilai kinerja dari siswa. LMS juga menyediakan fasilitas kepada siswa berupa fitur-fitur interaktif seperti percakapan video dan forum diskusi. Saat ini para pengembang LMS mempunyai spesifikasi standar dalam mengembangkan LMS yang disebut *Shareable Content Object Reference Model (SCORM)*. [1]

Seiring perkembangan dan inovasi dalam dunia teknologi informasi hingga saat ini, LMS terus berkembang dan semakin menambah tingkat efisiensi yang dimiliki untuk suatu sistem pembelajaran baik dalam suatu lembaga pembelajaran, perusahaan, maupun organisasi. Banyak manfaat yang dapat dirasakan bagi para pengguna LMS saat ini, seperti:

- Mudah beradaptasi dengan kondisi lembaga dan dapat dikembalikan seperti semula sewaktu-waktu dibutuhkan.
- Punya banyak pilihan yang memudahkan para pengajar atau instruktur dalam metode pemberian konten, mendesain material yang ingin disampaikan, serta teknik-teknik dalam melakukan evaluasi.
- Mengurangi biaya dalam pengembangan dan pemeliharaan sebuah sistem pembelajaran disuatu lembaga.

Meskipun tergolong muda dalam dunia teknologi informasi, LMS hingga saat ini terus berkembang dan beradaptasi sesuai tantangan dan perkembangan teknologi terkini. Konten-konten yang ditawarkan LMS semakin mendukung kebutuhan-kebutuhan saat ini. Integrasi LMS dengan platform-platform perangkat lunak lainnya kini semakin banyak dikembangkan. Media penyimpanan yang sebelumnya masih menggunakan server local, kini sudah banyak yang mulai bermigrasi menggunakan teknologi awan. Berangkat dari bentuk tradisional dalam suatu proses belajar antara pengajar dan siswa dan pengembangan sejumlah konten dan fitur yang mudah untuk dibuat dan dikelola, masa depan LMS akan berjalan dengan dinamis. [2]

Pada tugas akhir ini, LMS digunakan sebagai inti dari sistem yang dibangun, meliputi basis data pengguna dan fitur-fitur yang diperlukan untuk manajemen kebutuhan suatu *course* pembelajaran oleh pengajar dan siswa yang terkait.

2.2 *Multimedia Streaming*

Untuk memberikan kemampuan *real time* pada sistem ini, maka pada Tugas akhir ini memfasilitasi komunikasi antar pengguna yang berbasis *multimedia streaming*. *Multimedia streaming* adalah suatu perlakuan kepada sebuah atau beberapa berkas audio, video, dan multimedia untuk ditransmisikan melalui jaringan internet. Dalam media *streaming* setiap berkas yang ditransmisikan memiliki format yang mendukung dalam transmisi dan dapat dimainkan

secara langsung saat diterima oleh peminta berkas. Sebelum dilakukan transmisi, berkas audio, video, dan multimedia dikompresi terlebih dulu menjadi paket-paket data, kemudian ditransmisikan kepada peminta berkas secara kontinu dan *real time* oleh server.

Dalam sebuah sistem media *streaming*, peminta berkas tidak perlu mengunduh berkas yang diinginkan sampai selesai untuk melihat berkas, dalam sebuah media *streaming* hanya dibutuhkan beberapa detik untuk memulai dekompresi dari paket-paket data yang dikirimkan dan kemudian dapat langsung dilihat sambil menunggu berkas selesai diunduh.[3]

Multimedia *streaming* dapat dibagi menjadi dua jenis, yaitu on-demand dan real-time.

1. *On-demand Streaming*

Pada jenis ini, server *streaming* mentransmisikan berkas audio atau video berdasarkan permintaan yang dilakukan oleh pengguna dan hanya pengguna yang melakukan permintaan tersebut yang akan mendapatkan berkas yang ditransmisikan oleh server *streaming*.

Pada *on-demand*, apabila sistem menerima beban yang berat, transmisi berkas yang dilakukan bisa berhenti sementara hingga dapat mentransmisikan kembali berkas tersebut. Kondisi ini sering dikenal dengan nama *buffering*.[4]

2. *Real time Streaming*

Pada jenis ini, ada 2 jenis pengguna, pengirim dan penerima. Pengirim adalah pengguna yang menentukan berkas audio atau video apa yang akan ditransmisikan dan dilihat oleh penerima atau juga pengirim lainnya yang terhubung dengan *streaming* server.

Pada jenis *real time*, apabila server menerima beban yang berat sehingga kesulitan dalam mentransmisikan berkas, tidak seperti pada on-demand yang berhenti agar dapat mentransmisikan kembali dari titik berhenti, *real time streaming* akan melewati bagian yang terhenti tersebut dan mentransmisikan sisa

berkasnya. Hal ini dikarenakan pada *real time streaming* sudah ditentukan mengenai jumlah aliran berkas dan kecepatan transmisi berkas sehingga *real time streaming* harus menjaga aliran dan kecepatan itu[4].

Tugas akhir ini menggunakan jenis *real time streaming* dalam memfasilitasi interaksi antar pengguna dalam suatu kelas virtual.

2.2.1 Metode Pengiriman dalam Multimedia Streaming

Dalam pengiriman berkas audio atau video, ada beberapa metode yang sering digunakan.

- *Unicast*
Merupakan tipe transmisi data yang membutuhkan sebuah kanal yang menghubungkan Antara pengguna dan *streaming* server. Setiap data yang ditransmisikan oleh *streaming* server hanya akan dikirimkan kepada pengguna yang terhubung dengan kanal data.[3]
- *Multicast*
Merupakan tipe transmisi data yang membutuhkan suatu jaringan yang memiliki kapabilitas bagi setiap routernya untuk menyalin palet data ke banyak kanal data. Server dapat mengirimkan data secara kontinu kepada ribuan pengguna tanpa ada jeda waktu.[3]
- *Broadcast*
Merupakan tipe transmisi dimana pengguna menerima data yang dikirimkan secara pasif. Pasif berarti pengguna hanya dapat melihat, namun tidak dapat melakukan hal apapun, seperti memainkan, memutar ulang, atau menghentikan data yang telah dikirimkan[3].

Tugas akhir ini menggunakan metode *multicast* dalam pengiriman berkas multimedia. Sehingga pada setiap kelas virtual, pengguna mendapatkan semua kamera web pengguna lainnya dalam satu kelas virtual yang diikuti, tetapi tidak dari kelas virtual lainnya.

2.2.2 Media dalam *Streaming*

Untuk melakukan transmisi berkas audio atau video, *streaming* server membutuhkan sebuah media yang dapat menjadi wadah dan memiliki spesifikasi tertentu dalam pentransmisi berkas yang disebut protokol. *Streaming* melibatkan protokol dalam prosesnya pada empat lapisan terbawah OSI *layer* (*Open System Interconnection*)[4]. Dalam sistem ini, juga terjadi proses pengiriman berkas melalui protokol yang melibatkan 4 lapisan OSI *layer* terbawah, yaitu :

- **Lapisan Transport**, agar dapat mendapatkan data dari satu ujung keujung yang lain.
- **Lapisan Sesi**, agar dapat melakukan aktivitas transmisi data.
- **Lapisan Presentasi**, agar dapat menjembatani antara informasi yang terlihat pada aplikasi dengan informasi yang dikirim melalui jaringan.
- **Lapisan Aplikasi**, agar suatu aplikasi dapat terhubung dengan jaringan.

Saat ini kebanyakan aktivitas dalam internet menggunakan protocol TCP (*Transmission Control Protocol*), karena realibilitasnya untuk mendapatkan data yang baik, meskipun harus membutuhkan waktu yang lama apabila data sulit diakses atau diunduh. Namun dalam proses *real time streaming* yang membutuhkan kecepatan dalam waktu transmisi data, maka lebih sering digunakan protokol UDP (*User Datagram Protocol*). UDP lebih ringan jika dibandingkan dengan TCP. UDP akan terus memberikan informasi kepada pengguna meskipun data tersebut

tidak lengkap atau rusak, tidak seperti TCP yang akan melakukan pengiriman ulang hingga data yang dikirimkan sempurna.[4]

Hingga saat ini telah dikembangkan protokol-protokol lain yang mendukung *real time streaming* berdasarkan UDP, seperti:

- The RTP Family
Real Time Transport Protocol (RTP) merupakan protokol yang paling lama dan paling banyak di temui dalam multimedia *streaming*. Beberapa pengembangannya adalah RTCP dan RTSP.
- RTMP
Real Time Messaging Protocol (RTMP) merupakan protokol *real time* khusus yang dikembangkan oleh Flash, namun dapat diimplementasikan pada perangkat lunak lainnya.
- HLS (*HTTP Live Streaming*)
HLS merupakan protokol yang dikembangkan dan digunakan oleh Apple untuk iOS dan tidak dapat digunakan oleh perangkat lunak lainnya.
- Microsoft *Smooth Streaming*
Smooth Streaming merupakan protokol yang dikembangkan oleh Microsoft dengan menggunakan IIS dan *Silverlight*.

RTMP digunakan sebagai protokol dalam sistem ini karena sistem menggunakan Flash sebagai inti dari fitur *web conference*.

2.2.3 *Real Time Messanging Protocol* (RTMP)

Real Time Messanging Protocol (RTMP) digunakan untuk mentransmisikan audio, video, dan *meta-data* melalui suatu jaringan. Protokol ini dikembangkan dengan tujuan agar dapat memfasilitasi transmisi data dari server ke pengguna tanpa menggunakan aplikasi lainnya selain peramban pengguna dan *flash player plugin*. RTMP melakukan pemecahan data menjadi paket-

paket, lalu mentransmisikannya secara dua arah, sehingga setiap pengguna dapat mengatur data yang diterimanya maupun yang dikirimkan. Tidak hanya berupa berkas saja, RTMP juga memfasilitasi transmisi *meta-data* dan melakukan *remote client*. [5]

Meskipun demikian, RTMP memiliki beberapa kelemahan, seperti enkapsulasi keamanan yang kurang baik. Tetapi kelemahan tersebut dapat di atasi dengan beberapa protokol turunan dari RTMP, seperti :

- RTMPS, yang bekerja melalui koneksi sekuritas TSL/SSL.
- RTMPE yang bekerja menggunakan enkripsi dengan Adobe's security algorithm.
- RTMPT yang menggunakan HTTP sebagai protokol transport sehingga dapat melakukan HTTP *tunneling*. Pada sistem ini menggunakan RTMPT saat klien melakukan tunneling.
- RTMFP merupakan variasi dari *peer-to-peer* protokol dan menggunakan UDP sehingga dapat mentransmisikan data tanpa menggunakan komunikasi *server-side*.

Dalam penerapannya, RTMP sering digunakan dalam membangun aplikasi *video chat* atau *video conferences*. RTMP juga sering digunakan dalam untuk *streaming* langsung, sehingga pengguna dapat melihat berkas video yang sedang direkam oleh pengguna lain dan begitu pula sebaliknya.

Pada tugas akhir ini, RTMP digunakan untuk mengubah kualitas video sesuai dengan *bandwidth* jaringan yang berubah-ubah secara otomatis.

2.3 Kompresi *Lossless*

Algoritma kompresi *Lossless* adalah algoritma yang memungkinkan rekonstruksi berkas yang telak dikompresi, menjadi bentuk semula. Hal ini dapat dibandingkan dengan kompresi data

lossy, yang tidak memungkinkan data asli untuk direkonstruksi dari data terkompresi. Kompresi data lossless digunakan dalam berbagai aplikasi. Sebagai contoh, digunakan dalam format berkas ZIP yang populer. Hal ini juga sering digunakan sebagai komponen dalam teknologi kompresi lossy data [20].

Kompresi *lossless* digunakan ketika data asli dan data didekompresi harus identik. Contoh umum adalah program *executable* dan kode sumber. Beberapa format berkas gambar, terutama PNG, menggunakan kompresi *lossless*, sementara yang lain seperti TIFF dan MNG dapat menggunakan salah satu *lossless* atau metode *lossy*. GIF menggunakan metode kompresi *lossless*, tetapi kebanyakan implementasi GIF tidak mampu mewakili penuh warna, sehingga mereka quantize gambar (sering dengan dithering) 256 atau lebih sedikit warna sebelum encoding sebagai GIF [20].

Dalam tugas akhir ini, digunakan kompresi *lossless* untuk melakukan *streaming* video antar pengguna dalam sebuah kelas virtual, dimana dalam penerapannya, kompresi akan bersifat dinamis mengikuti kecepatan akses internet dari pengguna.

2.3.1 Metode dalam Kompresi *Lossless*

Metode kompresi *lossless* dapat dikategorikan sesuai dengan jenis data yang dikompresi. Beberapa jenis data untuk algoritma kompresi seperti teks, *executable*, gambar, dan suara. Pada prinsipnya, setiap algoritma kompresi lossless dapat digunakan pada semua jenis data, meskipun tidak semua data mampu mencapai kompresi yang signifikan dengan menggunakan kompresi lossless. Data suara, misalnya, tidak dapat dikompresi baik dengan algoritma kompresi lossless [20].

Kebanyakan program kompresi *lossless* menggunakan dua jenis algoritma: satu yang menghasilkan model statistik untuk input data, dan satu lagi yang memetakan data input ke *bit string*. Seringkali, hanya algoritma pertama yang sering disebutkan, sedangkan yang kedua hanya secara tersirat (melalui penggunaan umum, standarisasi dll) atau tidak ditentukan [20].

Ada beberapa algoritma pemodelan statistik untuk teks, seperti Burrows-Wheeler Transform, LZ77, LZW, PPM. Sedangkan algoritma encoding untuk menghasilkan urutan bit, meliputi Huffman Coding dan Arithmetic Coding [20].

Tugas akhir ini menggunakan gabungan algoritma kompresi *lossless* antara Huffman Coding dan LZ77, yang disebut algoritma Deflate.

2.3.2 Algoritma Kompresi *DEFLATE*

Algoritma Deflate adalah *lossless compression algorithm* yang mengkombinasikan antara algoritma *Huffman Code* dan LZ77, dimana mengadopsi teknik *sliding-window* pada LZ77 dan *prefix-tree* pada *Huffman Code* [19]. Algoritma *Deflate* tergolong algoritma yang fleksibel dalam menentukan bagaimana proses kompresi dilakukan. Ini yang menjadi keunikan pada algoritma *Deflate*, dimana kompresor pada algoritma ini yang menentukan mode apa yang digunakan untuk melakukan kompresi pada suatu data [18]. Ada tiga mode kompresi pada algoritma *Deflate*, yaitu :

- **Mode 1** - Tidak dikompresi sama sekali. Mode ini cocok digunakan untuk data yang sudah melewati proses kompresi. Data yang dikompresi menggunakan mode ini akan sedikit membesar, namun tidak sebanyak jika data tersebut sudah dikompresi. Kompresi dengan mode ini adalah kompresi yang paling mudah dilakukan karena pada dasarnya input dapat langsung diberikan ke output [18].
- **Mode 2** - Kompresi, diawali dengan LZ77 dan diteruskan dengan pengkodean *Huffman*. Pohon yang digunakan pada mode ini ditentukan oleh spesifikasi Deflate itu sendiri. Pohon *Huffman* pada mode ini disebut juga pohon *Huffman* statik. Menggunakan pohon Huffman statik berarti tidak memerlukan ruang ekstra untuk menyimpan pohon tersebut. Namun

dengan menggunakan pohon Huffman statik kemungkinan besar kompresi yang dilakukan tidaklah optimal [18].

- **Mode 3** - Kompresi, diawali dengan LZ77 dan dilanjutkan dengan pengkodean *Huffman*. Perbedaan mode ini dengan mode pada poin 2 adalah, pada mode ini pohon *Huffman* yang digunakan dirancang dan dibuat oleh kompresor dan disimpan bersama data-data yang dikompresi. Pohon *Huffman* ini disebut juga pohon *Huffman* dinamik. Dengan menggunakan mode ini berarti kita perlu menentukan karakter yang digunakan untuk kompresi blok. Karakter ini kemudian disimpan sebelum data dan bersifat tetap dalam artian tidak dapat diubah lagi [18].

