

**PENERAPAN SISTEM KOMUNIKASI *VIDEO CALL* DENGAN *VOICE*
OVER INTERNET PROTOCOL MENGGUNAKAN BRIKER PADA
RUANG LINGKUP GEDUNG PERKANTORAN**

SKRIPSI

DEVI MEGARITA BR SIBARANI

201401136



**PROGRAM STUDI S-1 ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN 2024**

**PENERAPAN SISTEM KOMUNIKASI *VIDEO CALL* DENGAN *VOICE*
OVER INTERNET PROTOCOL MENGGUNAKAN BRIKER PADA
RUANG LINGKUP GEDUNG PERKANTORAN**

SKRIPSI

**Diajukan untuk melengkapi tugas dan memenuhi syarat memperoleh ijazah
Sarjana Ilmu Komputer**

DEVI MEGARITA BR SIBARANI

201401136



**PROGRAM STUDI S-1 ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN 2024**

PERSETUJUAN

Judul : PENERAPAN SISTEM KOMUNIKASI *VIDEO CALL*
: DENGAN *VOICE OVER INTERNET PROTOCOL*
MENGUNAKAN BRIKER PADA RUANG
LINGKUP GEDUNG PERKANTORAN

Kategori : SKRIPSI

Nama : DEVI MEGARITA BR SIBARANI

Nomor Induk Mahasiswa : 201401136

Program Studi : SARJANA (S-1) ILMU KOMPUTER

Fakultas : ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVRSITAS SUMATERA UTARA

Telah diuji dan dinyatakan lulus di Medan, 24 Juni 2024

Dosen Pembimbing II



Pauzi Ibrahim Nainggolan, S.Komp., M.Sc.
NIP. 198809142020011001

Dosen Pembimbing I



Handrizal S.Si., M.Comp.Sc
NIP. 197706132017061001

Diketahui/Disetujui Oleh
Ketua Program Studi S-1 Ilmu Komputer



Dr. Amalia, S.T., M.T
NIP. 197812212014042001

PERNYATAAN

**PENERAPAN SISTEM KOMUNIKASI *VIDEO CALL* DENGAN *VOICE OVER*
INTERNET PROTOCOL MENGGUNAKAN BRIKER PADA RUANG
LINGKUP GEDUNG PERKANTORAN**

SKRIPSI

Saya mengakui bahwa skripsi ini adalah hasil penelitian saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing telah dicantumkan sumbernya.

Medan, 24 Juni 2024

A handwritten signature in black ink, appearing to read 'Devi Megarita Br Sibarani'.

Devi Megarita Br Sibarani
201401136

PENGHARGAAN

Segala puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa, karena telah memberikan rahmat dan anugrah-Nya, serta senantiasa memberikan kesehatan, kesempatan, bagi penulis sehingga mampu menyelesaikan penyusunan skripsi ini sebagai syarat untuk mendapatkan gelar Sarjana Komputer di Program Studi S-1 Ilmu Komputer, Universitas Sumatera Utara. Judul skripsi ini adalah “Penerapan Sistem Komunikasi *Video Call* Dengan *Voice over Internet Protocol* Menggunakan Briker Pada Ruang Lingkup Gedung Perkantoran”.

Dalam penulisan skripsi ini, penulis banyak mendapatkan bantuan dan bimbingan moril, material, spiritual, maupun administrasi. Oleh karena itu, dalam kesempatan ini penulis ini mengucapkan terima kasih kepada :


1. Bapak Prof. Dr. Muryanto Amin S.Sos., M.Si. selaku Rektor Universitas Sumatera Utara.
2. Ibu Dr. Maya Silvi Lydia B.Sc., M.Sc. selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.
3. Ibu Dr. Amalia, S.T., M.T. selaku Ketua Program Studi S-1 Ilmu Komputer Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.
4. Bapak Handrizal S.Si., M.Comp.Sc selaku Dosen Pembimbing I yang selalu memberikan bimbingan dan menyediakan waktu untuk membimbing penulis selama proses penyusunan skripsi ini.
5. Bapak Pauzi Ibrahim Nainggolan, S.Komp., M.Sc. selaku Dosen Pembimbing II yang telah memberi masukan, serta arahan kepada penulis selama penyusunan skripsi ini.
6. Seluruh bapak dan ibu dosen Fasilkom-TI USU, khususnya dosen Program Studi S-1 Ilmu Komputer yang telah mendidik dan memberi wawasan serta moral yang berharga, baik di bangku perkuliahan maupun setelah lulus.
7. Terkhusus dan yang paling spesial untuk kedua Orang Tua yang teramat dicintai, Bapak A. Sibarani dan Ibu R. Br. Situmorang, yang selalu memberikan doa, dukungan, dan dorongan moril maupun materil kepada penulis, sampai dengan studi ini selesai.
8. Kakak Putri Deswana, S.E dan Abang Iwan Thirta Sibarani, S.T, yang senantiasa memberikan dukungan dan doa yang berharga untuk penulis.

9. Dean Tambunan, selaku orang yang selalu mendoakan, mendukung, serta memberikan motivasi untuk penulis.
10. Sahabat penulis mulai dari awal perkuliahan ‘Calon S.Kom’, Vina Adlina Ramayani (071) dan Azzahra Mumtaza (075), yang sudah banyak memberikan kenangan manis serta dukungan dalam proses penyusunan skripsi ini.
11. Sahabat kos kosan, ‘BTS (Barisan Teman Sejati)’, Holiness, Jessica, dan Yunisa, yang saling memberi semangat, dukungan, dan kenangan manis kepada penulis.
12. Sahabat penulis, ‘Kureng Literasi’, Difanie, Faradhila, Erlin, Ultari yang juga memberikan banyak tawa dan rasa sayang, serta senantiasa memberikan dukungan kepada penulis.
13. Stambuk 2020 khususnya Kom B yang memberikan pengalaman belajar yang berharga dan akan selalu terkenang.
14. Kepada Kak Desi, Bang Juprianto, dan Bang Jeremia yang senantiasa memberikan dukungan, semangat serta bantuan.
15. Pengurus IMILKOM USU periode 2022/2023 khususnya Divisi Dana dan Usaha yang telah memberi pengalaman dengan aktivitas yang beragam dan bekerja sama dengan baik dalam menjalankan satu periode kepengurusan.

Dalam penyusunan skripsi ini penulis menyadari masih terdapat banyak kekurangan. Saran dan kritik yang membangun sangat penulis harapkan untuk meningkatkan kesempurnaan skripsi ini. Kiranya skripsi ini dapat memberikan manfaat bagi penulis, pembaca, dan semua pihak yang berkepentingan dalam skripsi ini.