Pada kompresi *Deflate*, data input kompresi dipecahkan dalam blok-blok. Tiap blok membuat header 3 bit di mana bit pertama adalah penanda blok terakhir pada keseluruhan data, 1 menandakan blok tersebut adalah blok terakhir, dan 0 menandakan masih ada blok lain yang harus diproses setelah blok ini, dan 2 bit terakhir pada header untuk menentukan mode kompresi yang dipakai pada blok ini dengan 00 menandakan penggunaan mode 1, 01 penggunaan mode 2, 10 untuk penggunaan mode 3 [18].

Adobe Flash Player pada versi 10, yang digunakan pada BigBlueButton, telah menggunakan perangkat lunak Zlib untuk melakukan kompresi pada *streaming* video. Zlib sendiri mengimplementasikan algoritma *Deflate* dalam melakukan kompresi data, sehingga pada aplikasi perangkat lunak ini menggunakan algoritma *Deflate* untuk melakukan kompresi pada *streaming* video.

2.4 Voice over IP (VoIP)

VoIP adalah teknologi yang memungkinkan komunikasi suara melalui jaringan komputer seperti Internet. VoIP mengubah sinyal suara analog menjadi paket data digital secara real time, serta

transmisi percakapan dua arah menggunakan Internet Protocol (IP). VoIP pada sistem ini digunakan untuk memberikan fasilitas kepada pengguna untuk dapat terhubung dalam suatu ruang konferensi suara, baik melalui jaringan internet atau PSTN (*Public Switched Telephone Network*).

Panggilan VoIP dapat dibuat di internet menggunakan layanan VoIP dan sistem komputer audio standard, atau beberapa penyedia layanan dukungan VoIP melalui telepon biasa yang menggunakan adaptor khusus untuk terhubung ke jaringan komputer rumah.

2.4.1 Fungsionalitas pada VoIP

VoIP harus memiliki beberapa fungsionalitas dasar sebagai berikut:

- **Signaling**, adalah kemampuan membuat atau bertukar informasi “*control*” yang digunakan untuk membentuk, memonitor, atau melepas koneksi antara dua *endpoint*. [6]

Ada 2 jenis *signaling* pada VoIP :

- o **Peer-to-Peer Signaling** : *end devices / gateway* bisa menginisiasi dan memutuskan panggilan dan menginterpretasikan *control message*.
- o **Client Server** : *endpoints / gateway* tidak memiliki kemampuan *call control*, sehingga semua notifikasi *event* dikirimkan dan diterima oleh server yang disebut dengan *call agent*.

Pada tugas akhir ini, digunakan Peer-to-Peer Signaling untuk melakukan koneksi VoIP.

- **Database services**, menyediakan akses layanan, seperti nomor *toll-free* atau *caller ID* memerlukan kemampuan database untuk mengetahui kapan panggilan ditempatkan atau informasi yang diperlukan tersedia. [6]

Pada tugas akhir ini, *database service* dapat digunakan untuk menyimpan *dialplan* yang dapat diakses oleh pengguna VoIP.

- **Bearer Control**, Panggilan suara dibawa melalui *channel* yang disebut *Bearer Channel*. Diperlukan pengontrolan yang tepat pada bearer untuk memastikan *signaling call connect* atau *disconnect* terkirim ke tiap *end devices*. *Signaling* yang tepat diperlukan untuk pengalokasian *channel* ketika terjadi *voice call* dan dealokasi *channel* ketika salah satu sisi memutuskan sambungan.[6]

Pada tugas akhir ini, FreeSwitch melakukan fungsi *bearer control* kepada para pengguna VoIP yang ingin terhubung pada kelas virtual BigBlueButton.

- **Codecs**, melakukan *coding* dan *decoding* antara sinyal analog dan digital. PSTN menggunakan *codec* G.711 untuk mengubah sinyal analog suara ke 64-kbps aliran suara. Pada VoIP, *codec* yang digunakan bisa mengkompresi suara hingga dibawah 64-kbps untuk lebih mengefisienkan kapasitas jaringan. Yang paling sering digunakan di WAN adalah G.729 yang bisa mengkompresi suara hingga 8 kbps.[6]

Pada tugas akhir ini, ada dua *codec* yang digunakan, yaitu Sorenson Spark pada Red5 server untuk mengatasi video pada VoIP dan Speex Wideband pada FreeSwitch untuk mengatasi audio pada Voip.

2.4.2 **Signalling Protocol pada VoIP**

Seperti halnya komunikasi jaringan pada umumnya, VoIP memiliki beberapa protokol standar [6]:

- H.323
Standar komponen, protokol, dan prosedur yang menyediakan komunikasi multimedia, real-time audio,

video, dan data menggunakan packet network, termasuk IP. H.323 merupakan rekomendasi dari *International Telecommunication Union Telecommunication Standardization Sector* (ITU-T) yang disebut H.32x.[6]

- MGCP (*Media Gateway Control Protocol*)

Metode yang digunakan pada PSTN *gateway control* dan *thin device*. Protokol ini mengontrol VoIP *gateway* yang tersambung dengan *call control devices external* yang disebut *call agent*. MGCP memungkinkan *signaling* pada *devices* yang tidak terlalu mahal, seperti *gateway* yang tidak meimplementasikan *voice-signaling* protokol, seperti H.323.

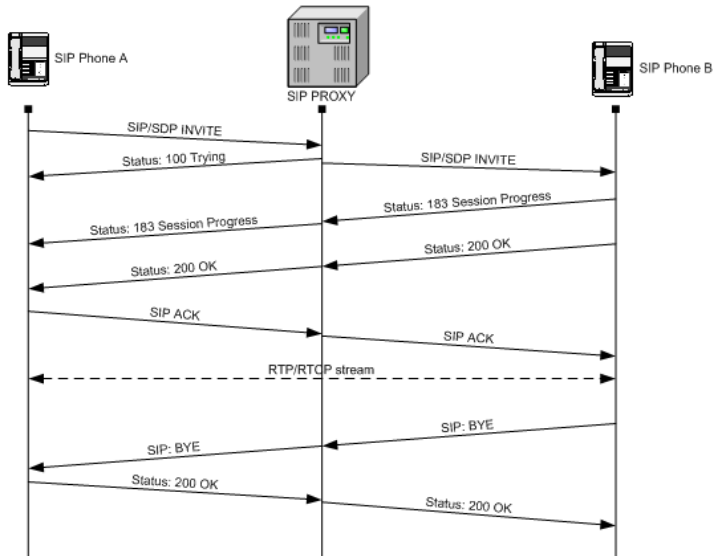
MGCP *Client/Server call control* protocol menggunakan arsitektur kontrol yang tersentralisasi. Pada protokol ini, *dial plan* disimpan pada *call agent* yang terpisah. *Call agent* bertugas mengontrol port pada *gateway*, melakukan *call control*. Pada cisco, *Communication Manager* yang akan bertindak sebagai *call agent*. Untuk berkomunikasi, MGCP mengadopsi *plain-text protocol* yang digunakan oleh *call-control devices* untuk manajemen *IP telephony gateway*.[6]

- SIP (*Session Intitation Protocol*)

SIP merupakan protokol *peer-to-peer alternatif* dari H.323. Protokol ini dikembangkan oleh *Internet Engineering Task Force (IETF) Multiparty Multimedia Session Control (MMUSIC) Working Group*. SIP termasuk mudah diimplementasikan karena menggunakan standar umum berbasis *World Wide Web*. Untuk berkomunikasi, SIP menggunakan *ASCII-text-based*.

SIP merupakan protokol yang lebih mendetail dengan perintah dan respon yang spesifik untuk melakukan atau mengakhiri panggilan. Memiliki fitur keamanan, proxy, dan layanan TCP/ UDP. SIP dengan SAP (*session*

announcement protocol) dan SDP (*session description protocol*), memberikan *announcement* tentang *multicast session* dari user-user yang ada di jaringan. SIP menggunakan *end-to-end signaling* antar *devices*. Pada Gambar 2.1 dapat mengilustrasikan mengenai pengimplementasian protokol SIP[6].



Gambar 2.1 Signalling SIP menggunakan SIP Proxy[6]

- SCCP (*Skinny Call Control Protocol*)

SCCP merupakan protokol Cisco *proprietary*. Protokol ini digunakan pada Cisco *Communication Manager* dan Cisco *IP Phone*. SCCP diklaim sebagai *session signaling protocol* yang ringan dan cocok digunakan pada Cisco *Communication Manager* (Cisco CallManager). SCCP adalah *Client-server protocol*, dimana *device* akan mengirimkan pesan ke Cisco UCM jika terjadi suatu *event* kemudian Cisco UCM akan memberikan

instruksi kepada *device* mengenai *event* apa yang perlu dilakukan. Jika diperlukan, SCCP dapat digunakan berdampingan dengan H.323, sehingga SCCP client tetap dapat berkomunikasi dengan terminal berprotokol H.323.[6]

Agar dapat lebih menjelaskan mengenai kelebihan dan kekurangan dari masing-masing signaling protocol pada VoIP dapat dilihat pada Tabel 2.1.

Tabel 2.1 Perbandingan *Signalling Protocol* pada VoIP

H.323	MGCP	SIP	SCCP
Peer-to-peer protocol	Client/server protocol	Peer-to-peer protocol	Client/server protocol
Dial-plan dan route pattern harus didefinisikan pada gateway	Dial plan dan route pattern didefinisikan langsung pada Cisco UCM di tiap cluster	Dial-plan dan route pattern harus didefinisikan pada gateway	
ISDN protocol (Q.921 dan Q.931) digunakan hanya antra link ISDN dan PSTN (hny sampai gateway) , shg semua signaling antara Cisco UCM dan H.323 gateway	Semua informasi ISDN layer 3 di-backhaul ke Cisco UCM server. Hanya ISDN Layer 2 (Q.921) dikirimkaan hingga gateway.	ISDN protocol (Q.921 dan Q.931) digunakan hanya antra link ISDN dan PSTN (hny sampai gateway)	

dihandle
protocol.

Protokol SIP digunakan dalam sistem ini untuk dapat menghubungkan pengguna yang ingin bergabung dalam ruang konferensi suara melalui jaringan PSTN.

2.5 *Asynchronous Javascript and XML (AJAX)*

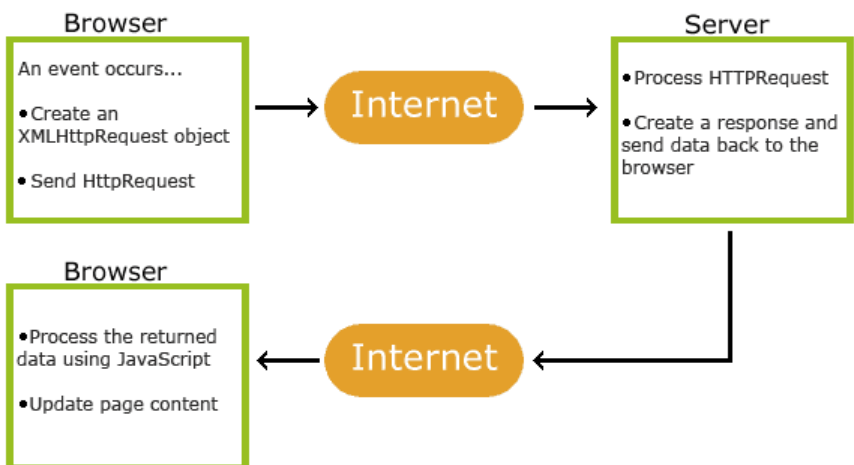
Asynchronous Javascript and XML (AJAX) merupakan cara untuk melakukan pertukaran data dengan server dengan menggunakan *Javascript*. Dimulai pada tahun 1999, saat XMLHttpRequest diperkenalkan pada peramban *Internet Explorer* dan yang lainnya, Jesse James Garret menciptakan sebuah istilah “AJAX” untuk menggambarkan bagaimana teknologi XMLHttpRequest melakukan kombinasi dengan Javascript dalam pengembangan sebuah *web*[7], seperti:

- HTML dan CSS untuk pembuatan struktur dan presentasi.
- *Document Object Model (DOM)* untuk melakukan perubahan secara dinamis.
- XML dan JSON untuk pertukaran data.
- Objek XMLHttpRequest untuk komunikasi secara asynchronous.

Dalam aplikasi *web* standar, interaksi antara pengguna dengan server bersifat *synchronous*, dimana ketika pengguna mengirimkan permintaan kepada server dan ketika server selesai mengolahnya maka akan dikirimkan kembali kepada pengguna. Jika suatu aplikasi *web* menggunakan AJAX, maka ketika pengguna melakukan permintaan kepada server, *Javascript* hanya mengirimkan konten yang diminta oleh pengguna. Pada proses permintaan ini, pengguna hanya diperlihatkan tampilan awal saja, sedangkan permintaan di proses oleh server dibalik layar dan kemudian ditampilkan kembali tanpa perlu melakukan reload

kembali pada halaman *web*. Oleh karena itu proses ini disebut *asynchronous*. [8]

Keunggulan AJAX adalah bersifat independen. AJAX dapat menangani sendiri permintaan dari pelanggan dan dapat melakukan pengolahan data dengan server tanpa harus melakukan interupsi pada interaksi pengguna dengan server. AJAX berjalan pada peramban melalui Javascript dan DOM untuk melakukan fungsinya. Pada Gambar 2.2 dapat dilihat bagaimana AJAX bekerja.



Gambar 2.2. Alur Kerja AJAX

Pada sistem ini, AJAX digunakan untuk melakukan pengiriman data JSON kepada berkas PHP sebagai *trigger* untuk mengubah nilai kompresi kamera web pada berkas XML yang bersangkutan.

2.6 Moodle

Moodle merupakan salah satu produk LMS, yang pada Tugas akhir ini digunakan sebagai LMS inti untuk melakukan manajemen terhadap pengguna dan *course* pelajaran. Moodle merupakan *platform* pembelajaran untuk para pendidik, administrator, dan

siswa yang menawarkan suatu sistem kuat, aman, dan terintegrasi sehingga dapat dikustomisasi berdasarkan kebutuhan lingkungannya. Moodle bersifat *open source*, yang berarti kita dapat menyalin, menggunakan, dan mengembangkannya selama kita tetap mencantumkan hak cipta dan menggunakan lisensi pada setiap karya yang kita ciptakan dengan menggunakan Moodle.

Moodle dapat diterapkan pada setiap computer yang dapat menjalankan PHP dan mendukung sistem basis data SQL. Selain itu, Moodle juga dapat berjalan pada sistem operasi Windows, Machintosh, dan berbagai macam distro Linux (seperti, Ubuntu, Debian, Linux Mint, dll).[9]

Purwarupa Moodle dibuat oleh Martin Dougiamas, Moodle Founder dan Lead Developer Moodle. Saat ia bekerja dan belajar di Curtin University, pengalamannya dalam bidang *WebCT* mendorongnya untuk melakukan riset mengenai metode alternative dalam pembelajaran online. Hingga pada tahun 1999, Martin Dougiamas dapat melakukan percobaan pertamanya dan mengeluarkan purwarupa berdasarkan papernya mengenai "*Improving the Effectiveness of Online Learning*". Hingga akhirnya Martin Dougiamas menggunakan nama Moodle dan mendaftarkannya sebagai merk dagang dari Moodle Trust.[10]

Dalam mendesain dan mengembangkan Moodle, para pengembang mengacu kepada sebuah konsep "*Social Contrustionist Pedagogy*". Berdasarkan konsep ini, ada empat hal utama yang menjadi acuan pengembangan, yaitu *Construtivism*, *Constructionism*, *Social Constructivism*, dan *Connected and Separate*. [11]

- *Contrutivism*

Dari sudut pandang ini, setiap orang akan secara aktif menerima suatu pengetahuan baru saat mereka berinteraksi dengan lingkungan yang baru. Moodle selalu menguji apakah fitur didalamnya, yang dapat kita lihat dan dengar, dapat diterima oleh para penggunanya dengan baik. Karena setiap pengguna bukanlah seperti bank ingatan yang secara

pasif dapat langsung menyerap dan menerima informasi yang baru.