Medan, 24 Juni2024

Penulis,



Devi Megarita Br Sibarani

PENERAPAN SISTEM KOMUNIKASI *VIDEO CALL* DENGAN *VOICER OVER INTERNET PROTOCOL* MENGGUNAKAN BRIKER PADA RUANG LINGKUP GEDUNG PERKANTORAN

ABSTRAK

Perkembangan teknologi jaringan komputer di era modern telah memberikan dampak yang signifikan, terutama melalui internet dimana mengubah cara berkomunikasi menjadi lebih fleksibel, tanpa batasan ruang, jarak, dan waktu. Komunikasi merupakan hal penting yang menjadi kebutuhan pokok bagi setiap orang di berbagai kondisi, terlebih lagi dalam sebuah perusahaan dimana komunikasi sangat dibutuhkan untuk saling bertukar informasi. Tanpa adanya komunikasi yang baik dapat menyebabkan miskomunikasi antar individu. Aktivitas telepon yang tinggi di perusahaan seringkali menyebabkan melonjaknya tagihan telepon yang harus dibayar, sehingga diperlukan pembangunan komunikasi *video call* untuk meningkatkan efisiensi komunikasi antar karyawan tanpa menggunakan telepon PSTN (*Public Switched Telephone Network*). Penelitian ini bertujuan untuk menciptakan sistem komunikasi *video call* menggunakan teknologi VoIP (*Voice over Internet Protocol*) dengan Briker sebagai *server*. Dengan demikian, teknologi ini menawarkan solusi yang tepat untuk mengurangi pemborosan biaya operasional suatu instansi. Cara kerja dari VoIP yaitu dengan mengubah format suara menjadi digital yang akan dikirimkan melalui jaringan LAN maupun internet. Briker sendiri adalah distro Linux yang di *design* khusus untuk menjadi *server* VoIP sehingga dapat membangun *call center* sendiri dengan menggunakan jaringan LAN. Metode yang digunakan dalam penelitian ini yaitu metode perhitungan terhadap parameter QoS yang diukur berdasarkan jarak. Dengan aplikasi *Wireshark*, paket data yang dikirim dan diterima pada lalu lintas jaringan akan diamati untuk mengukur parameter *Quality of Services* diantaranya *throughput*, *delay*, dan *jitter*. Hasil dari perhitungan parameter QoS memperoleh nilai *throughput* sebesar 10807,48 Kbps, nilai *delay* 1,03 ms, dan nilai *jitter* 1,06 ms. Dari hasil yang diperoleh, dapat disimpulkan bahwa sistem komunikasi VoIP dengan Briker sebagai *server* yang dibangun untuk diimplementasikan pada gedung perkantoran dapat berjalan dengan baik dengan hasil pengujian parameter-parameter QoS sesuai dengan standar yang telah direkomendasikan oleh ITU yaitu kategori bagus.

Kata Kunci: Internet, VoIP, Briker, QoS, *Wireshark*.

IMPLEMENTATION OF A VIDEO CALL COMMUNICATION SYSTEM
WITH VOICE OVER INTERNET PROTOCOL USING BRIKER IN THE
SCOPE OF OFFICE BUILDINGS

ABSTRACT

The development of computer network technology in the modern era has had a significant impact, especially through the internet which changes the way of communicating to be more flexible, without the limitations of space, distance, and time. Communication is an important thing that becomes a basic need for everyone in various conditions, especially in a company where communication is needed to exchange information. Without good communication can lead to miscommunication between individuals. High telephone activity in the company often causes a soaring telephone bill to be paid, so it is necessary to build *video call* communication to improve communication efficiency between employees without using PSTN (*Public Switched Telephone Network*) telephones. This research aims to create a *video call* communication system using VoIP (*Voice over Internet Protocol*) technology with Briker as a *server*. Thus, this technology offers the right solution to reduce the waste of operational costs of an agency. The way VoIP works is by converting the voice format into digital which will be sent over a LAN network or the internet. Briker itself is a Linux distro specifically *designed* to be a VoIP server so that it can build *its own call center* using a LAN network. The method used in this study is the calculation method for QoS parameters measured based on distance. With the *Wireshark* application, data packets sent and received on network traffic will be observed to measure *Quality of Services* parameters including *throughput*, *delay*, and *jitter*. The results of the calculation of QoS parameters obtained a *throughput* value of 10807.48 Kbps, a *delay* value of 1.03 ms, and a *jitter* value of 1.06 ms. From the results obtained, it can be concluded that the VoIP communication system with Briker as a *server* built to be implemented in office buildings can run well with the results of testing QoS parameters in accordance with the standards that have been recommended by ITU namely good category.

Keywords: Internet, VoIP, Briker, QoS, *Wireshark*.

DAFTAR ISI

PERSETUJUAN	ii
PERNYATAAN.....	iii
PENGHARGAAN.....	iv
ABSTRAK	vi
ABSTRACT	vii
DAFTAR ISI.....	viii
DAFTAR TABEL	x
DAFTAR GAMBAR.....	xi
BAB I.....	1
PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	4
1.6 Metodologi Penelitian	4
1.7 Penelitian Relevan.....	5
1.8 Sistematika Penulisan	6
BAB II.....	8
LANDASAN TEORI.....	8
2.1 Video Call.....	8
2.2 LAN.....	8
2.3 Voice over Internet Protocol.....	9
2.4 Briker	9
2.5 PortSIP UC	10
2.6 Wireshark	11
2.7 Access Point.....	12
2.8 Switch	13
2.9 Kabel Unshield Twisted Pair (UTP)	13
2.10 QoS (Quality of Service)	14
2.10.1 Parameter QoS (Quality of Service).....	14
BAB III.....	16

ANALISIS DAN PERANCANGAN.....	16
3.1 Analisis Sistem.....	16
3.1.1 Analisis masalah	16
3.1.2 Analisis kebutuhan.....	18
3.2 Perancangan Sistem.....	19
3.2.1 Diagram Umum.....	19
3.2.2 Use Case Diagram.....	20
3.2.3 Activity Diagram.....	20
3.3 Flowchart (Diagram Alir).....	22
3.3.1 Flowchart Perancangan Sistem IPPBX.....	22
3.3.2 Flowchart of VoIP Network System Using Briker Server	24
BAB IV	26
IMPLEMENTASI DAN PENGUJIAN.....	26
4.1 Implementasi	26
4.1.1 Implementasi Panggilan dari IP ke IP	26
4.1.2 Perancangan Server.....	26
4.1.3 Bentuk Desain Sistem	28
4.2 Pengujian	28
4.2.1 Konfigurasi Briker IPPBX Dalam Pembuatan Server VoIP.....	28
4.2.2 Konfigurasi Access Point	32
4.2.3 Konfigurasi Extension.....	33
4.2.4 Konfigurasi Adapter Wi-Fi	34
4.2.7 Hasil Pengujian Panggilan	39
4.2.8 Pengukuran dan Analisis Parameter QoS berdasarkan Jarak.....	41
BAB V	59
PENUTUP.....	59
5.1 Kesimpulan.....	59
5.2 Saran	59
DAFTAR PUSTAKA	60
CURRICULUM VITAE.....	63

DAFTAR TABEL

Tabel 2.1 Standar Delay berdasarkan ITU-T	15
Tabel 2.2 Standar Jitter berdasarkan ITU-T.....	15
Tabel 4.1 Tabel Alat yang Digunakan	27
Tabel 4.2 Daftar Pengujian Panggilan	39
Tabel 4.3 Hasil Perhitungan Setiap Parameter QoS	56

DAFTAR GAMBAR

Gambar 2.1 Cara Kerja VoIP	9
Gambar 2.2 Topology Briker (Briker Documentation)	10
Gambar 2.3 Tampilan PortSIP	11
Gambar 2.4 Access Point	12
Gambar 2.5 Switch.....	13
Gambar 2.6 Kabel UTP	13
Gambar 3.1 Diagram Umum Sistem.....	19
Gambar 3.2 Use Case Diagram.....	20
Gambar 3.3 Activity Diagram.....	21
Gambar 3.4 Sequence Diagram	22
Gambar 3.5 Flowchart Perancangan Sistem IPPBX.....	23
Gambar 4.1 Proses Perancangan Server	27
Gambar 4.2 Tampilan Rangkaian Perancangan Sistem	28
Gambar 4.3 Tahap awal	29
Gambar 4.4 Pemilihan Mode Install	29
Gambar 4.5 Proses Instalasi	30
Gambar 4.6 Briker berhasil di Install.....	30
Gambar 4.7 Layar Tampilan Login.....	30
Gambar 4.8 Tampilan untuk mengganti IP address Briker.....	31
Gambar 4.9 Custom VMnet	31
Gambar 4.10 Akses Briker melalui Browser	32
Gambar 4.11 <i>Login Access Point</i>	32
Gambar 4.12 <i>Setting Wireless Access Point</i>	32
Gambar 4.13 Halaman Extension	33
Gambar 4.14 Detail Halaman Extension.....	34
Gambar 4.15 Network Connection	35
Gambar 4.16 Ethernet Properties	35
Gambar 4.17 TCP/Ipv4	36
Gambar 4.18 Login PortSIP	37
Gambar 4.19 Tampilan Client VoIP yang sudah teregistrasi pada Laptop.....	37
Gambar 4.20 Tampilan Client yang sudah teregistrasi di Smartphone.....	38

Gambar 4.21 Tampilan Bridge Adapter.....	38
Gambar 4.22 ipconfig /all pada cmd.....	39
Gambar 4.23 Hasil Panggilan Masuk Smartphone Client ke Smartphone Client.....	39
Gambar 4.24 Hasil Panggilan Masuk PC Server ke PC Client.....	40
Gambar 4.25 Hasil Panggilan Keluar Smartphone Client ke PC Client.....	40
Gambar 4.26 Hasil Panggilan Masuk PC Server ke Smartphone Client	41
Gambar 4.27 Jarak 10 Meter.....	41
Gambar 4.28 Jarak 20 meter	42
Gambar 4.29 Jarak 30 meter	43
Gambar 4.30 Jarak 40 meter	44
Gambar 4.31 Jarak 50 meter	45
Gambar 4.32 Jarak 60 meter	46
Gambar 4.33 Jarak 70 meter	47
Gambar 4.34 Jarak 80 meter	48
Gambar 4.35 Jarak 90 meter	49
Gambar 4.36 Jarak 100 meter	50
Gambar 4.37 Jarak 110 meter	51
Gambar 4.38 Jarak 120 meter	52
Gambar 4.39 Jarak 130 meter	53
Gambar 4.40 Jarak 140 meter	54
Gambar 4.41 Jarak 150 meter	55
Gambar 4.42 Hasil Pengukuran Jitter dengan Jarak	57
Gambar 4.43 Hasil Pengukuran Throughput dengan Jarak	57
Gambar 4.44 Hasil Pengukuran Delay dengan Jarak.....	58

BAB I

PENDAHULUAN

1.1 Latar Belakang

Saat ini, perkembangan teknologi jaringan komputer berkembang pesat di zaman modern, hal ini membawa perubahan yang sangat nyata khususnya dalam bidang komunikasi. Dalam konteks ini, keberadaan Internet mempunyai dampak yang signifikan di segala bidang dan banyak digunakan oleh berbagai kelompok di berbagai belahan dunia untuk berkomunikasi (Saputra, 2023).

Interconnected Network atau biasa dikenal dengan internet merupakan sistem jaringan komputer yang saling terkoneksi secara menyeluruh tanpa batas ruang, jarak dan waktu. Akses internet kini semakin mudah dijangkau oleh setiap orang karena meningkatnya jumlah layanan paket yang ditawarkan oleh berbagai (*provider*) penyedia layanan (Nurajizah, 2020).

Untuk mengakses koneksi internet, dibutuhkan layanan dari penyedia internet yang dikenal sebagai *Internet Service Provider* (ISP). Sebuah perusahaan yang menyediakan layanan akses internet disebut dengan ISP dimana biasanya membebankan biaya bulanan kepada pengguna atas layanan yang mereka sediakan (Kusuma, 2020). Dalam hal ini, perlu dipahami bahwa kualitas layanan internet akan berpengaruh langsung pada efisiensi komunikasi di lingkungan perkantoran.

Salah satu kebutuhan mendasar dalam sebuah lingkungan perkantoran adalah komunikasi yang efektif. Dengan ruangan yang tersebar, tantangan utama adalah kurangnya efisiensi komunikasi antar karyawan, seperti lambatnya respon dari karyawan dalam menanggapi permasalahan penting. Disisi lain, penggunaan telepon PSTN (*Public Switched Telephone Network*) memunculkan sebuah permasalahan, dimana biaya yang signifikan bergantung pada peningkatan penggunaan (Fauzi, 2022). Solusi untuk masalah ini yaitu diperlukan adanya pembangunan komunikasi *video call* sebagai salah satu opsi untuk berkomunikasi dengan rekan yang tidak selalu bisa ditemui secara langsung dan anggaran komunikasi telepon dapat diminimalkan.

Seiring kemajuan *Information and Technology Communication* (ITC), menghadirkan sebuah teknologi yang dikenal dengan *Voice over Internet Protocol* (VoIP), sehingga melahirkan inovasi baru dimana memungkinkan komunikasi melalui internet atau jaringan lokal. Teknologi ini bisa diterapkan di berbagai tempat seperti perusahaan, kantor, kampus, atau perumahan dengan menggunakan infrastruktur jaringan yang sudah tersedia. Biasanya, setiap divisi atau ruangan di kantor telah dilengkapi dengan komputer, untuk memfasilitasi komunikasi antar karyawan. Penggunaan komputer menjadi krusial di sini, karena infrastruktur VoIP cukup memerlukan jaringan nirkabel (*wireless*), PC/laptop, dan *smartphone* sebagai perangkat dasar (Adibasyah, 2023).

Penggunaan fitur VoIP untuk melakukan panggilan melalui infrastruktur jaringan data tentunya dapat menekan tarif yang biasanya dikeluarkan suatu instansi ketika memakai sistem telepon tradisional seperti *Public Switched Telephone Network* (PSTN), *Global System for Mobile Communications* (GSM), dan *Code-Division Multiple Access* (CDMA). Dalam menerapkan VoIP, diperlukan Briker sebagai *server* VoIP.

Seorang pengembang aplikasi Briker yaitu Anton Raharja menuturkan yakni “Briker *Open Source* IPPBX adalah proyek Perangkat Lunak Bebas dan *Open Source* untuk membangun Distribusi Linux yang menyediakan layanan terkait telekomunikasi seperti PBX atau IP PBX, dan fitur telepon lainnya”. Dengan versi terbaru 2.1.1 “Merapi”, dapat mempermudah pekerjaan karena fitur multilayanan yang dapat berjalan diatas jaringan IP diantaranya *Call center*, *Voice over Internet Protocol*, *ISDN (Integrated Service Digital Network)*, *Billing System*, *DID (Direct Inward Dialing)*, *ACD (Automatic CallDistribution)*, *Group Hunting*, *Conference Call*, *Call Data Record (CDR)*, *Web- based Management System*, *Fax over IP*, *Call Recording System*, *Voicemail*, *Interactive Voice Response (IVR)*, *Least Cost Routing (LCR)* dan sebagainya. Distributor (distro) Briker merupakan sebuah linux yang digunakan dalam penelitian ini karena khusus dibuat untuk menyediakan server VoIP. Pemilihan distributor ini karena Briker memiliki stabilitas dan keandalan yang bagus serta memiliki salah satu fitur yang paling menarik yaitu LCR yang memungkinkan Briker mencari jalur dengan biaya paling murah.