- *Constructionism*

Constructionism ini menegaskan bahwa sejatinya suatu pembelajaran dapat lebih efektif ketika para pengajar dapat membuat sesuatu untuk dijadikan pengalaman oleh siswanya. Hal ini dapat berupa kalimat yang diucapkan pengajar, artikel pada internet, membuat presentasi atau menjelaskan dengan berbagai macam bentuk interpretasi lainnya. Moodle memfasilitasi para pengajar dan siswa dengan fitur-fitur yang dapat mendukung hal tersebut.

- *Social Constructivism*

Konsep ini memasukkan sudut pandang *Constructivism* kedalam ranah sosial. Ketika beberapa orang membangun kelompok-kelompok dan saling memberikan pengetahuan satu sama lain dan dengan skala waktu yang cukup intensif, maka hal ini akan menjadi budaya. Budaya ini akan membuat seseorang akan terus menambah pengetahuannya sepanjang waktu. Moodle memfasilitasi hal tersebut dengan fitur grup yang dapat dibuat dalam satu *course* yang diinginkan.

- *Connected and Separate*

Konsep ini menjelaskan mengenai motivasi seseorang dalam berdiskusi dapat dibagi menjadi tiga bagian:

1. *Separate Behaviour*, dimana seseorang mencoba untuk tetap obyektif dan faktual, dan cenderung mempertahankan ide mereka sendiri sembari mencari celah pada ide orang lain.
2. *Connected Behaviour*, pendekatan lebih empatik dimana seseorang mencoba menerima subyektivitas, mendengarkan dan mencoba mengajukan pertanyaan untuk dapat memahami sudut pandang orang lain.
3. *Construct Behaviour*, dimana seseorang tidak melakukan dua pendekatan diatas dan cenderung

memilih satu ide yang sesuai dengan kondisinya saat itu.

Moodle mencoba untuk membawa para penggunanya agar dapat menerapkan *Connected Behaviour*, karena dalam suatu komunitas hal tersebut akan menjadi suatu stimulan yang kuat untuk belajar dan mencapai hal yang lebih baik.

Berdasarkan konsep-konsep tersebut, Moodle dapat memberikan perkembangan yang positif bagi para pengajar dan siswa. Moodle juga sangat digemari dan banyak digunakan para pengembang LMS karena terus berkembang dan banyak fitur-fitur tambahan yang dikembangkan oleh kontributor Moodle. Hal inilah yang mendasari mengapa Moodle digunakan dalam tugas akhir ini.

2.6.1 Konten pada Moodle

Pada dasarnya sistem pada Moodle berbasis pada *course* yang dapat di kustomisasi oleh pengajar. Pada suatu *course*, pengajar dapat menyajikan materi atau referensi kepada para siswa sebagai bahan belajar atau bahan kegiatan. Sebuah *course* dapat mengandung konten untuk satu periode tahun ajaran, satu sesi pembelajaran, atau macam-macam varian lainnya tergantung kepada pengajar atau pendiri *course* tersebut dan dalam satu *course* dapat di kendalikan oleh satu pengajar atau sekelompok pengajar.

Metode pendaftaran siswa pada Moodle juga ditentukan oleh pendiri *course*, apakah siswa dapat mendaftarkan sendiri, secara manual oleh para pengajar, ataupun secara otomatis oleh admin. Moodle juga menawarkan *course* yang dapat disusun berdasarkan kategori sehingga lebih mudah dalam pencarian.[12]

Namun, terdapat kekurangan pada Moodle, yakni Moodle tidak memfasilitasi pengguna untuk berinteraksi secara *real time* melalui media audio visual, sehingga Moodle membutuhkan aplikasi *third-party* untuk dapat mengatasi permasalahan ini.

2.7 BigBlueButton

BigBlueButton merupakan *open source web conferencing system* untuk memfasilitasi pembelajaran secara *online*. Salah satu tujuan BigBlueButton sendiri dibuat adalah untuk mengatasi permasalahan jarak antar pengajar dan siswa atau kebutuhan lainnya yang membutuhkan sesi tatap muka, oleh karena itu digunakanlah BigBlueButton dalam sistem ini sebagai aplikasi perangkat lunak untuk membuat suatu kelas virtual yang dapat berbagi teks, presentasi, suara, dan kamera *web* antara pengajar dan siswa.

BigBlueButton memberikan kemudahan dalam berkomunikasi dengan fitur-fitur multimedia yang interaktif, seperti berbagi kamera *web*, bercakap menggunakan teks ataupun suara, dan berbagi dokumen, baik berbentuk PDF atau jenis berkas dokumen lainnya.[17]

Projek BigBlueButton sendiri mulai dirintis oleh pada tahun 2007 oleh Dr. Tony Bailetti, direktur *Technology and Innovation Management (TIM)* pada *Carleton University*. Projek ini berangkat dari keinginan TIM untuk mawadahi para mahasiswa secara global yang ingin mengikuti sesi perkuliahan pada Carleton University yang terkendala dengan jarak dan tingginya biaya untuk menyewa *web conference* komersil. Dr. Tony Bailetti menantang salah seorang dalam TIM, Richard Alam, yang kebetulan sedang melakukan tesis mengenai analisis terhadap *open source* business model untuk mencoba mengembangkan *open source web conference* untuk mengatasi kendala bagi mahasiswa global pada Carleton University. Bekerjasama dengan Denis Zgonjanin, mahasiswa *System and Computer Engineering, Carleton University*. Richard berhasil mengembangkan BigBlueButton versi awal menggunakan red5. Setelah itu, Fred Dixon, seorang pengusaha dan pengembang perangkat lunak, mempelajari BigBlueButton dan melakukan kerja sama dengan Richard hingga BigBlueButton menjadi *open source* terkemuka seperti saat ini.[13]

Hingga saat ini, BigBlueButton semakin banyak digunakan para pengembang di seluruh dunia untuk memfasilitasi *web conference* dalam suatu instansi atau perusahaan. Para pengembang

lepas dan kontributor BigBlueButton memiliki forum yang sangat hidup dan terbuka bagi siapa saja yang ingin mengetahui atau bereksperimen lebih mengenai BigBlueButton. Hal ini yang membuat BigBlueButton digunakan pada tugas akhir ini.

2.7.1 Komponen *Open source* pada BigBlueButton

BigBlueButton bukan merupakan *open source* yang dikembangkan secara mandiri. Dalam pengembangannya, BigBlueButton didukung oleh *open source* lainnya yang menjadi komponen dalam BigBlueButton. Saat ini tercatat ada empat belas *open source* yang menjadi komponen pendukung BigBlueButton[14].

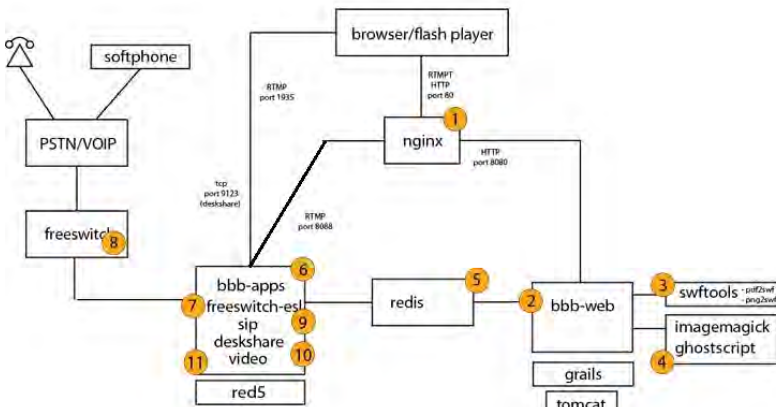
- Flex SDK
Karena BigBlueButton berfokus pada fitur multimedia interaktif, maka BigBlueButton menggunakan Flex SDK untuk mengembangkan fitur-fitur yang dimiliki. Flex menyediakan BigBlueButton produktivitas yang tinggi dalam pembuatan *web* yang ekspresif untuk semua jenis peramban dan sistem operasi.
- Ghostscript
BigBluebutton menggunakan Ghostscript untuk mengkonversi berkas presentasi jenis PDF menjadi berkas dengan jenis SWF.
- Grails
Grails merupakan *web frame work* yang digunakan BigBlueButton. Grails mengambil keuntungan dari bahasa pemrograman Groovy dan konvensi di atas konfigurasi untuk memberikan pengalaman pembangunan yang produktif dan stream-lined
- Apache ActiveMQ
ActiveMQ merupakan *opensource* yang mengelola pengiriman pesan dan integrasi pola pada suatu sistem.

BigBlueButton menggunakan ActiveMQ untuk mengirimkan pesan dalam proses pengolahan data.

- Image Magick
Open source ini melakukan maipulasi jenis berkas dan dapat diimplementasikan pada suatu *web*. Bigbluebutton menggunakan Image Magick untuk mengkonversi berkas data PDF menjadi SWF.
- MySQL
MySQL merupakan *open source* basis data yang digunakan oleh BigBlueButton.
- Nginx
Dibaca “engine x”, merupakan HTTP server dan *mail proxy* yang digunakan oleh BigBlueButton.
- Red5
Red5 merupakan *open source* Flash Server. Fitur-fitur dalam BigBluebutton yang berbasis Flash ditangani oleh Red5.
- SWFTools
SWFTools memiliki banyak fitur dan kemampuan untuk bekerja dengan berkas Adobe Flash. Fitur-fitur dalam BigBlueButton dibuat menggunakan SWFTools.
- LibreOffice
BigBlueButton menggunakan LibreOffice dalam melakukan konversi berkas menjadi SWF.
- FreeSwitch
FreeSwitch merupakan *open source telephony platform* yang digunakan untuk melakukan koneksi VoIP. BigBlueButton menggunakan FreeSwitch untuk menangani fitur VoIP *conference*.
- Redis
Redis merupakan perangkat lunak *open source key-value data*. BigBlueButton menggunakan Redis untuk melakukan pendataan internal, seperti jadwal akses, data pengguna, dan informasi lain yang relevan.

Komponen-komponen BigBlueButton tersebut semua digunakan dalam satu aplikasi yang dapat memfasilitasi pengguna dalam berinteraksi dalam kelas virtual. Jika pengguna ingin mengunggah berkas presentasi, maka LibreOffice, Grails, dan ImageMagick yang berperan untuk mengkonversi dan menyebarkan kepada pengguna lain. Jika pengguna ingin melakukan *audio video conference*, maka Red5 dan FreeSwitch yang berperan dalam pengaturan dan penyebaran berkas audio dan video ke pengguna lainnya.

2.7.2 Arsitektur BigBlueButton



Gambar 2.3. Arsitektur BigBlueButton[16]

Secara keseluruhan, arsitektur pada BigBlueButton dapat diilustrasikan seperti pada Gambar 2.3. Dalam arsitektur BigBlueButton, dapat dijelaskan sebagai berikut :

- Ketika pengguna masuk ke dalam ruang konferensi, maka permintaan akan dikirimkan kepada Nginx, lalu kemudian di salurkan kepada *bbb-web* dan Grails yang akan melakukan pengolahan permintaan. Begitu juga saat pengguna ingin keluar dari konferensi.[15]

- Ketika pengguna akan menggunakan fitur multimedia, seperti berbagi kamera *web*, konferensi suara atau berbagi desktop, permintaan akan dikirimkan kepada *nginx*, lalu disalurkan kepada *bbn-web*. *Bbb-web* akan menerjemahkan permintaan pengguna dan memberikan pancingan kepada *bbb-apps* melalui *redis*. Setelah *bbb-apps* mendapatkan terjemahan permintaan dari *bbb-web*, kemudian diakses dan dikirimkan kembali kepada pengguna.[15]
- Ketika pengguna akan menggunakan fitur pengunggahan presentasi, akan melawati proses yang hampir sama dengan fitur multimedia lainnya. Hanya saja sebelum dikirimkan ke *bbb-web* kepada *bbb-apps*, berkas presentasi akan dikonversi terlebih dahulu menggunakan *swftools*, *Image Magick*, atau *Ghostscript*, kemudian diteruskan kepada *bbb-apps* melalui *redis*.[15]
- Ketika pengguna ingin menggunakan fitur VoIP untuk bergabung dengan konferensi suara dengan menggunakan IP Phone atau telepon reguler, pengguna akan diteruskan oleh *FreeSwitch* ke *bbb-apps*. Kemudian akan ditampilkan kepada pengguna lain. Apabila pengguna VoIP dikenakan sebuah even (contoh fitur *mute*) dalam ruang konferensi, maka permintaan tersebut akan diteruskan oleh *Nginx* kepada *bbb-apps* dan melalui *FreeSwitch* akan diteruskan perlakuan yang dikenakan kepada pengguna VoIP.[15]

Tugas akhir ini lebih terfokus pada penggunaan fitur multimedia pada *BigBlueButton*, terutama mengenai kualitas kompresi kamera web yang akan menyesuaikan dengan kecepatan akses internet pengguna.

BAB III

PERANCANGAN PERANGKAT LUNAK

Pada bab ini akan dijelaskan mengenai hal-hal yang berkaitan dengan perancangan sistem yang akan dibuat. Bab ini akan diawali dengan penjelasan umum aplikasi hingga alur proses implementasinya.

3.1 Deskripsi Umum Perangkat Lunak

Tugas akhir ini mencoba membuat suatu sistem pembelajaran yang bersifat *online* dan *real time* dengan mengimplementasikan *Learning Management System Moodle* dan *BigBlueButton*. Moodle merupakan suatu sistem perangkat lunak manajemen pembelajaran yang bersifat gratis. Moodle dibuat untuk memenuhi kebutuhan para pengajar mengajar dalam suatu kelas online dengan fokus tertentu, dengan fitur-fitur yang interaktif dan memudahkan pengajar dan juga peserta didiknya. Moodle memiliki fitur untuk mengunggah dan mengunduh berkas materi, mengadakan kuis online, pengumpulan tugas, dan juga grup diskusi yang semakin memudahkan dalam berinteraksi antar pengajar dengan peserta didik. Namun, Moodle sendiri belum mendukung interaksi dalam bentuk *real time*.

Salah satu nilai lebih yang terdapat pada Moodle adalah Moodle memberikan kesempatan yang besar kepada para pengembang perangkat lunak untuk berkreasi dalam membuat fitur-fitur, aplikasi, dan *platform* yang mendukung Moodle semakin kuat. Termasuk untuk mengatasi permasalahan terkait interaksi *real time* yang belum di masukkan dalam lingkup Moodle oleh pengembang Moodle sendiri. Salah satu *platform* yang dapat mendukung interaksi *real time* pada Moodle adalah *BigBlueButton*.

BigBlueButton merupakan *open source web conferencing* yang dapat diterapkan pada berbagai macam sistem operasi. *BigBlueButton* memberikan fitur kepada pengajar untuk mengadakan suatu kelas virtual yang dapat diikuti oleh para peserta didik. Pengajar sebagai moderator dapat berperan sebagai presenter dan juga menunjuk peserta sebagai presenter juga dengan dilengkapi

dengan fitur mengunggah berkas presentasi. Presenter juga dapat memberikan penekanan pada presentasinya dengan menulis, highlight, atau melingkari bagian penting dengan fitur papan tulis. Selain itu presenter juga dapat berbicara kepada peserta dalam forum, melihat wajah peserta pada kamera komputer, serta berkiriman pesan kepada peserta, baik personal maupun *broadcast*. Dan pada BigBlueButton versi terbaru (0.81), presenter dapat merekam sesi kelas virtual tersebut sehingga dapat disimpan dan dilihat kembali oleh peserta maupun presenter.

Berdasarkan pada kekuatan dan fitur-fitur dari BigBlueButton tersebut, BigBlueButton dicoba dikembangkan untuk menerapkan fitur-fitur tersebut kedalam Moodle untuk mengatasi permasalahan interaksi *real time* untuk membuat Moodle semakin kuat. Dari dua *platform* yang berbeda, Moodle dan BigBlueButton akan diintegrasikan sehingga fitur yang terdapat di keduanya dapat digunakan oleh pengguna dalam satu tempat. Selain itu, untuk memperkuat daya dukung BigBlueButton dalam menangani kelas virtual, akan dilakukan pengembangan pada BigBluebutton berupa penambahan fitur, yaitu video kompresi secara dinamis, dimana saat pengguna melakukan konferensi video dengan membagikan kamera *web*, secara otomatis kompresi video kamera akan mengikuti kecepatan akses internet yang dimiliki pengguna.

3.2 Arsitektur Umum Perangkat Lunak

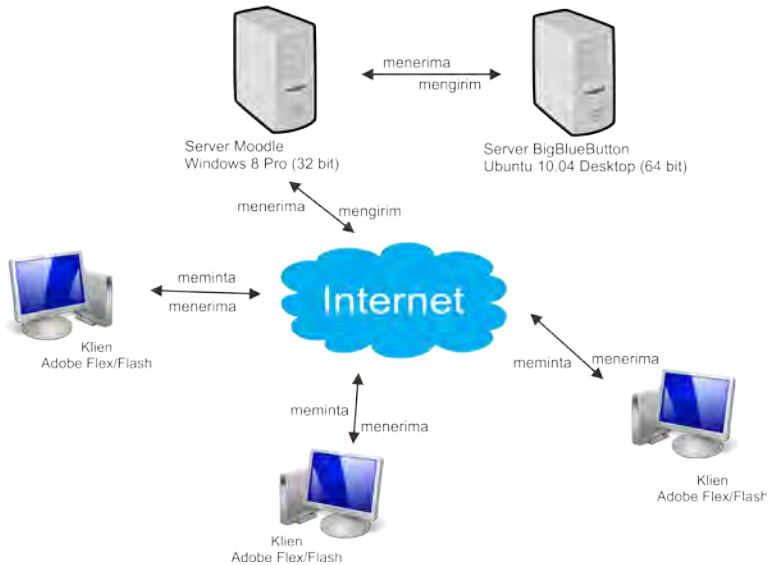
Pada sistem ini, digunakan dua buah server, yang pertama sebagai server untuk implementasi Moodle, dan yang kedua digunakan untuk implementasi BigBlueButton. Untuk ilustrasi mengenai arsitektur umum perangkat lunak ini dapat dilihat pada Gambar 3.1.

Untuk masuk dan penggunaan fitur-fitur umum dalam sistem, pengguna (baik guru maupun peserta didik) akan diarahkan kepada server Moodle. Kemudian jika pengguna (pengajar) ingin mengadakan kelas virtual dalam suatu course yang disediakan oleh Moodle, server Moodle akan melakukan permintaan kepada server BigBlueButton untuk membuat reservasi penggunaan kelas virtual.

Kemudian setelah itu, BigBlueButton akan memberikan izin untuk akses kelas BigBlueButton kepada moderator (pengajar

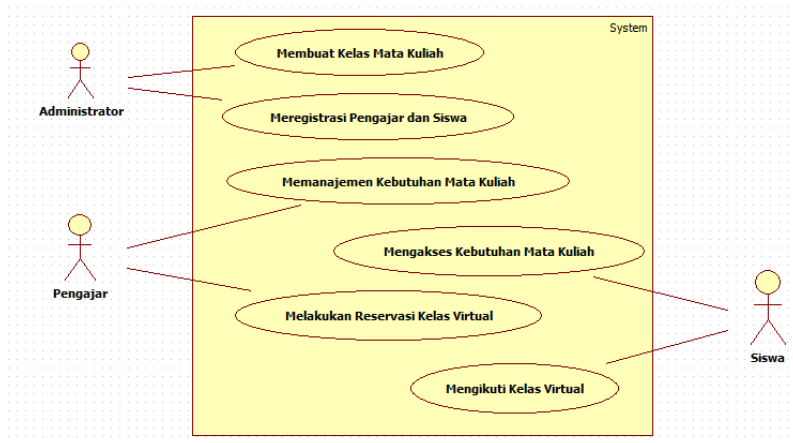
yang membuat reservasi kelas). BigBlueButton akan memberikan semacam kunci kepada moderator. Dengan kondisi tersebut, para peserta yang ingin mengakses kelas virtual pada BigBlueButton harus menunggu hingga moderator masuk dan memulai kelas virtual tersebut.

Kemudian untuk meninggalkan kelas virtual, pengguna cukup menutup jendela kelas virtualnya pada browser, maka akan langsung memutuskan koneksi dengan server BigBlueButton.



Gambar 3.1. Arsitektur Umum Perangkat Lunak

3.3 Diagram *Use case* Perangkat Lunak



Gambar 3.2. Diagram *Use case* Perangkat Lunak

Diagram *Use case* menggambarkan fungsionalitas sistem beserta aktor yang terlibat. Dalam diagram *use case* pada Gambar 3.2. dijelaskan bahwa pada sistem ini terdapat tiga aktor, yaitu administrator, pengajar dan siswa.

Pada sistem ini administrator dapat membuat kelas mata kuliah pada sistem sesuai dengan kebutuhan lembaga. Administrator juga bertugas untuk memasukkan para pengguna dari sistem dan menentukan status dari masing-masing pengguna, apakah seorang pengajar atau siswa. Administrator juga yang menentukan siapa yang menjadi pengajar dan siswa di setiap kelas yang dibuat.

Jika seorang pengguna memiliki status pengajar, maka pengguna tersebut dapat memanajemen kebutuhan kelas mata kuliahnya, seperti mengunggah materi dan referensi terkait kegiatan belajar dan aktifitas siswa, membuat tempat untuk pengumpulan tugas secara online, atau membuat kuis online untuk siswa. Selain itu, pengguna dengan status pengajar dapat melakukan melakukan reservasi kelas virtual untuk kelas mata kuliahnya.

Untuk pengguna yang memiliki status sebagai siswa, maka pengguna tersebut dapat mengakses segala yang dibutuhkan dari kelas mata kuliah. Pengguna dapat mengunduh materi atau referensi terkait kegiatan belajar dan aktifitasnya, mengunggah tugas, ataupun ikut dalam kuis online yang dibuat oleh pengajar. Pengguna dengan status siswa dapat mengikuti kelas virtual ketika pengajar telah masuk kedalam kelas virtual terlebih dulu.

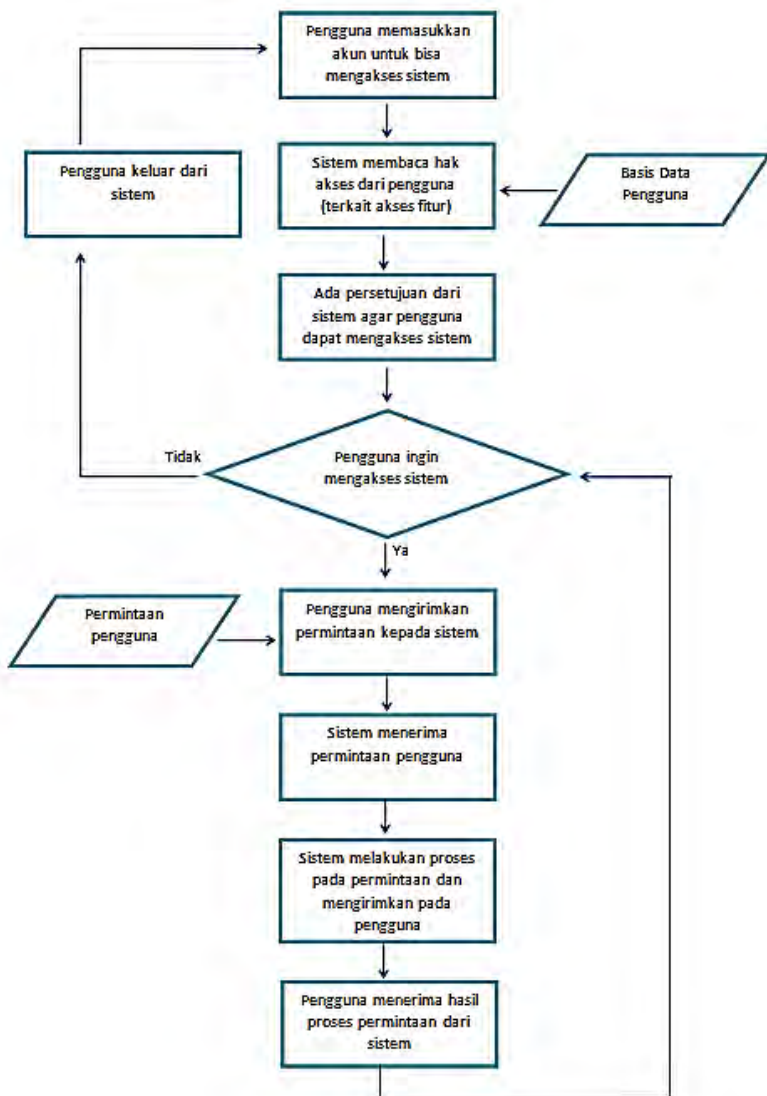
3.4 Alur Kerja dan Diagram Alir Perangkat Lunak

Setiap perangkat lunak memiliki alur kerja atau urutan urutan yang harus dilakukan untuk menjalankan sistem tersebut. Pada aplikasi ini terdapat beberapa proses yang harus dijalankan agar pengguna dapat menggunakan sistem sesuai dengan kebutuhannya.

3.4.1 Alur Kerja dan Diagram Alir Penggunaan Sistem

Dalam penggunaan sistem ini, terdapat alur yang dapat diikuti oleh pengguna seperti yang diilustrasikan pada Gambar 3.3. Berikut diagram alir sistem:

1. Pengguna melakukan autentikasi akun yang dimiliki.
2. Sistem akan membaca hak akses pengguna berdasarkan hak akses.
3. Sistem akan memberikan ijin kepada pengguna untuk mengakses sistem berdasarkan hak akses yang dimiliki.
4. Jika pengguna tidak ingin mengakses sistem kembali, maka pengguna bisa memilih pilihan untuk keluar dari sistem.
5. Jika pengguna ingin mengakses sistem, pengguna akan melakukan permintaan kepada server sistem.
6. Server sistem akan menanggapi permintaan pengguna dan kemudian dilakukan proses berdasarkan permintaan yang dikirimkan pengguna.
7. Server sistem akan mengirimkan hasil proses kepada pengguna.



Gambar 3.3. Diagram Alir Penggunaan Sistem

3.4.2 Alur Kerja dan Diagram Alir Permintaan kepada Server BigBlueButton

Untuk dapat mengakses server BigBlueButton, terdapat alur tahapan yang dijalankan seperti yang terdapat pada Gambar 3.4. Penjelasan mengenai alur tahapan tersebut dapat dijelaskan sebagai berikut:

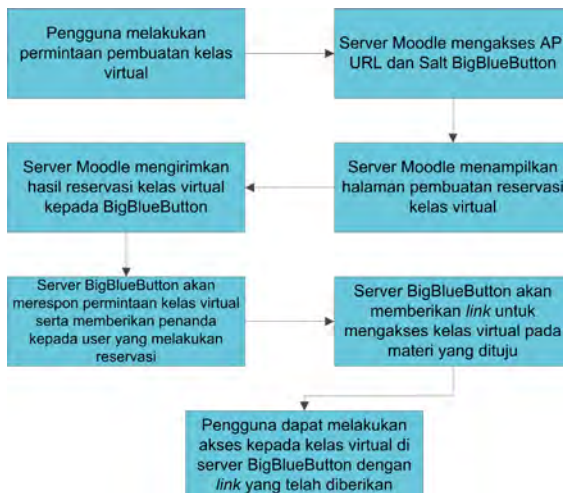
1. Pengguna melakukan permintaan untuk pembuatan kelas virtual menggunakan plugin yang telah diimplementasikan pada server Moodle.
2. Server Moodle akan mengakses API URL dan Salt pada server BigBlueButton.
3. Server Moodle akan menampilkan halaman untuk melakukan reservasi kelas virtual.
4. Server Moodle akan mengirimkan hasil reservasi kelas virtual kepada server BigBlueButton.
5. Server BigBlueButton akan menanggapi permintaan dari server Moodle dan memberikan penanda kepada pengguna yang melakukan reservasi.
6. Server BigBlueButton akan memberikan link untuk mengakses kelas virtual yang telah direservasi.
7. Pengguna dapat mengakses kelas virtual melalui link yang telah diberikan.

3.4.3 Alur Kerja dan Diagram Alir Perubahan Nilai Kompresi pada BigBlueButton

Pada sistem ini terdapat suatu fitur dimana kualitas kamera *web* yang di bagikan dapat berubah secara dinamis mengikuti kecepatan akses internet dari pengguna. Secara umum, pada Gambar 3.5 dapat mengilustrasikan alur perubahan nilai kompresi untuk kualitas kamera *web*. Penjelasan mengenai alur tahapan tersebut dapat dijelaskan sebagai berikut:

1. Pengguna masuk kedalam sistem BigBlueButton dengan menggunakan peramban yang telah dilengkapi dengan Flash Plugin.
2. Ketika pengguna masuk dari sisi klien BigBlueButton, kecepatan akses pengguna akan dibaca oleh sistem.

3. Setelah mendapatkan nilai kecepatan akses pengguna, sistem akan menentukan berapa nilai kompresi yang diberikan berdasarkan kecepatan akses pengguna. Kemudian sistem akan mengirimkan sebuah pancingan kepada server untuk dapat mengubah nilai kompresi sesuai dengan kebutuhan pengguna.
4. Setelah server mendapatkan nilai kecepatan akses pengguna, maka server akan mengubah berkas konfigurasi kompresi kamera *web* sesuai dengan nilai kecepatan akses yang didapat.
5. Kemudian server akan mengirimkan kembali kepada pengguna berkas konfigurasi kamera *web* yang telah diubah sesuai dengan kebutuhan untuk digunakan dalam menjalankan fitur multimedia pada kelas virtual.
6. Setelah mendapatkan berkas konfigurasi tersebut, maka pengguna dapat langsung menggunakan fitur multimedia sesuai dengan kecepatan akses internet yang dimiliki pengguna.



Gambar 3.4 Diagram Alir Permintaan kepada Server BigBlueButton



Gambar 3.5. Diagram Alir Perubahan Nilai Kompresi

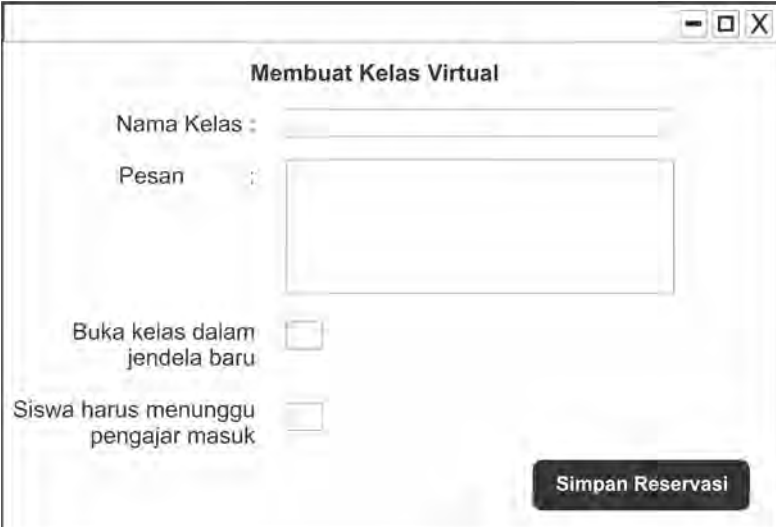
3.5 Perancangan Antar Muka Perangkat Lunak

Pengerjaan Tugas akhir ini lebih menekankan kepada bagaimana cara mengintegrasikan antara fitur-fitur Moodle dan juga BigBlueButton. Agar pengguna dapat membuat kelas virtual, secara umum maka pengguna akan dihadapkan pada antarmuka seperti yang terdapat pada Gambar 3.6.

Keterangan mengenai Gambar 3.6:

1. Pengajar dapat memasukkan nama kelas virtual yang telah dibuat pada text area Nama kelas yang telah disediakan.
2. Pengajar dapat memasukkan pesan dalam kelas virtual yang telah dibuat, seperti peraturan dalam kelas atau apapun pada text area Pesan yang telah disediakan.