1.2 Rumusan Masalah

Dari latar belakang diatas, media komunikasi menjadi salah satu aspek penting dalam menjaga komunikasi antar karyawan. Biasanya, media komunikasi yang dipakai untuk gedung perkantoran meliputi telepon, *email*, dan sebagainya, sehingga belum memadai untuk memenuhi kebutuhan komunikasi secara efektif. Untuk mengatasi hal ini, penting untuk meningkatkan sistem komunikasi *video call* antar karyawan di gedung perkantoran. Peningkatan ini dapat dicapai melalui pembangunan VoIP dimana Briker sebagai *server* dalam melakukan panggilan *video call*, yang bertujuan untuk meningkatkan efisiensi komunikasi antar karyawan dan mengurangi pemborosan biaya operasional suatu instansi karena tidak diperlukan lagi komunikasi menggunakan telepon PSTN (*Public Switched Telephone Network*).

1.3 Batasan Masalah

Penelitian ini memiliki beberapa batasan masalah, yakni :

1. Pembangunan VoIP dilakukan untuk penunjang kebutuhan komunikasi dan mempermudah koordinasi antar karyawan.
2. Dalam perancangan jaringan VoIP, sistem operasi Briker digunakan sebagai *server* VoIP.
3. Skema jaringan yang digunakan dalam VoIP hanya menggunakan jaringan lokal.
4. Penggunaan VoIP berbasis aplikasi yang sudah ada.
5. Tidak membahas masalah keamanan rancangan tersebut karena hanya mencakup jaringan lokal yang umumnya hanya memiliki sedikit resiko dan ancaman serangan jaringan.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini, sebagai berikut :

1. Menerapkan sistem komunikasi *video call* berbasis *Voice over Internet Protocol* (VoIP) menggunakan Briker di lingkungan gedung perkantoran.
2. Mengimplementasikan VoIP antara dua *client* dan satu VoIP *server* yang menggunakan Briker sebagai VoIP *server*.

3. Mengukur dan menganalisa kinerja sistem melalui setiap parameter *Quality of Services* yang berguna mengetahui kualitas layanan dari jaringan VoIP, dengan harapan dapat menyediakan solusi komunikasi yang efektif dan efisien di lingkungan perkantoran.

1.5 Manfaat Penelitian

Diharapkan melalui penelitian ini dapat berkontribusi bagi perusahaan perkantoran dalam hal menghemat biaya yang diperlukan saat melakukan komunikasi *video call* antar karyawan dengan pembangunan sistem komunikasi *video call* menggunakan *Voice over Internet Protocol*.

1.6 Metodologi Penelitian

Penelitian ini menerapkan beberapa metode yang digunakan, diantaranya :

1. Studi Pustaka

Informasi yang dikumpulkan pada tahap awal penelitian berasal dari berbagai sumber, termasuk buku, artikel penelitian, jurnal, dan skripsi tentang penerapan sistem komunikasi *video call Voice over Internet Protocol* menggunakan Briker di gedung perkantoran. Tujuannya adalah mendapatkan sebanyak mungkin informasi terkait penelitian yang sedang dilakukan.

2. Pemodelan Sistem

Selama tahap pemodelan sistem, proses perancangan dasar dikerjakan menggunakan Cisco Packet Tracer sebagai dasar topologi jaringan yang akan dibangun. Kemudian dibuat desain dari *use case diagram*, *activity diagram*, *sequence diagram*, dan diagram alir (flowchart).

3. Implementasi Sistem

Dalam tahap ini, membahas apa saja *hardware* (perangkat keras) dan *software* (perangkat lunak) yang digunakan untuk membuat layanan VoIP, serta bagaimana tahapan-tahapan konfigurasi yang akan dirancang menggunakan *server Briker*.

4. Pengujian Sistem

Pengujian sistem dilakukan pada tahap ini untuk mengetahui apakah sistem sudah memenuhi spesifikasi yang diharapkan dari penelitian ini.

5. Dokumentasi Sistem

Untuk tahap ini, peneliti akan melakukan dokumentasi pada setiap tahap yang terjadi dalam penelitian dan laporan penelitian atau skripsi akan menjadi kesimpulan akhir dari penelitian ini.

1.7 Penelitian Relevan

Dibawah ini termasuk penelitian terdahulu yang berkaitan dengan penelitian ini, meliputi:

1. Berdasarkan penelitian (Ahmad Fauzi et al., 2022) dengan judul “Perancangan Jaringan *Voice Over Internet Protocol* (VoIP) menggunakan *Platform Asterisknow*” menjelaskan bahwa penggunaan teknologi VoIP di sebuah perusahaan atau lembaga dapat meningkatkan efisiensi komunikasi dengan menghilangkan kebutuhan menulis pesan panjang atau melakukan *video call* dengan memakai *bandwidth* yang besar. Interaksi antar cabang serta kantor pusat bisa dilakukan dengan menggunakan jaringan IP, yang berarti jaringan PSTN tidak diperlukan lagi, dan mengurangi biaya untuk komunikasi.
2. Berdasarkan penelitian (Antoni, 2023) yang berjudul “*VoIP (Voice over Internet Protocol) Based on Asterisk Using Trixbox at SMKN 1 Bangkinang*” dijelaskan bahwa keputusan mengenai sistem operasi yang ideal untuk server VoIP didasarkan pada pertimbangan kenyamanan, dukungan teknis, dan kesesuaian keseluruhan untuk kebutuhan spesifik organisasi. Kemampuan server Trixbox VoIP untuk menangani panggilan SIP dari banyak pelanggan terdaftar merupakan terobosan dalam kemampuan komunikasi, memfasilitasi komunikasi suara yang lancar dan efisien.
3. Berdasarkan penelitian (Nwaogwugwu et al., 2021) yang berjudul “*A Low-Cost Telephony System for Small/Medium Scale Businesses in Nigeria: An Implementation of VoIP on a LAN Using Mini SIP Server*” menjelaskan bahwa desain *prototype* menggunakan beberapa perangkat keras server miniSIP dan *softphone* Zoiper menunjukkan bahwa VoIP dapat berhasil diimplementasikan secara fleksibel untuk menyediakan layanan berbasis data tambahan melalui panggilan jaringan area lokal berbiaya rendah.
4. Berdasarkan penelitian (Marcel et al., 2020) yang berjudul “*Integration of a Voice Over Internet Protocol (VoIP) Solution with Internet Protocol Version 6*

(Ipv6) in a Internet Protocol Version 4 (Ipv4) Data Network to Increase Employee Productivity” dijelaskan bahwa VoIP ternyata efektif dalam simulasi jaringan Ipv4/Ipv6 dan cukup meyakinkan untuk bertindak sebagai solusi alternatif terhadap PSTN tradisional. Ketika desain integrasi dirancang dengan baik, telepon analog biasa dapat meneruskan sinyal suara ke jaringan digital dan berhasil melakukan panggilan yang dapat terhubung ke telepon IP dan dapat dengan mudah berkomunikasi satu sama lain.