3. Menu Checklist "Buka kelas dalam jendela baru" memberikan fasilitas bagi pengguna apabila ketika memasuki kelas virtual, ingin berada pada jendela peramban yang baru.
4. Menu Checklist "Siswa harus menunggu pengajar masuk" memberikan batasan kepada siswa, dimana siswa tidak dapat mengakses kelas virtual hingga pengajar masuk terlebih dahulu ke dalam kelas virtual.
5. Tombol "Simpan Reservasi" akan menyimpan reservasi kelas virtual yang telah dibuat oleh pengajar. Kelas virtual tersebut dapat dilihat dan diakses langsung dari course tempat kelas virtual tersebut dibuat.



The image shows a web form titled "Membuat Kelas Virtual" (Create Virtual Class). The form contains the following elements:

- A text input field for "Nama Kelas" (Class Name).
- A text area for "Pesan" (Message).
- A checkbox labeled "Buka kelas dalam jendela baru" (Open class in new window).
- A checkbox labeled "Siswa harus menunggu pengajar masuk" (Students must wait for teacher to enter).
- A dark button labeled "Simpan Reservasi" (Save Reservation).

Gambar 3.6 Rancangan Tampilan Antar Muka Reservasi Kelas Virtual

BAB IV

IMPLEMENTASI PERANGKAT LUNAK

Pada bab ini membahas mengenai lingkungan implementasi perangkat lunak dan implementasi dari rancangan perangkat lunak yang telah dijabarkan pada bab 3. Implementasi meliputi implementasi arsitektur dan antar muka untuk klien dan server.

4.1 Lingkungan Implementasi

Aplikasi ini dibangun dengan lingkungan perangkat lunak dan perangkat keras seperti penjabaran berikut:

4.1.1 Lingkungan Perangkat Lunak

Perangkat lunak yang digunakan untuk merancang dan mengimplementasikan sistem ini adalah:

- Sistem Operasi Linux Ubuntu 10.04 64 bit untuk server BigBlueButton
- Sistem Operasi Windows 8 Pro 32 bit untuk server Moodle
- Gnome-Terminal
- PHP 5.3.2 dan MySQL 5.1.33.
- Web server Apache Tomcat 7.0 untuk Moodle.
- Nginx server untuk BigBlueButton.
- Red5 untuk *streaming* server berbasis protokol RTMP.
- Star UML untuk membuat diagram *usecase*
- Microsoft Office Visio 2007 untuk membuat diagram alir
- Corel Draw X5 untuk membuat arsitektur dan rancangan antarmuka sistem.

4.1.2 Lingkungan Perangkat Keras

Spesifikasi perangkat keras yang digunakan pada lingkungan pengembangan sistem ini adalah sebagai berikut:

- Notebook dengan prosesor Intel(R) Core(TM) i3 CPU M 350 @2,27Ghz(4CPU's) dan RAM 2GB.
- Personal Computer dengan prosesor Intel(R) Core (TM) i3-3210 CPU @ 3,20GHz (2 CPU's) dan RAM 4 GB.

4.2 Implementasi Server Moodle

Sistem ini menggunakan Moodle sebagai inti dari sistem. Basis data penggunaan untuk melakukan kelas virtual juga di buat melalui Moodle, oleh karena itu, pada server Moodle dilakukan instalasi MySQL, Apache, dan PHP. Setelah melakukan instalasi MySQL, Apache, dan PHP, kemudian dibuat basis data kosong yang akan digunakan saat melakukan instalasi Moodle.

Pada sistem ini, Moodle yang digunakan adalah Moodle versi 2.4 yang dapat diunduh pada situs resmi Moodle. Kemudian setelah melakukan instalasi, maka pengguna sebagai administrator melakukan pengaturan, seperti membuat kelas dan memasukkan data pengguna.

4.3 Implementasi Server BigBlueButton

BigBlueButton terdiri dari berbagai macam *open source* lainnya yang menjadi komponennya. Dalam masing-masing komponen terdapat konfigurasi yang dapat membuat BigBlueButton menjalankan fungsinya sebagai *web conference*, baik dari sisi server maupun sisi klien.

4.3.1 Konfigurasi pada Nginx

Nginx memberikan jalur kepada BigBlueButton agar dapat diakses oleh para penggunanya. Pada Nginx, BigBlueButton diberikan port 80 sebagai jalur aksesnya dan diatur alamat IP sesuai dengan alamat IP server. Pada Nginx, juga melakukan konfigurasi pendaftaran direktori yang dapat diakses oleh pengguna.

4.3.2 Konfigurasi pada Red5

Red5 secara keseluruhan melakukan konfigurasi terhadap aplikasi multimedia pada BigBlueButton. Ada dua hal yang dikonfigurasi Red5 terhadap BigBlueButton, yaitu fitur *streaming* dan VoIP *conference*.

4.3.3 Konfigurasi pada Tomcat6

Pada Tomcat6, dilakukan konfigurasi yang menjadi konfigurasi utama dari *web* aplikasi BigBlueButton. Konfigurasi ini mengatur BigBlueButton agar dapat mengenali link direktori untuk menjalankan aplikasi multimedia BigBlueButton, batasan-batasan untuk ruang konferensi, pengaturan port dan dial number untuk VoIP *conference* serta kode salt dari BigBlueButton.

4.3.4 Konfigurasi pada FreeSwitch

Pada FreeSwitch dilakukan konfigurasi agar dapat terhubung antara server dengan pengguna VoIP. Konfigurasi yang dilakukan meliputi pengaturan Codec yang digunakan, fitur tombol pada telepon (fitur hold, music, dan lain-lain), dan melakukan panggilan.

4.3.5 Konfigurasi Klien BigBlueButton

Agar pengguna dapat menggunakan fitur-fitur pada ruang konferensi BigBlueButton, maka dibutuhkan suatu konfigurasi untuk menjalankannya. Pada konfigurasi ini, terdapat parameter dan modul yang dapat diatur untuk menjalankan fitur-fitur multimedia BigBlueButton. Melalui konfigurasi ini, juga dapat dilakukan pengadaan atau peniadaan modul sesuai dengan kebutuhan.

4.4 Implementasi Pengintegrasian Moodle dan BigBlueButton

Untuk dapat melakukan integrasi antara Moodle dan BigBlueButton, pada Tugas akhir ini menggunakan dua plugin tambahan yang di tambahkan pada Moodle, yaitu BigBlueButtonBN dan RecordingsBN. BigBlueButtonBN untuk dapat membuat ruang konferensi pada BigBlueButton melalui Moodle, sedangkan RecordingsBN untuk dapat mengakses ruang koferensi yang direkam oleh BigBlueButton melalui Moodle. Kemudian dibutuhkan url salt dari BigBlueButton yang ditanamkan dalam aplikasi *third-party*, seperti Moodle, agar dapat melakukan autentikasi kepada server BigBluebutton ketika akan mengaksesnya.

4.5 Implementasi Fitur Pengubahan Nilai Kompresi pada Kamera Web BigBlueButton

Untuk dapat melakukan pengubahan nilai kompresi pada kamera *web* BigBlueButton, dapat dilakukan pengubahan nilai pada atribut “CamQualityPicture” pada konfigurasi klien yang berupa berkas XML, yang terdapat pada sisi server. Karena pancingan data, yang berupa kecepatan akses pengguna, untuk mengubah nilai kompresi berada pada sisi klien, maka dibutuhkan AJAX untuk mengirimkan masukan data kepada suatu fungsi PHP, yang kemudian akan mengubah konfigurasi XML dan kemudian dikirimkan kepada pengguna.

4.5.1 Membaca Kecepatan Akses Pengguna

Pada javascript BigBlueButton.html, diletakkan fungsi seperti pada Gambar 4.1 untuk mengunduh suatu berkas gambar dengan jenis PNG yang terletak pada server. Nilai kecepatan mengunduh tersebut disimpan dalam sebuah variable, kemudian di konversi ke dalam satuan Mbps.

```

download.onload = function () {
    duration = time of download picture
    bitsLoaded = downloadSize * 8
    get speed in Bps = (bitsLoaded / duration)
    get speed in Kbps = (speedBps / 1024)
    get speed in Mbps = (speedKbps / 1024)
}

```

Gambar 4.1 Fungsi Membaca Kecepatan Akses Pengguna

4.5.2 Konversi Nilai Kecepatan Pengguna dengan Nilai Kompresi

```

var change;
if (speed tidak 0) {
if (speed antara 0-6 mbps) change=50;
if (speed antara 6,1-7,5 mbps) change=70;
if (speed diatas 7,5 mbps) change=90;
}

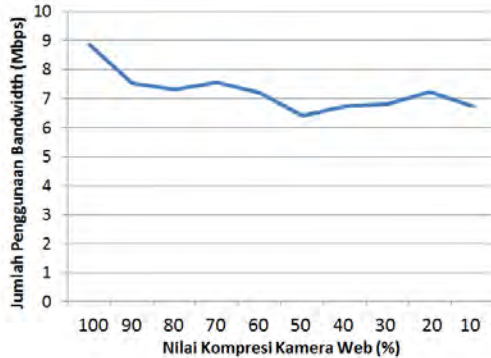
```

Gambar 4.2 Fungsi Konversi Nilai Kecepatan dengan Nilai Kompresi

Pada javascript BigBlueButton.html, diletakkan suatu fungsi seperti pada Gambar 4.2 untuk menentukan nilai kompresi sesuai dengan berapa nilai kecepatan akses pengguna. Perubahan nilai kompresi ditentukan berdasarkan pada hasil uji coba penulis terhadap perbandingan antara nilai kompresi dengan skala 10 dengan jumlah bandwidth yang digunakan sesuai dengan grafik yang ditunjukkan pada Gambar. 4.3. Ada tiga kondisi yang diberikan yaitu :

- Jika kecepatan akses internet berada diantara 0 hingga 6 mbps, maka akan diberikan nilai kompresi 50.
- Jika kecepatan akses internet berada diantara 6,1 mbps hingga 7,5 mbps, maka akan diberikan nilai kompresi 70.

- Jika kecepatan akses internet berada diatas 7,5 mbps, maka akan diberikan nilai kompresi 90.



Gambar 4.3 Grafik Perbandingan Nilai Kompresi dengan Jumlah Penggunaan Bandwidth

4.5.3 Pengiriman Nilai Kompresi Menggunakan AJAX

Setelah didapatkan nilai kompresi yang akan diubah, atau disebut JSON, maka JSON ini akan dikirimkan dengan menggunakan AJAX kepada fungsi PHP yang terdapat dalam server untuk mengubah konfigurasi klien dengan nilai kompresi yang ditentukan dengan menggunakan fungsi seperti pada Gambar 4.4.

```
declare array
decare temp

change = temp.change
push temp to array

JSON str = JSON to string
make new XMLHttpRequestObject()
request.setRequestHeader('Content-type','application/json','true')]
```

Gambar 4.4 Fungsi Mengirim Nilai Kompresi dengan AJAX

4.5.4 Perubahan Nilai Kompresi Menggunakan PHP XML DOM

```

value = $_GET['JSONstr'];
doc = new DOMDocument();
doc->load(xml path);
xpath= new DOMXPath(doc);

foreach (module dalam modules){
module->setAttribute(camQualityPicture sesuai value);
}
save(to xml path);?>

```

Gambar 4.5 Fungsi Perubahan Nilai Kompresi pada Berkas XML

Setelah javascript berhasil mengirimkan JSON kepada PHP, maka PHP akan mengambil JSON tersebut sebagai nilai untuk mengubah nilai kompresi. Agar PHP dapat mengakses berkas XML konfigurasi klien, digunakan teknologi DOM yang terdapat pada PHP5. PHP membuka kanal kepada berkas XML yang terkait dan kemudian melakukan pencarian berdasarkan tag yang terdapat dalam berkas XML (pada kasus ini PHP akan mencari tag attribute “CamQualityPicture”). Setelah mendapatkan tag tersebut, PHP akan melakukan setAttribute terhadap tag itu sesuai dengan JSON yang didapatkan diawal. Setelah melakukan setAttribute, kemudian PHP akan menulis kembali berkas XML tersebut sesuai dengan modifikasi yang telah dilakukan, seperti fungsi yang ditunjukkan pada Gambar 4.5.

4.6 Implementasi Rancangan Antarmuka

Pada Gambar 4.6 dapat dilihat hasil dari implementasi antarmuka perangkat lunak yang telah dirancang pada bab 3. Antarmuka tersebut yang akan digunakan pengguna untuk melakukan reservasi kelas virtual BigBlueButton pada Moodle.

The screenshot displays the Moodle user interface for a course titled 'Tingkat Menengah'. The page is in the 'General settings' section, where the 'Virtual classroom name' is set to 'Diskusi EDI'. A red box highlights the 'Welcome message' field, which contains the text: 'Selamat datang di forum bebas diskusi EDI! Sajikan EDI masing-masing kelas pada diskusi.' The interface includes a navigation menu on the right with options like 'My home', 'Site pages', 'My profile', and 'Current course'. The 'Current course' menu is expanded, showing sub-items such as 'Participants', 'Reports', 'General', 'Etika Diskusi Ilmiah', 'Misi, Visi, dan Misi Organisasi', 'Departementalisasi dan Staffing', and 'Identifikasi Masalah'. The browser address bar shows the URL: 'http://www.moodle.com...'. The page title is 'Tingkat Menengah'.

Gambar 4.6 Implementasi Rancangan Antarmuka

BAB V

UJI COBA DAN EVALUASI

Pada bab ini akan dibahas mengenai uji coba dan evaluasi perangkat lunak. Perangkat lunak diuji coba dari segi fungsionalitas dan performa dengan berbagai skenario. Pada bab ini juga dibahas lingkungan uji coba.

5.1 Lingkungan Uji Coba

Lingkungan uji coba menjelaskan apa saja uji coba yang dilakukan dan perangkat apa sajakah yang digunakan dalam melakukan uji coba ini, berikut penjabarannya:

5.1.1 Lingkungan Perangkat Lunak

Lingkungan perangkat lunak yang digunakan untuk uji coba adalah sebagai berikut :

- 1 Sistem operasi Linux Ubuntu 10.04 desktop
- 1 Sistem operasi Windows 8 Pro 32bit
- 1 Peramban Google Chrome version 28.0.1500.72 m
- 1 Peramban Mozilla Firefox versi 22.0

5.1.2 Lingkungan Perangkat Keras

Lingkungan perangkat keras yang digunakan untuk uji coba adalah sebagai berikut :

- Notebook dengan prosesor Intel(R) Core(TM) i3 CPU M 350 @2,27Ghz(4CPUs) dan RAM 2GB.
- Personal Computer dengan prosesor Intel(R) Core (TM) i3-3210 CPU @ 3,20GHz (2 CPUs) dan RAM 4 GB
- 1 WebCamera M-Tech 5 Megapixels
- 1 Headphone AKG K420
- 1 Mikrofon Wave M10

5.2 Uji Coba Fungsionalitas

Uji coba fungsionalitas dilakukan untuk melihat apakah fungsi-fungsi dasar aplikasi berjalan sebagaimana mestinya. Uji coba fungsionalitas meliputi semua *use case* yang telah dijelaskan pada Bab 3.

5.2.1 Uji Coba Autentikasi pada Moodle

Seperti dapat dilihat pada Gambar 5.1, pada saat akan memasuki sistem perangkat lunak, pengguna harus memasukkan username dan password sebagai autentikasi dalam sistem, lalu kemudian barulah pengguna dapat menggunakan fitur-fitur dalam Moodle sesuai dengan status yang diberikan oleh administrator.



Gambar 5.1 Uji Coba Autentikasi Pengguna

5.2.2 Uji Coba Mengunggah Berkas Pendukung pada Moodle

Bagi pengguna yang memiliki status sebagai pengajar, maka pengguna tersebut dapat mengelola course mata kuliah yang diajarnya sesuai yang dibutuhkan. Salah satu yang dilakukan pengajar untuk mengelola course mata kuliahnya adalah mengunggah berkas pendukung, berupa materi atau referensi yang mendukung kegiatan perkuliahan seperti pada Gambar 5.2. Pengguna dapat menggunakan tombol ambil berkas atau langsung melakukan

drag and drop berkas yang diinginkan ke halaman pengunggahan berkas. Pengguna juga dapat memberikan nama kepada berkas yang diunggah.

The screenshot shows the Moodle file upload interface. It is divided into three main sections: 'Content', 'Options', and 'Common module settings'. The 'Content' section features a 'Select files' window with a file manager interface. The 'Options' section includes settings for 'Display' (Automatic), 'Show size', 'Show type', 'Display resource name', and 'Display resource description'. The 'Common module settings' section includes 'Visible' (Show) and 'ID number'. At the bottom, there are buttons for 'Save and return to course', 'Save and display', and 'Cancel'. A red asterisk indicates that some fields are required.

Gambar 5.2 Uji Coba Mengunggah Berkas

5.2.3 Uji Coba Melakukan Reservasi Kelas Virtual BigBlueButton pada Moodle

Adding a new BigBlueButtonBN to Teknik Pengolahan Data

General settings

Virtual classroom name*

Welcome message

Open

BigBlueButton in a new window

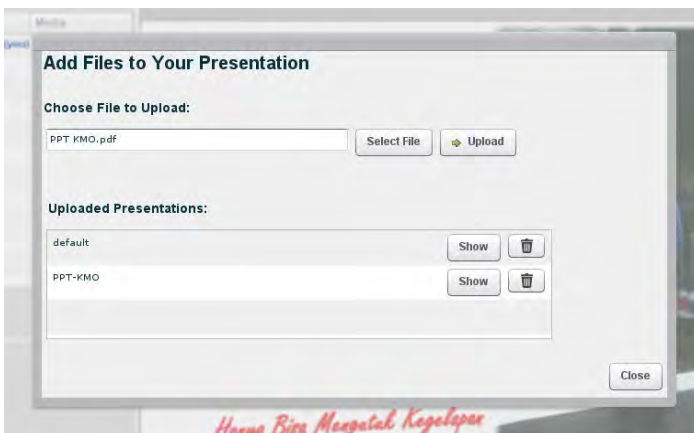
Students must wait until a moderator joins

Gambar 5.3 Uji Coba Reservasi Kelas Virtual

Bagi pengguna yang memiliki status pengajar, maka pengguna tersebut dapat melakukan reservasi kelas BigBlueButton untuk course mata kuliah yang dikelola. Seperti yang dapat dilihat pada Gambar 5.3, pengguna dapat memberikan nama kelas, pesan yang ingin disampaikan diawal sesi kelas, melakukan checklist pada fitur pengajar harus masuk terlebih dulu sebelum siswa dan membuka sesi kelas virtual pada jendela baru di peramban.

5.2.4 Uji Coba Mengunggah Berkas Presentasi pada BigBlueButton

Bagi para pengajar atau siswa yang akan melakukan presentasi, dapat mengunggah dan membagikan presentasi yang dimiliki kepada para pengguna lainnya dengan menekan tombol unggah presentasi pada kiri bawah panel presentasi. Setelah itu, pengguna dapat memilih berkas yang diinginkan untuk diunggah seperti yang ditunjukkan pada Gambar 5.4. Berkas presentasi yang dapat diunggah berupa tipe PDF dengan ukuran maksimal 30 Mb.



Gambar 5.4 Uji Coba Unggah Presentasi

5.2.5 Uji Coba *Editing* Berkas Presentasi

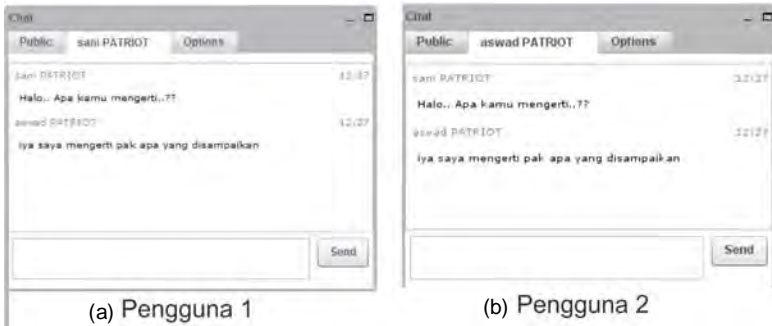
Setelah mengunggah berkas presentasi, pengguna juga diberikan fitur untuk dapat melakukan editing terhadap berkas presentasi yang diunggah sesuai dengan kebutuhan. Pada bagian kanan panel presentasi, terdapat toolbox untuk melakukan editing pada berkas presentasi. Pengguna dapat menambahkan bentuk atau teks untuk membantu pengguna dalam menjelaskan presentasi kepada para pengguna lainnya seperti yang disebutkan pada Gambar 5.5.



Gambar 5.5 Uji Coba Editing pada Berkas Presentasi

5.2.6 Uji Coba Menggunakan Fitur Chat BigBlueButton

Pada saat pengguna berada dalam sesi kelas virtual BigBlueButton, pengguna dapat menggunakan melakukan chatting kepada pengguna lain yang berada dalam satu ruang kelas virtual. Pengguna dapat menggunakan panel chat untuk memilih siapa yang akan dikirim pesan atau menyebarkan pesan ke semua pengguna seperti yang ditunjukkan pada Gambar 5.6.



Gambar 5.6 Uji Coba Chat antara Dua Pengguna: (a) Pengguna 1, (b) Pengguna 2

5.2.7 Uji Coba Bergabung Dalam Konferensi Suara

Pada saat awal pengguna masuk kedalam kelas virtual, pengguna akan dihadapkan pada pilihan untuk bergabung dalam konferensi suara atau tidak, seperti yang dapat dilihat pada Gambar 5.7. Jika pengguna tidak ingin langsung bergabung dalam konferensi suara, maka setelah pengguna keluar dari halaman pilihan untuk bergabung dalam konferensi suara, pengguna dapat menekan tombol mikrofon yang terdapat di kiri atas halaman kelas virtual untuk bergabung dalam konferensi suara.



Gambar 5.7 Uji Coba Bergabung dengan Konferensi Suara

5.2.8 Uji Coba Berbagi Kamera Web

Pada saat berada dalam sesi kelas virtual, pengguna dapat membagikan kamera *web* dari komputer kepada pengguna lain dalam satu ruang kelas virtual seperti pada Gambar 5.8. Untuk melakukan hal tersebut, pengguna dapat menekan tombol kamera *web* yang terdapat pada kiri atas halaman kelas virtual. Sebelum membagikan kamera *web* yang dimiliki, pengguna juga dapat mengatur perangkat keras yang digunakan serta resolusi dari kamera *web* yang ingin dibagikan.



Gambar 5.8 Uji Coba Membagikan Kamera Web

5.3 Uji Coba Performa

Pada uji coba performa sistem perangkat lunak ini akan dilakukan dua macam pengujian, yaitu:

- Uji coba kompresi dinamis kamera *web*.
- Uji coba kemampuan server dalam menangani jumlah pengguna.

5.3.1 Uji Coba Kompresi Dinamis Kamera Web

Pada uji coba ini, dilakukan oleh dua komputer klien yang digunakan untuk mengirimkan kamera *web* dan menerima hasil kiriman kamera *web*. Kamera *web* yang dibagikan pada uji coba ini memiliki resolusi 640x480 pixels.

Ada tiga kondisi yang akan mengubah nilai kompresi kamera *web*, yaitu:

- Kecepatan akses internet pengguna berada pada 0 - 6 Mbps diberikan nilai kompresi 50% dari kualitas semula.
- Kecepatan akses internet pengguna berada pada 6,1 – 7,5 Mbps diberikan nilai kompresi 70% dari kualitas semula.
- Kecepatan akses internet pengguna berada > 7,5 Mbps diberikan nilai kompresi 90% dari kualitas semula.

5.3.1.1 Uji Coba Nilai Kompresi Pada Kecepatan Akses Internet 0-6 Mbps

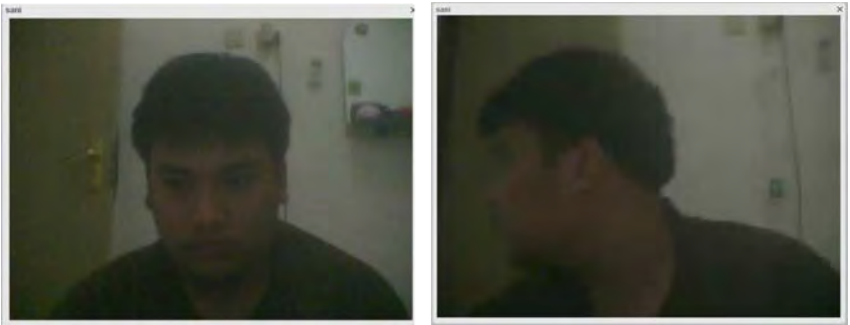
Pada Gambar 5.9, dapat dilihat kamera web diberikan nilai kompresi sebesar 50% dari kualitas semula yang diterapkan pada pengguna dengan kecepatan akses 0-6 Mbps.

5.3.1.2 Uji Coba Nilai Kompresi Pada Kecepatan Akses Internet 6,1-7,5 Mbps

Pada Gambar 5.10, dapat dilihat kamera web diberikan nilai kompresi sebesar 70% dari kualitas semula yang diterapkan pada pengguna dengan kecepatan akses 6,1-7,5 Mbps.

5.3.1.3 Uji Coba Nilai Kompresi Pada Kecepatan Akses Internet > 7,5 Mbps

Pada Gambar 5.11, dapat dilihat kamera web diberikan nilai kompresi sebesar 90% dari kualitas semula yang diterapkan pada pengguna dengan kecepatan akses > 7,5 Mbps..



Pengirim

Penerima

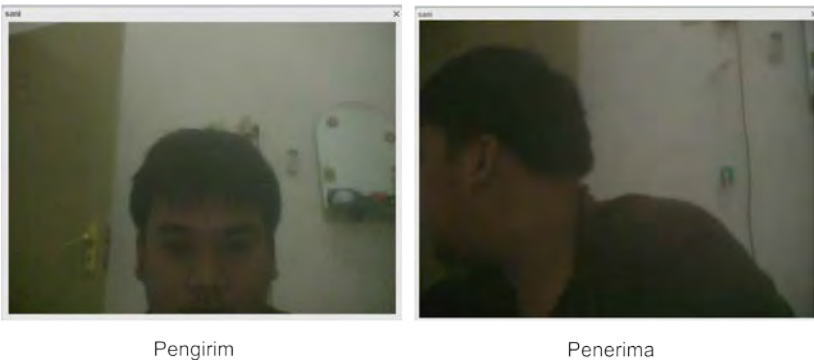
Gambar 5.9 Uji Coba Kompresi Kamera *Web* dengan Kecepatan Akses 0-6 Mbps



Pengirim

Penerima

Gambar 5.10 Uji Coba Kompresi Kamera *Web* dengan Kecepatan Akses 6,1-7,5 Mbps



Gambar 5.11 Uji Coba Kompresi Kamera *Web* dengan Kecepatan Akses > 7,5 Mbps

5.3.2 Uji Coba Kemampuan Server dalam Menangani Jumlah Pengguna Kelas Virtual

Pada uji coba kemampuan server saat menjalankan aplikasi perangkat lunak ini, server menggunakan 36% dari total memori server untuk menjalankan aplikasi perangkat lunak ini. Nilai ini dapat dilihat pada Tabel 5.1 yang menjelaskan kondisi awal dari server sebelum menjalankan aplikasi perangkat lunak dan setelah server menjalankan aplikasi perangkat lunak.

Untuk melihat kemampuan server dalam menangani pengguna kelas virtual, maka akan dilakukan skenario uji coba dengan menggunakan 5, 10 dan 20 pengguna kelas virtual dimana masing-masing pengguna melakukan *streaming* audio dan video serta menerima presentasi yang diunggah presenter.

5.3.2.1 Uji Coba Kemampuan Server dengan 5 Pengguna

Pada uji coba kemampuan server dengan 5 pengguna kelas virtual, memori server bekerja hingga 77% dari total memori server. Hal ini ditunjukkan pada Tabel 5.1, dimana dapat dilihat bahwa

terjadi peningkatan penggunaan memori server kurang lebih sebesar 100 Mb untuk menangani 5 pengguna kelas virtual.

5.3.2.2 Uji Coba Kemampuan Server dengan 10 Pengguna

Pada uji coba kemampuan server dengan 10 pengguna kelas virtual, memori server bekerja hingga 79% dari total memori server. Hal ini ditunjukkan pada Tabel 5.1, dimana dapat dilihat terjadi peningkatan penggunaan memori server hingga 23 Mb dari sebelumnya untuk menangani 10 pengguna kelas virtual.

5.3.2.3 Uji Coba Kemampuan Server dengan 20 Pengguna

Pada uji coba kemampuan server dengan 20 pengguna kelas virtual, memori server bekerja hingga 82% dari total memori server. Hal ini ditunjukkan pada Tabel 5.1, dimana dapat dilihat terjadi peningkatan penggunaan memori server hingga 70 Mb dari sebelumnya untuk menangani 20 pengguna kelas virtual.

Tabel 5.1 Uji Coba Kemampuan Server

No	Case	Memory Performance (kb)				CPU Usage
		Total	Used	Free	Buffer	
1	Awal (sistem belum berjalan)	1.982.820	730.840	1.251.980	54.396	37%
2	Awal (sistem sudah)	1.982.820	1.438.536	544.284	54.844	73%

	berjalan)					
3	5 pengguna	1.982.820	1.535.208	447.612	98.276	77%
4	10 pengguna	1.982.820	1.558.192	424.628	99.204	79%
5	20 pengguna	1.982.820	1.628.336	354.484	100.376	82%

5.3.3 Uji Coba Besar Penggunaan Bandwidth pada Kelas Virtual

Untuk melihat besar penggunaan bandwidth pada kelas virtual, maka akan dilakukan uji coba dengan skenario menggunakan 5, 10, dan 20 pengguna, dimana masing-masing pengguna melakukan *streaming* audio dan video serta diberikan nilai kompresi kamera *web* sebesar 90 persen dari kualitas semula.

5.3.3.1 Uji Coba Penggunaan Bandwidth dengan 5 Pengguna

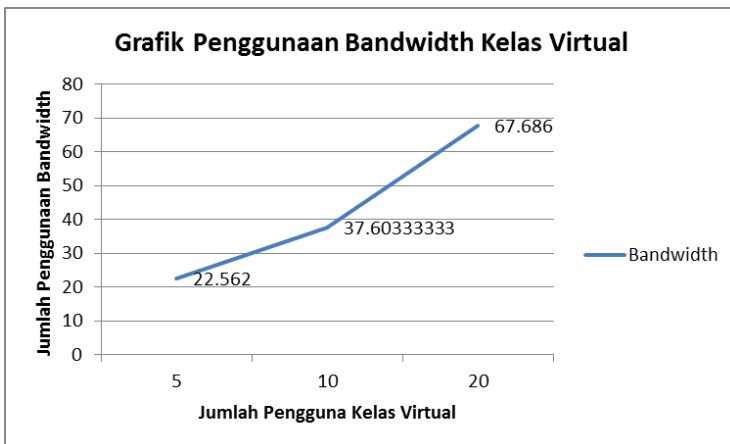
Pada uji coba penggunaan bandwidth dengan 5 pengguna kelas virtual, sistem menggunakan bandwidth sebesar 22,562 Mbps seperti yang ditunjukkan pada Gambar 5.12.

5.3.3.2 Uji Coba Penggunaan Bandwidth dengan 10 Pengguna

Pada uji coba penggunaan bandwidth dengan 10 pengguna kelas virtual, sistem menggunakan bandwidth sebesar 37,603 Mbps seperti yang ditunjukkan pada Gambar 5.12. Terjadi peningkatan jumlah penggunaan bandwidth hingga 15 Mbps dari kondisi 5 pengguna.

5.3.3.3 Uji Coba Penggunaan Bandwidth dengan 20 Pengguna

Pada uji coba penggunaan bandwidth dengan 20 pengguna kelas virtual, sistem menggunakan bandwidth sebesar 67,686 Mbps seperti yang ditunjukkan pada Gambar 5.12. Terjadi peningkatan jumlah penggunaan bandwidth hingga 30 Mbps dari kondisi 10 pengguna.