5. Berdasarkan penelitian (Ali *et al.*, 2022) yang berjudul “*Covert VoIP Communication based on Audio Steganography*” dijelaskan bahwa dua codec G.711 dan iLBC merupakan *codec* yang digunakan. Untuk mewujudkan *bandwidth* komunikasi *real-time*, *encoder* dan *decoder* G.711 digunakan untuk mengompresi dan mendekompresi *cover speech* dan data stego. Sedangkan pembicaraan rahasia dikompresi dan didekompresi masing – masing menggunakan *enkoder* dan *dekoder* iLbc.

1.8 Sistematika Penulisan

Sistematika penulisan skripsi yang digunakan dalam penelitian ini adalah sebagai berikut :

BAB 1 PENDAHULUAN

Pembahasan bab ini meliputi latar belakang penelitian, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian, penelitian relevan, dan sistematika penulisan.

BAB 2 LANDASAN TEORI

Bab ini menjelaskan beberapa teori yang berkaitan erat dengan penelitian yang sedang dilakukan, misalnya apa itu *Voice over Internet Protocol*, Briker sebagai *server*, serta alat-alat yang digunakan yang didapat dari berbagai sumber.

BAB 3 ANALISIS DAN PERANCANGAN

Melakukan perancangan sebuah sistem komunikasi *Voice over Internet Protocol*, yang mana Briker dipilih sebagai *server*. Selanjutnya, analisis

data akan dilakukan menggunakan aplikasi Wireshark. Ini akan menunjukkan hasil kualitas layanan VoIP berdasarkan parameter QoS masing-masing.

BAB 4 IMPLEMENTASI DAN PENGUJIAN

Implementasi dan pengujian sistem sesuai dengan tahapan analisis dan perancangan yang sudah dijelaskan akan dibahas dalam bab ini.

BAB 5 KESIMPULAN DAN SARAN

Dalam bab ini menjelaskan secara singkat kesimpulan terkait hasil penting yang diperoleh serta saran yang diberikan penulis sebagai masukan kepada penelitian mendatang.

BAB II

LANDASAN TEORI

2.1 Video Call

Video call atau panggilan video adalah salah satu bentuk komunikasi dimana setiap orang dapat berkomunikasi secara langsung melalui layar video dan audio, sehingga memungkinkan komunikasi *real-time* antar individu di lokasi yang berbeda (Pratiwi, 2017). Video call memudahkan setiap manusia untuk berkomunikasi secara *face-to-face* melalui telepon, membuktikan bahwa inovasi komunikasi saat ini sudah maju dan jarak jauh tidak lagi menjadi kendala.

Berikut ini beberapa protokol yang mendukung layanan panggilan *video call*:

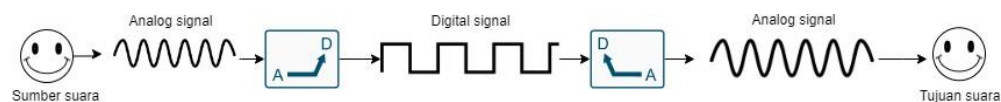
1. *Transmission Control Protocol* (TCP), sebuah protokol yang bertindak menjamin keandalan koneksi komunikasi *end-to-end*.
2. *User Datagram Protocol* (UDP), adalah lapisan transport yang cara kerjanya mirip TCP namun kurang dapat diandalkan (*unreliable*).
3. *Real-Time Transport Protocol* (RTP), sebuah protokol transportasi yang beroperasi secara *real-time*.
4. *Internet Protocol* (IP), adalah protokol penting untuk menangani pengalamatan dan merutekan paket data.

2.2 LAN

Local Area Network (LAN) mengacu pada sebuah jaringan dimana dibangun dalam lingkup terbatas, misalnya gedung atau ruangan tertentu. Jaringan lokal sering juga disebut sebagai jaringan *private*. Jaringan lokal digunakan untuk menghubungkan perangkat-perangkat dalam lingkup kecil yang memungkinkan setiap perangkat untuk berbagi sumber daya (*resource*) satu sama lain, misalnya printer atau penyimpanan data bersama (Putra, 2020).

2.3 Voice over Internet Protocol

Voice over Internet Protocol (VoIP) adalah teknologi yang memungkinkan pengiriman data, video, dan suara dalam format paket melalui jaringan IP. Jaringan IP adalah infrastruktur pengiriman data yang berbasis *Packet-Switch*, yang memungkinkan pengguna untuk melakukan panggilan melalui internet atau jaringan IP (Khomar, 2021). Dengan memanfaatkan jaringan IP dapat menekan pengeluaran biaya karena tidak memerlukan infrastruktur telepon konvensional dan menggunakan *bandwidth* lebih efisien dibandingkan telepon biasa (Berlian, 2020). Dalam jaringan VoIP, pengguna dapat melakukan panggilan menggunakan perangkat seperti PC, *smartphone*, atau telepon berbasis IP, dan terlepas dari layanan *Public Switched Telephone Network* (PSTN) yang mengenakan biaya berdasarkan durasi panggilan.

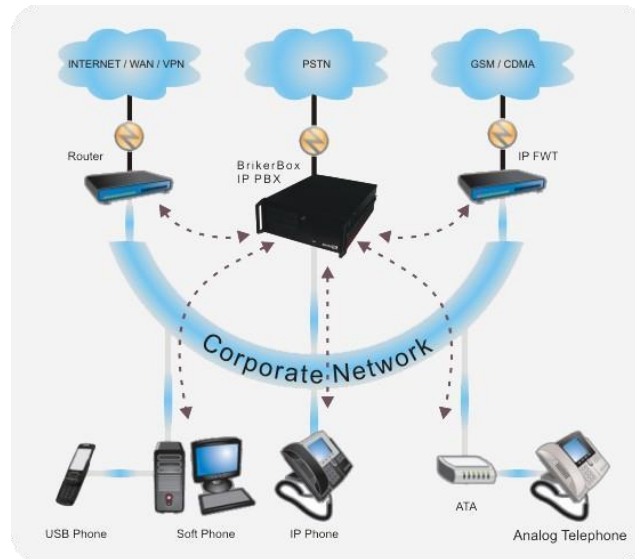


Gambar 2.1 Cara Kerja VoIP

Gambar 2.1 diatas menunjukkan cara kerja VoIP, dengan proses pengiriman dimulai dengan mengubah sinyal menjadi format digital. ADC (*Analog to Digital Converter*) mengubah sinyal analog menjadi sinyal digital sebelum sampai ke tujuan. Setelah sampai, DAC (*Digital to Analog Converter*) mengubah data digital menjadi sinyal analog, sehingga penerima dapat menerima data sesuai dengan sinyal yang dikirimkan.

2.4 Briker

Briker merupakan distribusi Linux yang menyediakan aplikasi server untuk memudahkan penerapan layanan VoIP. Briker memungkinkan pengguna untuk membuat PBX khusus yang disesuaikan untuk perangkat seluler seperti *smartphone* dan laptop dengan menggunakan Session Initiation Protocol (SIP) (Putra, W.A.M, 2019). Direkomendasikan penggunaan satu server Briker per organisasi, lembaga atau bisnis, dengan batasan maksimal 1.000 ekstensi dan maksimal 240 percakapan simultan.



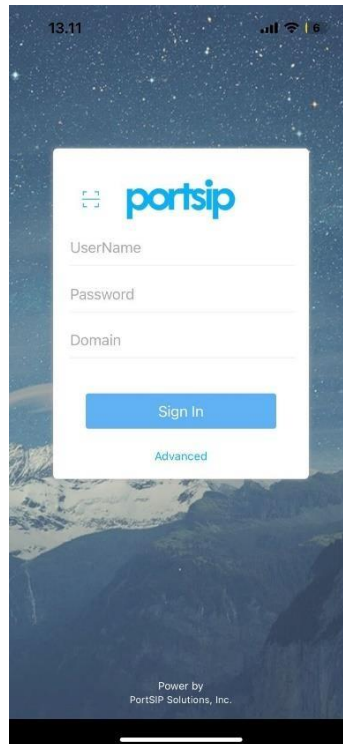
Gambar 2.2 Topology Briker (Briker Documentation)

Gambar 2.2 di atas menampilkan posisi Briker dalam jaringan TCP/IP. Briker bisa dipasang dengan perangkat keras khusus untuk terkoneksi dengan jaringan telekomunikasi yang ada, baik itu analog atau digital.

Sistem topologi ini dapat diterapkan di tempat lain, dimana Briker harus terhubung satu sama lain. Hal ini biasa disebut dengan *trunking*. Dalam prakteknya, ketersediaan *bandwidth* akan mendukung banyaknya trunk dan kekuatan pemrosesan perangkat yang akan menjalankan Briker. Briker tidak hanya terhubung ke Briker lain, tetapi juga dapat terhubung ke server VoIP. Disisi lain, karena Briker mendukung banyak protokol, Briker juga dapat diintegrasikan ke dalam jaringan yang menggunakan standar H.323.

2.5 PortSIP UC

Dalam penelitian ini, PortSIP UC adalah aplikasi softphone yang dipilih dimana aplikasi ini berbasis protokol SIP yang berfungsi memfasilitasi pengguna melakukan panggilan suara atau video. Aplikasi PortSIP bisa diunduh pada sistem operasi iOS yaitu *App Store* dan Android yaitu *Play Store*, serta dapat dinikmati secara gratis oleh pengguna untuk panggilan suara (Tricahya, 2023). Berikut adalah tampilan antarmuka dari aplikasi PortSIP UC ditunjukkan pada Gambar 2.3.



Gambar 2.3 Tampilan PortSIP

Keterangan :

- a. *Username*, merupakan identitas pengguna yang sudah terdaftar pada *server Briker*.
- b. *Password*, merupakan kode sandi untuk pengguna.
- c. *Domain*, alamat IP Briker yang sedang aktif, atau bisa dibilang adalah alamat hub dari sebuah *server Briker*.
- d. *Sign In*, proses masuk atau log masuk ke akun pengguna yang sudah terdaftar.
- e. *Advanced*, merupakan opsi pengaturan tambahan *login* pada aplikasi PortSIP.

2.6 Wireshark

Salah satu alat *Network Analyzer* yang banyak dipakai oleh *Network Administrator* untuk mengontrol dan menilai performa jaringan mereka adalah *Wireshark*. Keunggulan *wireshark* terletak pada *interface* yang memiliki *Graphical User Interface* (GUI) atau tampilan antarmuka pengguna grafis sehingga mudah dipakai (Putra, 2019).

Wireshark merupakan perangkat lunak yang dirancang sebagai penganalisa paket data di dalam sebuah jaringan. Dengan kemampuannya untuk melacak paket secara langsung, *wireshark* dapat menangkap dan menampilkan data dengan jelas. Dengan memanfaatkan Pcap untuk menangkap paket jaringan, *wireshark* bekerja dengan menangkap paket data dari berbagai protokol yang beroperasi pada berbagai jenis jaringan yang umum ditemukan dalam lalu lintas jaringan. Di jendela hasil *capture*, paket data yang ditangkap akan ditampilkan secara *real-time*. Melalui lisensi *General Public License* (GNU), *wireshark* dapat digunakan, dibagikan, dan dimodifikasi secara bebas karena bersifat *open source* (Rendra, 2022).

2.7 Access Point

Access point memiliki dua komponen, yaitu *transceiver* dan antena, yang mana kedua komponen tersebut berfungsi sebagai pusat untuk menerima dan memancarkan sinyal antara *client server*. Berbeda dengan *router* yang berfungsi mengatur aliran data, fungsi *access point* ialah menerima atau menolak perangkat yang akan terhubung ke jaringan local yang sama, bergantung pada kecocokan kata sandi yang dimasukkan oleh pengguna (Putra, 2020). *Access point* juga dikenal sebagai *Wireless Local Area Network* (WLAN), berperan dalam pengiriman dan penerimaan data dari adapter nirkabel. Secara konseptual, *access point* dapat dianggap sebagai penghubung antara jaringan kabel dengan jaringan nirkabel (Rendra, 2022). Untuk tampilan dari *access point* dapat dilihat pada Gambar 2.4 dibawah ini :



Gambar 2.4 Access Point

(Sumber : <https://connect-indo.co.id/product/53/188/Page-TP-Link-AP-Indoor-TL-WA901ND-450Mbps-Wireless-N-Access-Point>)

2.8 Switch

Menurut definisi, *switch* adalah sebuah komponen fisik dalam sistem komputer yang berfungsi sebagai penghubung dalam jaringan komputer. Peran *switch* adalah menghubungkan mengatur proses pertukaran paket data. Sebuah *switch* memiliki banyak *port*. Ketika ada *frame* data yang datang di *port* mana pun, *switch* akan memeriksa alamat tujuan kemudian mengirimkannya ke perangkat atau tujuan yang dituju (Kusuma, 2020). Berikut tampilan *switch* ditunjukkan pada Gambar 2.5 dibawah ini :

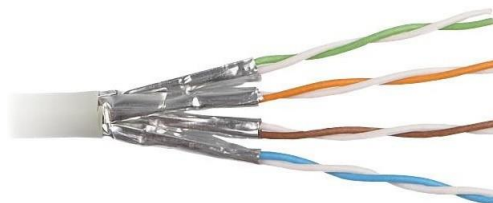


Gambar 2.5 Switch

(Sumber : <https://tekno.kompas.com/read/2023/03/12/14150077/4-perbedaan-hub-dan-switch-dalam-jaringan-komputer-yang-perlu-diketahui>)

2.9 Kabel Unshield Twisted Pair (UTP)

Kabel UTP, yang terdiri dari empat pasang kabel, digunakan untuk membangun jaringan komputer. Terdapat dua jenis kabel twisted pair yaitu *Shielded* dan *Unshielded*. Kabel UTP *shielded* memiliki lapisan perlindungan alumunium, sedangkan *unshielded* adalah kabel tanpa perlindungan. Kedua jenis kabel ini dapat dihubungkan dengan menggunakan konektor RJ-45 atau RJ-11 (Rendra, 2022). Pada Gambar 2.6 dibawah ini memperlihatkan empat pasang kabel *Unshield Twisted Pair*, sebagai berikut:



Gambar 2.6 Kabel UTP

(Sumber : <https://jarkomtutorial.wordpress.com/>)

2.10 QoS (Quality of Service)

Menurut panduan CCITT E.800, *Quality of Service* (QoS) adalah penilaian kinerja umum suatu layanan, yang menentukan tingkat kepuasan pengguna terhadap layanan tersebut. QoS mengacu pada kemampuan jaringan untuk menyediakan layanan yang lebih baik pada trafik jaringan tertentu. *Delay*, *jitter*, dan *throughput*, dan adalah beberapa parameter QoS.