Gambar 5.12 Uji Coba Besar Penggunaan Bandwidth Kelas Virtual

BAB VI

PENUTUP

Pada bab ini akan dibahas mengenai kesimpulan yang dapat diambil dari perancangan sistem hingga hasil pengujian. Selain itu juga akan dibahas mengenai hasil yang sudah dicapai dan belum dicapai. Pada bab ini juga akan menjawab pertanyaan yang dikemukakan pada Bab 1. Dan yang terakhir terdapat beberapa saran untuk pengembangan selanjutnya.

6.1 Kesimpulan

Dari hasil pengamatan selama perancangan, pengimplementasian, dan proses uji coba sistem didapatkan kesimpulan sebagai berikut.

- Sistem perangkat lunak ini telah dapat mengintegrasikan dua perangkat lunak, yaitu Moodle yang merupakan aplikasi pembelajaran *online* dan BigBlueButton yang merupakan aplikasi *web conferencing*.
- Sistem telah dapat mengubah nilai kompresi kamera *web* yang digunakan pengguna untuk disebarkan kepada pengguna lain berdasarkan kecepatan akses internet pengguna, dimana pada saat kecepatan 0-6 Mbps menjadi 50% dari kualitas semula, 6,1-7,5 Mbps menjadi 70% dari kualitas semula, dan lebih dari 7,5 Mbps menjadi 90% dari kualitas semula.
- Untuk dapat melakukan eksekusi berkas PHP pada server BigBlueButton membutuhkan konfigurasi lanjut pada Nginx, yakni dengan memberikan konfigurasi *fastcgi* pada konfigurasi standar BigBlueButton.
- Untuk dapat menjalankan sistem perangkat lunak ini dibutuhkan kapasitas memori yang besar, karena untuk

menjalankan sistem dapat menggunakan 36% dari total memori yang tersedia dan untuk menangani 20 pengguna, sistem mendorong performa server hingga 82%.

- Untuk menjalankan fitur kelas virtual secara optimal, butuh penyediaan infrastruktur yang dapat memfasilitasi penggunaan bandwidth hingga 70 Mbps, karena dalam menangani 20 pengguna, sistem menggunakan bandwidth sebesar 68 Mbps.

6.2 Saran

Setelah melakukan perancangan, pengimplementasian dan pengujian, ditemui kekurangan pada sistem yang sudah dibuat. Terdapat beberapa saran yang dapat dilakukan untuk mengurangi kekurangan tersebut.

- Menggunakan server dengan spesifikasi yang lebih besar, terutama pada prosesor dan memori karena sistem perangkat lunak ini menggunakan kapasitas yang cukup tinggi.

DAFTAR PUSTAKA

- [1] Margaret Rouse. (2005, September) SerachCIO. [Online]. <http://searchcio.techtarget.com/definition/learning-management-system>
- [2] Mindflash. (2013) Mindflash. [Online]. <http://www.mindflash.com/learning-management-systems/what-is-lms>
- [3] Liu Murong, "Overview and Appliance of Some Streaming Media ," Lahti University of Applied Sc iences, Bachelor's Thesis 2013.
- [4] Gary McGath. (2013, May) garymcgath.com. [Online]. <http://www.garymcgath.com/streamingprotocols.html>
- [5] DataArt. (2013) DataArt. [Online]. <http://blog.dataart.com/real-time-messaging-protocol/>
- [6] Kevin Wallace, "VoIP Fundamentals," in *Authorized Self-Study Guide Cisco Voice over IP (CVoice)*:.: Cisco Press, 2006, ch. 5, p. 504.
- [7] TheoryApp. (2013, January) TheoryApp. [Online]. <http://theoryapp.com/ajax-asynchronous-javascript-and-xml/>
- [8] About.com. (2013) About.com. [Online]. <http://webdesign.about.com/od/ajax/a/aa101705.html>
- [9] About-Moodle. (2014, January) Moodle. [Online]. [http://docs.moodle.org/26/en/About Moodle](http://docs.moodle.org/26/en/About_Moodle)
- [10] History-Moodle. (2014, January) Moodle. [Online]. <http://docs.moodle.org/26/en/History>
- [11] Philosophy-Moodle. (2014, January) Moodle. [Online]. <http://docs.moodle.org/26/en/Philosophy>
- [12] Moodle. (2013, April) Moodle. [Online]. [http://docs.moodle.org/26/en/Moodle site - basic structure](http://docs.moodle.org/26/en/Moodle_site_-_basic_structure)

- [13] History-BigBlueButton. (2013) BigBlueButton. [Online]. <http://bigbluebutton.org/history/>
- [14] BigBlueButton. (2012) BigBlueButton. [Online]. <http://bigbluebutton.org/components/>
- [15] Fred Dixon. (2013, November) BigBlueButton-Google Project Hosting. [Online]. <https://code.google.com/p/bigbluebutton/wiki/ArchitectureOverview>
- [16] Fred Dixon. (2013, November) BigBlueButton - Google Project House. [Online]. http://bigbluebutton.googlecode.com/svn/trunk/bbb-images/arch_Diagrams/architecture_diagram_08.png
- [17] BigBlueButton Inc. (2012) BigBlueButton. [Online]. " <http://www.bigbluebutton.org/>
- [18] Nurul Fithria Lubis, "Penggunaan Pohon Huffman pada Algoritma Deflate," Institut Teknologi Bandung, Bandung, 2011.
- [19] Aditya Mahendra, "Rancang Bangun Aplikasi Kompresi File Menggunakan Metode Deflate," Universitas Islam Negeri Maulana Malik Ibrahim, Malang, 2012.
- [20] Maximum Compression (2011) www.maximumcompression.com. [Online]. http://www.maximumcompression.com/lossless_vs_lossy.php

BIODATA PENULIS



Resto Ajie Suyanto, lebih akrab disapa Ajie, lahir di kota Sangatta, Kalimantan Timur pada tanggal 6 September 1991. Penulis merupakan anak pertama dan satu-satunya. Selama hidup hingga saat ini, penulis telah menjajaki tiga kota, yakni Sangatta, Balikpapan, dan Surabaya.

Menempuh pendidikan di SD YPPSB Sangatta, SD Latihan YBBSU Balikpapan. Kemudian melanjutkan ke SMP Negeri 1 Balikpapan, dan setelah itu melanjutkan di SMA Negeri 1 Balikpapan. Setelah lulus dari SMA, penulis melanjutkan ke jenjang perguruan tinggi di Institut Teknologi Sepuluh Nopember Surabaya di Fakultas Teknologi Informasi Jurusan Teknik Informatika.

Selain pada bidang teknologi informasi, penulis memiliki ketertarikan pada dunia desain grafis, kepelatihan dan organisasi, yang diaktualisasikan selama dunia perkuliahan. Penulis aktif di Himpunan Mahasiswa Teknik Computer-Informatika (HMTC) sebagai Wakil Ketua periode 2011-2012, dalam dunia kepelatihan penulis aktif sebagai Pemandu LKMM tingkat fakultas dan Pemandu Patriot LKMM TM ITS 2013.

Prestasi yang pernah diraih penulis salah satunya adalah menjadi ketua tim dengan proposal PKM-M 2013 di danai dengan judul “Traffic Educational Program (Sosialisasi Tata Tertib Lalu Lintas) kepada Siswa SD di Surabaya sebagai Dasar Pembentukan Generasi Muda Sadar Hukum”

Penulis dapat dihubungi melalui *email* di restoajie@gmail.com. Selain itu, penulis juga dapat dihubungi melalui sosial media seperti Facebook pada Resto Ajie Suyanto atau Twitter pada @restoajie.

LAMPIRAN

A. Data Throughput Kelas Virtual pada Nilai Kompresi 10

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbits/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	896	7.34	3.979	0	624	0
3	1.0 - 2.0	965	7.9	3.177	1	673	0.14858841
3	2.0 - 3.0	887	7.27	2.217	1	619	0.161550889
3	3.0 - 4.0	902	7.39	1.289	0	628	0
3	4.0 - 5.0	999	8.18	1.201	0	696	0
3	5.0 - 6.0	896	7.34	2.169	0	624	0
3	6.0 - 7.0	960	7.87	4.196	1	670	0.149253731
3	7.0 - 8.0	1010	8.48	4.781	0	721	0
3	8.0 - 9.0	953	7.81	2.815	0	664	0
3	9.0 - 10.0	784	6.42	2.503	0	546	0
	0.0 - 10.0	9252	7.6	2.8327	3	6465	0.045939303

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbits/sec]	Jitter [ms]	Lost	Total	Datagram [%]
4	0.0 - 1.0	891	7.3	2.259	0	621	0
4	1.0 - 2.0	742	6.08	3.819	1	518	0.193050193
4	2.0 - 3.0	856	7.01	3.263	0	596	0
4	3.0 - 4.0	758	6.21	3.958	1	529	0.189035917
4	4.0 - 5.0	428	3.5	2.997	2	300	0.666666667
4	5.0 - 6.0	448	3.67	7.301	1	313	0.319488818
4	6.0 - 7.0	436	3.58	17.483	1	305	0.327868852
4	7.0 - 8.0	490	4.01	18.43	4	345	1.15942029
4	8.0 - 9.0	533	4.36	2.34	1	372	0.268817204
4	9.0 - 10.0	952	7.8	2.396	0	663	0
	0.0 - 10.0	6534	5.352	6.4246	11	4562	0.312434794

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbps/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	835	6.84	6.17	1	583	0.171526587
3	1.0 - 2.0	982	8.04	2.745	0	684	0
3	2.0 - 3.0	960	7.87	2.692	0	669	0
3	3.0 - 4.0	952	7.8	1.795	0	663	0
3	4.0 - 5.0	998	8.17	2.5	0	695	0
3	5.0 - 6.0	952	7.8	3.096	0	663	0
3	6.0 - 7.0	808	6.62	2.203	0	563	0
3	7.0 - 8.0	757	6.2	2.756	0	527	0
3	8.0 - 9.0	853	6.99	3.324	0	594	0
3	9.0 - 10.0	807	6.61	2.508	0	562	0
	0.0 - 10.0	8904	7.294	2.9789	1	6203	0.017152659

B. Data Throughput Kelas Virtual pada Nilai Kompresi 20

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbps/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	970	7.95	2.785	1	677	0.147710487
3	1.0 - 2.0	1009	8.27	1.959	0	703	0
3	2.0 - 3.0	788	6.46	2.52	1	550	0.181818182
3	3.0 - 4.0	1000	8.4	2.862	0	714	0
3	4.0 - 5.0	972	7.96	3.171	0	677	0
3	5.0 - 6.0	823	6.74	4.128	2	575	0.347826087
3	6.0 - 7.0	932	7.63	2.392	0	649	0
3	7.0 - 8.0	930	7.62	4.164	0	648	0
3	8.0 - 9.0	923	7.56	1.979	0	643	0
3	9.0 - 10.0	953	7.81	2.563	1	665	0.15037594
	0.0 - 10.0	9300	7.64	2.8523	5	6501	0.08277307

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbits/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	729	5.97	1.945	2	510	0.392156863
3	1.0 - 2.0	769	6.3	2.351	0	536	0
3	2.0 - 3.0	708	5.8	5.538	0	493	0
3	3.0 - 4.0	624	5.12	3.617	0	435	0
3	4.0 - 5.0	844	6.91	3.527	0	588	0
3	5.0 - 6.0	937	7.68	2.389	0	653	0
3	6.0 - 7.0	1022	8.37	2.743	0	712	0
3	7.0 - 8.0	867	7.1	2.094	0	604	0
3	8.0 - 9.0	906	7.42	2.083	1	632	0.158227848
3	9.0 - 10.0	723	6.5	1.965	0	532	0
	0.0 - 10.0	8129	6.717	2.8252	3	5695	0.055038471

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbits/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	900	7.37	2.692	0	627	0
3	1.0 - 2.0	912	7.47	1.884	0	635	0
3	2.0 - 3.0	1011	8.28	1.919	0	704	0
3	3.0 - 4.0	1070	8.98	1.713	1	765	0.130718954
3	4.0 - 5.0	989	8.1	2.698	0	689	0
3	5.0 - 6.0	810	6.63	6.498	0	689	0
3	6.0 - 7.0	810	6.63	4.007	0	564	0
3	7.0 - 8.0	830	6.8	3.816	1	565	0.17699115
3	8.0 - 9.0	797	6.53	2.458	0	555	0
3	9.0 - 10.0	802	6.57	2.908	0	559	0
	0.0 - 10.0	8931	7.336	3.0593	2	6352	0.03077101

C. Data Throughput Kelas Virtual pada Nilai Kompresi 30

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbits/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	949	7.77	1.692	0	661	0
3	1.0 - 2.0	856	7.01	3.84	0	596	0
3	2.0 - 3.0	937	7.68	2.019	1	654	0.152905199
3	3.0 - 4.0	985	8.07	1.967	6	686	0.874635569
3	4.0 - 5.0	794	6.5	5.116	0	553	0
3	5.0 - 6.0	775	6.35	2.905	0	540	0
3	6.0 - 7.0	761	6.23	3.217	0	530	0
3	7.0 - 8.0	790	6.47	2.52	0	550	0
3	8.0 - 9.0	771	6.32	4.707	0	537	0
3	9.0 - 10.0	834	6.83	2.086	0	581	0
	0.0 - 10.0	8452	6.923	3.0069	7	5888	0.102754077

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbits/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	856	7.01	1.524	2	598	0.334448161
3	1.0 - 2.0	777	6.36	3.381	0	541	0
3	2.0 - 3.0	821	6.73	2.464	1	573	0.17452007
3	3.0 - 4.0	728	5.96	4.023	1	508	0.196850394
3	4.0 - 5.0	577	4.73	3.93	2	404	0.495049505
3	5.0 - 6.0	680	5.57	5.98	0	474	0
3	6.0 - 7.0	678	5.55	3.911	0	472	0
3	7.0 - 8.0	718	5.88	4.613	0	500	0
3	8.0 - 9.0	741	6.07	3.451	0	516	0
3	9.0 - 10.0	679	5.56	3.82	0	473	0
	0.0 - 10.0	7255	5.942	3.7097	6	5059	0.120086813

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbits/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	857	7.02	1.162	1	598	0.16722408
3	1.0 - 2.0	834	6.83	1.484	1	582	0.171821306
3	2.0 - 3.0	1016	8.33	4.419	1	709	0.141043724
3	3.0 - 4.0	881	7.22	7.869	0	614	0
3	4.0 - 5.0	930	7.62	5.759	0	648	0
3	5.0 - 6.0	975	7.99	3.484	0	679	0
3	6.0 - 7.0	902	7.39	1.932	1	629	0.158982512
3	7.0 - 8.0	924	7.57	2.752	0	644	0
3	8.0 - 9.0	983	8.06	2.33	0	685	0
3	9.0 - 10.0	966	7.91	3.654	1	674	0.148367953
	0.0 - 10.0	9268	7.594	3.4845	5	6462	0.078743957

D. Data Throughput Kelas Virtual pada Nilai Kompresi 40

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbits/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	906	7.42	3.142	0	631	0
3	1.0 - 2.0	848	6.95	1.75	36	627	5.741626794
3	2.0 - 3.0	993	8.14	1.065	0	692	0
3	3.0 - 4.0	985	8.07	4.56	0	686	0
3	4.0 - 5.0	973	7.97	6.025	0	678	0
3	5.0 - 6.0	870	7.13	3.339	0	606	0
3	6.0 - 7.0	861	7.06	3.804	0	600	0
3	7.0 - 8.0	880	7.21	1.432	0	613	0
3	8.0 - 9.0	817	6.69	2.531	0	569	0
3	9.0 - 10.0	768	6.29	3.828	0	535	0
	0.0 - 10.0	8901	7.293	3.1476	36	6237	0.574162679