2.10.1 Parameter QoS (Quality of Service)

Berikut ini merupakan parameter dari *Quality of Services*, diantaranya :

1. **Throughput**, yang diukur dalam satuan bit per detik (bps), adalah ukuran kecepatan pengiriman data jaringan. *Throughput* adalah jumlah total data yang sukses dikirim selama waktu tertentu dibagi dengan waktu tersebut.

$$\text{Throughput} = \frac{\text{Jumlah bit yang dikirim}}{\text{Waktu pengiriman paket}} \dots\dots\dots(2.1)$$

Throughput ditentukan oleh beberapa faktor :

- a. Perangkat jaringan yang digunakan
 - b. Jenis data yang ditransfer
 - c. Topologi jaringan yang digunakan
 - d. Jumlah pengguna di jaringan
 - e. Spesifikasi PC *client*/pengguna
 - f. Spesifikasi *server* komputer
 - g. Gangguan induksi listrik dan kondisi cuaca
2. **Delay (Latency)**, adalah waktu yang diperlukan untuk mengirimkan data dari sumber (pengirim) ke tujuan (penerima). *Delay* diukur sebagai waktu antara pengiriman satu paket data dengan paket data berikutnya, biasanya dalam hitungan detik.

Berikut adalah rumus untuk menghitung *delay* :

$$\text{delay} = \frac{\text{duration}}{\text{total packet}} \dots\dots\dots(2.2)$$

Keterangan :

Duration = jumlah waktu keseluruhan yang dibutuhkan untuk mengirim paket

Total *packet* = jumlah total paket yang dikirim

Tabel 2.1 Standar *Delay* berdasarkan ITU-T

Besar <i>Delay</i>	Kategori
<150 ms	Sangat Bagus
150 ms s/d 300 ms	Bagus
300 ms s/d 450 ms	Jelek
>450 ms	Sangat Jelek

3. **Jitter**, mengacu pada variasi waktu yang terjadi antara kedatangan paket data, yang merupakan perubahan seketika tanpa akumulasi dari posisi ideal sinyal digital. Agar mendapatkan kualitas layanan (QoS) yang optimal, upaya untuk mengurangi nilai jitter harus dilakukan secara maksimal.

Tabel 2.2 Standar *Jitter* berdasarkan ITU-T

Besar <i>Jitter</i>	Kategori
0 ms	Sangat Bagus
0 – 75 ms	Bagus
76 ms s/d 125 ms	Sedang
125 ms s/d 225 ms	Buruk

Kinerja *jitter* bergantung pada besarnya nilai yang dihasilkan. Ketika nilai *jitter* semakin besar, maka semakin buruk kinerja jaringan. Maka, jitter harus dikurangi seefisien mungkin. Berikut rumus yang dipakai dalam perhitungan *jitter* :

$$\text{mean jitter} = \frac{\text{total jitter}}{\text{total pengiriman paket}} \dots\dots\dots(2.3)$$

BAB III

ANALISIS DAN PERANCANGAN

3.1 Analisis Sistem

Analisis sistem adalah teknik pemecahan masalah dengan menguraikan masalah didalam suatu sistem menjadi komponen-komponen yang lebih kecil sehingga mudah dipahami. Tahapan ini terbagi menjadi dua proses, yaitu analisis masalah dan analisis kebutuhan. Analisis masalah berfokus untuk mengidentifikasi dan memahami masalah dan mendapatkan solusi terbaik dari permasalahan tersebut, sedangkan analisis kebutuhan adalah menggambarkan secara rinci terkait informasi dan langkah-langkah yang diperlukan dalam perancangan sistem.

3.1.1 Analisis masalah

Dalam analisis masalah akan dilakukan tahapan mengidentifikasi sebab - akibat terhadap sebuah permasalahan. Pada penelitian ini pula akan dilakukan analisis bagaimana server Briker yang akan di konfigurasikan ke dalam teknologi *Voice over Internet Protocol* mampu untuk menyelesaikan permasalahan komunikasi antar karyawan dalam sebuah gedung perkantoran.

Pada tahap mengidentifikasi masalah, penelitian ini menggunakan metode 5-*Whys* untuk mempermudah proses analisis. Metode 5-*Whys* merupakan metode tanya-jawab sederhana yang cukup efektif ketika fokus utamanya adalah mengidentifikasi sebab dan akibat dari suatu masalah. Metode ini dilakukan dengan bertanya “mengapa” secara berulang sebanyak 5 kali atau lebih, diantaranya sebagai berikut:

1. Mengapa diperlukan sistem komunikasi *video call*?

Seperti yang diketahui, dalam sebuah gedung perkantoran, jarak antar ruangan seringkali menjadi kendala yang mengakibatkan kurang optimalnya komunikasi antar karyawan. Padahal, komunikasi merupakan salah satu kebutuhan mendasar dalam lingkungan perusahaan perkantoran. Oleh karena itu, diperlukan pembangunan sistem komunikasi video call dengan tujuan

membantu karyawan ketika melaksanakan komunikasi yang tidak selalu dapat dilakukan secara langsung. Hal ini diharapkan dapat meningkatkan kinerja dan efisiensi komunikasi antara sesama karyawan.

2. Mengapa penggunaan komunikasi melalui internet dianggap lebih baik daripada menggunakan kabel telepon tradisional?

Memanfaatkan internet untuk berkomunikasi, seperti VoIP, memiliki sejumlah keunggulan dibandingkan dengan telepon tradisional (PSTN), karena lebih ekonomis dengan tidak perlu infrastruktur tambahan dan menggunakan lebar data (bandwidth) yang lebih kecil. Terlebih lagi, komunikasi melalui VoIP dapat diakses oleh berbagai perangkat, baik itu komputer maupun smartphone, berbeda dengan telepon tradisional yang hanya dapat digunakan pada perangkat telepon khusus.

3. Mengapa VoIP (*Voice over Internet Protocol*) diperlukan dalam sistem ini?

VoIP sangat penting untuk memberikan solusi dalam melakukan komunikasi antar karyawan tanpa harus menggunakan telepon PSTN. Menggunakan telepon PSTN dapat menyebabkan biaya bertambah seiring dengan semakin seringnya digunakan. Manfaat penggunaan VoIP terletak pada kemampuannya untuk menyampaikan suara melalui paket data di jaringan IP dengan biaya yang lebih terjangkau bagi pengguna.

4. Mengapa penggunaan teknologi VoIP dianggap sebagai alternatif yang lebih hemat biaya dibandingkan dengan telepon PSTN dalam lingkup jaringan lokal?