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbits/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	926	7.59	4.738	0	645	0
3	1.0 - 2.0	751	6.15	4.956	2	525	0.380952381
3	2.0 - 3.0	744	6.09	3.03	1	519	0.192678227
3	3.0 - 4.0	725	5.94	3.045	1	506	0.197628458
3	4.0 - 5.0	778	6.37	3.752	0	542	0
3	5.0 - 6.0	876	7.17	3.028	0	610	0
3	6.0 - 7.0	825	6.76	5.144	0	575	0
3	7.0 - 8.0	807	6.61	3.015	0	562	0
3	8.0 - 9.0	748	6.13	2.473	0	521	0
3	9.0 - 10.0	758	6.21	2.296	0	528	0
	0.0 - 10.0	7938	6.502	3.5477	4	5533	0.077125907

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbits/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	676	5.54	5.371	0	471	0
3	1.0 - 2.0	767	6.28	3.432	0	534	0
3	2.0 - 3.0	749	6.14	3.242	0	522	0
3	3.0 - 4.0	771	6.32	4.321	1	538	0.185873606
3	4.0 - 5.0	858	7.03	2.838	0	598	0
3	5.0 - 6.0	820	6.71	2.412	0	571	0
3	6.0 - 7.0	795	6.52	2.396	0	554	0
3	7.0 - 8.0	818	6.7	3.047	0	570	0
3	8.0 - 9.0	825	6.76	2.321	0	575	0
3	9.0 - 10.0	709	5.81	2.806	0	494	0
	0.0 - 10.0	7788	6.381	3.2186	1	5427	0.018587361

E. Data Throughput Kelas Virtual pada Nilai Kompresi 50

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbps/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	917	7.51	4.538	2	641	0.31201248
3	1.0 - 2.0	932	7.63	2.016	0	649	0
3	2.0 - 3.0	976	8	1.732	0	680	0
3	3.0 - 4.0	775	6.35	3.622	0	540	0
3	4.0 - 5.0	771	6.32	1.958	0	537	0
3	5.0 - 6.0	676	5.54	3.907	1	472	0.211864407
3	6.0 - 7.0	693	5.68	4.586	0	483	0
3	7.0 - 8.0	670	5.49	4.131	0	467	0
3	8.0 - 9.0	676	5.54	4.628	0	471	0
3	9.0 - 10.0	676	5.54	4.415	0	471	0
	0.0 - 10.0	7762	6.36	3.5533	3	5411	0.052387689

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbps/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	508	4.16	2.055	7	361	1.939058172
3	1.0 - 2.0	548	4.49	2.878	94	476	19.74789916
3	2.0 - 3.0	815	6.68	2.215	1	569	0.175746924
3	3.0 - 4.0	873	7.15	2.673	0	608	0
3	4.0 - 5.0	805	6.6	3.167	0	561	0
3	5.0 - 6.0	893	7.31	2.107	1	623	0.160513644
3	6.0 - 7.0	922	7.55	2.8	0	642	0
3	7.0 - 8.0	942	7.71	1.942	1	657	0.152207002
3	8.0 - 9.0	893	7.31	2.847	0	622	0
3	9.0 - 10.0	883	7.23	2.995	0	615	0
	0.0 - 10.0	8082	6.619	2.5679	104	5734	2.21754249

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbits/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	820	6.71	5.879	0	571	0
3	1.0 - 2.0	884	7.24	2.844	0	616	0
3	2.0 - 3.0	732	6	4.028	2	512	0.390625
3	3.0 - 4.0	721	5.9	6.667	0	502	0
3	4.0 - 5.0	650	5.33	5.192	0	453	0
3	5.0 - 6.0	676	5.54	3.041	2	473	0.422832981
3	6.0 - 7.0	769	6.3	4.762	0	536	0
3	7.0 - 8.0	807	6.61	2.226	0	562	0
3	8.0 - 9.0	732	6	2.681	0	510	0
3	9.0 - 10.0	802	6.57	1.62	0	559	0
	0.0 - 10.0	7593	6.22	3.894	4	5294	0.081345798

F. Data Throughput Kelas Virtual pada Nilai Kompresi 60

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbits/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	705	5.77	4.757	0	491	0
3	1.0 - 2.0	963	7.89	2.893	0	671	0
3	2.0 - 3.0	1020	8.56	1.454	0	728	0
3	3.0 - 4.0	791	6.48	2.335	0	551	0
3	4.0 - 5.0	1070	8.95	1.565	0	761	0
3	5.0 - 6.0	893	7.31	3.428	0	622	0
3	6.0 - 7.0	1021	8.36	4.929	0	711	0
3	7.0 - 8.0	979	8.02	1.716	0	682	0
3	8.0 - 9.0	958	7.84	3.122	1	668	0.149700599
3	9.0 - 10.0	1040	8.7	1.599	0	740	0
	0.0 - 10.0	9440	7.788	2.7798	1	6625	0.01497006

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbits/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	864	7.08	2.868	0	602	0
3	1.0 - 2.0	821	6.73	4.481	0	572	0
3	2.0 - 3.0	785	6.43	2.885	0	547	0
3	3.0 - 4.0	863	7.07	2.184	1	602	0.166112957
3	4.0 - 5.0	583	4.77	3.802	1	407	0.245700246
3	5.0 - 6.0	688	5.63	3.104	0	479	0
3	6.0 - 7.0	761	6.23	3.662	0	530	0
3	7.0 - 8.0	848	6.95	1.926	0	591	0
3	8.0 - 9.0	791	6.48	2.245	0	551	0
3	9.0 - 10.0	771	6.32	4.501	0	537	0
	0.0 - 10.0	7775	6.369	3.1658	2	5418	0.04118132

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbits/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	980	8.03	3.736	0	683	0
3	1.0 - 2.0	969	7.94	1.926	0	675	0
3	2.0 - 3.0	970	7.95	0.851	1	677	0.147710487
3	3.0 - 4.0	1020	8.54	1.633	0	726	0
3	4.0 - 5.0	881	7.22	5.047	0	614	0
3	5.0 - 6.0	866	7.09	1.858	0	603	0
3	6.0 - 7.0	955	7.82	2.421	0	665	0
3	7.0 - 8.0	802	6.57	3.738	1	560	0.178571429
3	8.0 - 9.0	808	6.62	5.337	2	565	0.353982301
3	9.0 - 10.0	794	6.5	5.125	4	557	0.718132855
	0.0 - 10.0	9045	7.428	3.1672	8	6325	0.139839707

G. Data Throughput Kelas Virtual pada Nilai Kompresi 70

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbps/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	926	7.59	4.137	0	645	0
3	1.0 - 2.0	1000	8.42	1.273	0	716	0
3	2.0 - 3.0	927	7.6	3.269	0	646	0
3	3.0 - 4.0	1003	8.22	5.615	0	699	0
3	4.0 - 5.0	1050	8.77	1.342	1	747	0.133868809
3	5.0 - 6.0	985	8.07	3.62	3	686	0.437317784
3	6.0 - 7.0	943	7.73	2.792	1	658	0.151975684
3	7.0 - 8.0	811	6.64	2.033	0	565	0
3	8.0 - 9.0	782	6.41	2.076	0	545	0
3	9.0 - 10.0	876	7.17	3.214	0	610	0
	0.0 - 10.0	9303	7.662	2.9371	5	6517	0.072316228

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbps/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	869	7.11	1.397	0	605	0
3	1.0 - 2.0	1020	8.53	1.87	1	726	0.137741047
3	2.0 - 3.0	1003	8.22	1.857	0	699	0
3	3.0 - 4.0	1020	8.54	3.675	1	727	0.137551582
3	4.0 - 5.0	1080	9.06	1.885	0	770	0
3	5.0 - 6.0	973	7.97	3.046	0	678	0
3	6.0 - 7.0	1030	8.3	5.582	0	706	0
3	7.0 - 8.0	745	6.1	3.819	0	519	0
3	8.0 - 9.0	891	7.3	2.028	0	621	0
3	9.0 - 10.0	833	6.82	3.195	0	580	0
	0.0 - 10.0	9464	7.795	2.8354	2	6631	0.027529263

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbits/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	972	7.96	3.334	0	677	0
3	1.0 - 2.0	889	7.28	2.151	1	620	0.161290323
3	2.0 - 3.0	1000	8.41	2.76	2	717	0.278940028
3	3.0 - 4.0	1009	8.27	1.3	1	704	0.142045455
3	4.0 - 5.0	610	5	4.883	1	426	0.234741784
3	5.0 - 6.0	825	6.76	3.163	0	575	0
3	6.0 - 7.0	853	6.99	3.002	0	594	0
3	7.0 - 8.0	844	6.91	2.722	0	588	0
3	8.0 - 9.0	861	7.06	4.07	0	600	0
3	9.0 - 10.0	886	7.26	2.335	0	617	0
	0.0 - 10.0	8749	7.19	2.972	5	6118	0.081701759

H. Data Throughput Kelas Virtual pada Nilai Kompresi 80

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbits/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	902	7.39	3.363	0	628	0
3	1.0 - 2.0	946	7.75	1.968	0	659	0
3	2.0 - 3.0	1010	8.5	1.982	0	723	0
3	3.0 - 4.0	965	7.9	1.738	0	672	0
3	4.0 - 5.0	960	7.87	3.245	0	669	0
3	5.0 - 6.0	986	8.08	2.439	1	688	0.145348837
3	6.0 - 7.0	825	6.76	3.814	0	575	0
3	7.0 - 8.0	983	8.06	1.959	0	685	0
3	8.0 - 9.0	1008	8.26	4.764	0	702	0
3	9.0 - 10.0	978	8.01	1.662	0	681	0
	0.0 - 10.0	9563	7.858	2.6934	1	6682	0.014534884

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbits/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	1010	8.48	2.94	0	721	0
3	1.0 - 2.0	1070	8.94	2.728	1	761	0.131406045
3	2.0 - 3.0	929	7.61	3.258	1	648	0.154320988
3	3.0 - 4.0	899	7.36	3.008	1	627	0.159489633
3	4.0 - 5.0	838	6.87	1.484	0	584	0
3	5.0 - 6.0	791	6.48	1.773	0	551	0
3	6.0 - 7.0	742	6.08	3.368	0	517	0
3	7.0 - 8.0	824	6.75	2.48	0	574	0
3	8.0 - 9.0	790	6.47	3.23	0	550	0
3	9.0 - 10.0	833	6.82	5.312	0	580	0
	0.0 - 10.0	8726	7.186	2.9581	3	6113	0.044521667

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbits/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	804	6.59	3.914	0	560	0
3	1.0 - 2.0	890	7.29	2.604	0	620	0
3	2.0 - 3.0	795	6.52	3.033	0	554	0
3	3.0 - 4.0	873	7.15	1.572	0	608	0
3	4.0 - 5.0	824	6.75	3.557	0	574	0
3	5.0 - 6.0	858	7.03	4.756	0	598	0
3	6.0 - 7.0	785	6.43	3.597	0	547	0
3	7.0 - 8.0	837	6.86	2.778	0	583	0
3	8.0 - 9.0	879	7.2	2.88	0	612	0
3	9.0 - 10.0	910	7.46	2.183	0	634	0
	0.0 - 10.0	8455	6.928	3.0874	0	5890	0

I. Data Throughput Kelas Virtual pada Nilai Kompresi 90

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbps/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	969	7.94	3.014	0	675	0
3	1.0 - 2.0	670	5.49	5.51	2	469	0.426439232
3	2.0 - 3.0	820	6.71	3.83	1	572	0.174825175
3	3.0 - 4.0	899	7.36	3.837	2	626	0.319488818
3	4.0 - 5.0	780	6.39	3.635	1	544	0.183823529
3	5.0 - 6.0	782	6.41	3.113	0	545	0
3	6.0 - 7.0	827	6.77	2.472	0	576	0
3	7.0 - 8.0	818	6.7	2.363	0	570	0
3	8.0 - 9.0	867	7.1	3.847	0	604	0
3	9.0 - 10.0	814	6.67	5.992	0	567	0
	0.0 - 10.0	8246	6.754	3.7613	6	5748	0.110457675

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbps/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	787	6.44	4.466	0	548	0
3	1.0 - 2.0	942	7.71	2.837	0	656	0
3	2.0 - 3.0	1020	8.57	2.233	0	729	0
3	3.0 - 4.0	927	7.6	3.847	1	647	0.154559505
3	4.0 - 5.0	1010	8.47	1.712	1	721	0.138696255
3	5.0 - 6.0	1040	8.75	3.239	0	744	0
3	6.0 - 7.0	1040	8.75	2.168	0	744	0
3	7.0 - 8.0	1090	9.15	2.687	0	778	0
3	8.0 - 9.0	1050	8.78	3.099	0	747	0
3	9.0 - 10.0	992	8.13	2.683	0	691	0
	0.0 - 10.0	9898	8.235	2.8971	2	7005	0.029325576

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbps/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	894	7.33	3.293	2	625	0.32
3	1.0 - 2.0	1002	8.21	1.189	1	699	0.143061516
3	2.0 - 3.0	893	7.31	3.074	2	624	0.320512821
3	3.0 - 4.0	950	7.79	2.221	4	666	0.600600601
3	4.0 - 5.0	897	7.35	2.997	2	627	0.318979266
3	5.0 - 6.0	662	5.42	5.392	3	464	0.646551724
3	6.0 - 7.0	854	7	1.613	1	596	0.167785235
3	7.0 - 8.0	979	8.02	2.827	0	682	0
3	8.0 - 9.0	1010	8.47	3.541	0	720	0
3	9.0 - 10.0	1050	8.83	4.016	1	752	0.132978723
	0.0 - 10.0	9191	7.573	3.0163	16	6455	0.265046989

J. Data Throughput Kelas Virtual pada Nilai Kompresi 100
(Tanpa Kompresi)

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbps/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	1180	9.89	2.301	0	841	0
3	1.0 - 2.0	998	8.17	2.909	0	695	0
3	2.0 - 3.0	1240	10.4	1.323	0	884	0
3	3.0 - 4.0	1262	10.6	2.509	0	902	0
3	4.0 - 5.0	1154	9.63	1.606	1	820	0.12195122
3	5.0 - 6.0	1106	9.23	2.491	0	785	0
3	6.0 - 7.0	1320	11.1	2.086	0	941	0
3	7.0 - 8.0	1347	11.3	1.382	0	962	0
3	8.0 - 9.0	1343	11.2	2.551	0	955	0
3	9.0 - 10.0	1247	11.3	1.858	0	957	0
	0.0 - 10.0	12197	10.282	2.1016	1	8742	0.012195122

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbits/sec]	Jitter [ms]	Lost	Total	Datagram [%]
4	0.0 - 1.0	980	8.03	2.974	0	683	0
4	1.0 - 2.0	1077	8.95	3.054	0	761	0
4	2.0 - 3.0	1043	8.76	1.902	0	745	0
4	3.0 - 4.0	1064	8.9	3.052	0	757	0
4	4.0 - 5.0	1050	8.8	2.168	0	748	0
4	5.0 - 6.0	1069	8.93	2.272	0	759	0
4	6.0 - 7.0	1094	9.13	2.57	0	776	0
4	7.0 - 8.0	1083	9.06	3.265	0	770	0
4	8.0 - 9.0	871	7.14	3.52	0	607	0
4	9.0 - 10.0	998	8.17	2.508	0	695	0
	0.0 - 10.0	10329	8.587	2.7285	0	7301	0

ID	Interval [sec]	Transfer [Kbytes]	Bandwidth [Mbits/sec]	Jitter [ms]	Lost	Total	Datagram [%]
3	0.0 - 1.0	873	7.15	2.398	0	608	0
3	1.0 - 2.0	1003	8.22	2.297	0	699	0
3	2.0 - 3.0	945	7.74	2.495	0	658	0
3	3.0 - 4.0	1000	8.4	1.893	0	714	0
3	4.0 - 5.0	986	8.08	2.896	0	687	0
3	5.0 - 6.0	958	7.84	2.596	0	667	0
3	6.0 - 7.0	912	7.47	2.751	0	635	0
3	7.0 - 8.0	952	7.8	2.612	0	663	0
3	8.0 - 9.0	857	7.02	2.931	0	597	0
3	9.0 - 10.0	927	7.6	2.9	0	646	0
	0.0 - 10.0	9413	7.732	2.5769	0	6574	0