Dalam kasus VoIP, untuk melakukan panggilan melalui infrastruktur jaringan data tentu mengurangi biaya yang biasanya dikeluarkan suatu lembaga ketika menggunakan sistem telepon tradisional seperti *Public Switched Telephone Network* (PSTN), *Global System for Mobile Communications* (GSM), *Code-Division Multiple Access* (CDMA). Oleh karena itu, konteks VoIP di sini hanya digunakan di area lokal dan tidak memerlukan layanan *Internet Service Provider* (ISP) pada umumnya.

5. Mengapa penting untuk menkonfigurasi Briker agar *Voice over Internet Protocol* berjalan?

Distribusi Linux yang memungkinkan pengimplementasian layanan VoIP didalamnya adalah Briker. VoIP adalah teknologi yang mengubah sinyal analog ke digital, sementara Briker memiliki perangkat keras khusus untuk koneksi analog dan digital. Ini mengindikasikan bahwa VoIP mampu berfungsi di server Briker karena keduanya saling berhubungan.

3.1.2 Analisis kebutuhan

Analisis kebutuhan merupakan proses untuk mengidentifikasi dan memahami kebutuhan yang harus dipenuhi dalam perancangan untuk memenuhi tujuan. Analisis kebutuhan melibatkan kebutuhan fungsional dan kebutuhan non-fungsional yang telah ditetapkan secara optimal.

3.1.2.1 Kebutuhan fungsional

Kebutuhan fungsional melibatkan segala proses yang dikerjakan oleh sistem guna memperoleh destinasi yang ditetapkan. Dalam hal ini, penelitian memiliki kebutuhan fungsional utama, yakni :

1. Server Briker dapat merespon *request video call* dari *client*
2. Server Briker dapat membaca nomor yang teregistrasi pada ip-phone *client*
3. Server Briker dapat menerima *request* panggilan dan menolak *request* panggilan.

3.1.2.2 Kebutuhan Non-Fungsional

Kebutuhan tambahan untuk mendukung sebuah sistem disebut kebutuhan non-fungsional, termasuk kinerja, keamanan, dan batasan sistem. Penelitian ini memiliki beberapa kebutuhan non-fungsional, yaitu:

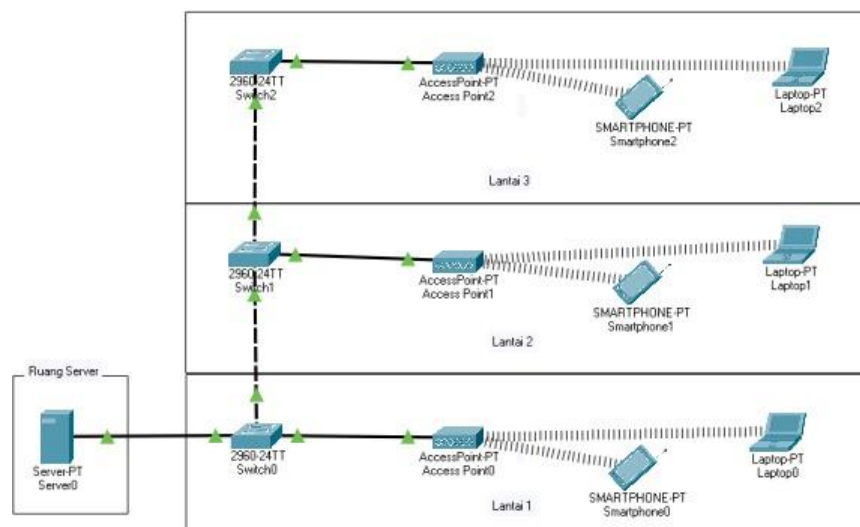
1. Perancangan dan implementasi hanya berada didalam jaringan LAN
2. *Software* yang digunakan yaitu *PortSIP UC Client*
3. *Hardware* yang dibutuhkan berupa *access point*, *switch*, laptop, dan *smartphone*
4. Melakukan perhitungan parameter *Throughput*, *Delay*, dan *Jitter*.

3.2 Perancangan Sistem

Dalam merancang sistem, beberapa diagram yang dibutuhkan termasuk diagram umum, *use case diagram*, *activity diagram*, *sequence diagram*, diagram alir (flowchart), dan *user interface*.

3.2.1 Diagram Umum

Diagram umum merupakan representasi visual tentang bagaimana sistem dapat berjalan dengan menunjukkan bagaimana proses, aliran, dan interaksi antara komponen-komponen dalam sistem berlangsung. Seluruh desain aplikasi ini akan dijelaskan secara rinci pada Gambar 3.1 dibawah ini :

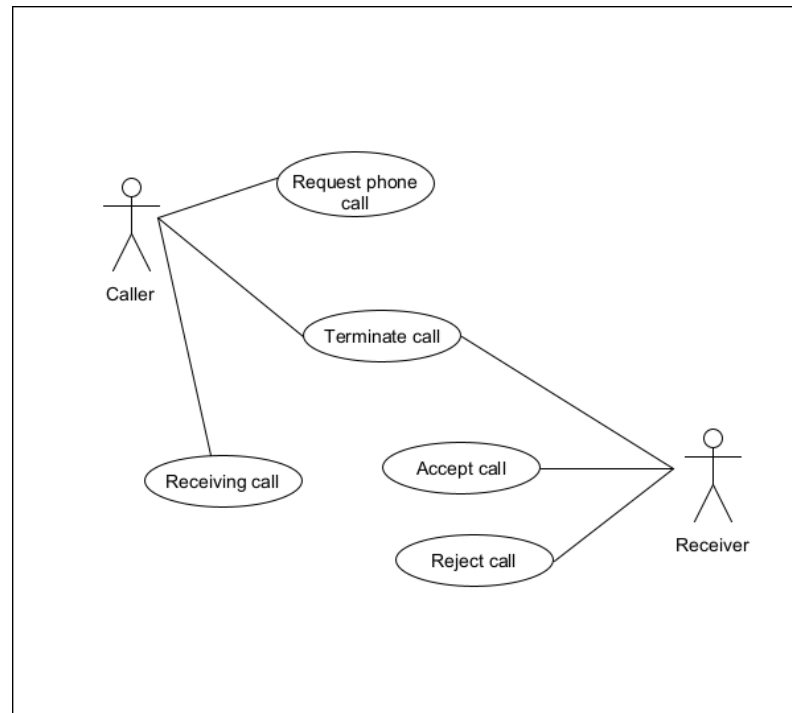


Gambar 3.1 Diagram Umum Sistem

Untuk Gambar 3.1 memperlihatkan bagaimana topologi VoIP *video call* diatas bekerja. Untuk laptop ataupun *smartphone* yang sudah terinstall PortSIP UC akan menjadi *client* VoIP dan terhubung satu sama lain didalam jaringan lokal, membentuk koneksi VoIP yang dihubungkan oleh *access point*, yang kemudian *access point* akan menghantarkan sinyal panggilan ke *switch* dan bisa terhubung dengan *server* Briker VoIP.

3.2.2 Use Case Diagram

Diagram use case adalah bagian dari *Unified Modeling Language* (UML) yang menunjukkan hubungan antara sistem dan aktor. Pada Gambar 3.2 dibawah ini adalah representasi *diagram use case* yang diterapkan pada sistem ini.

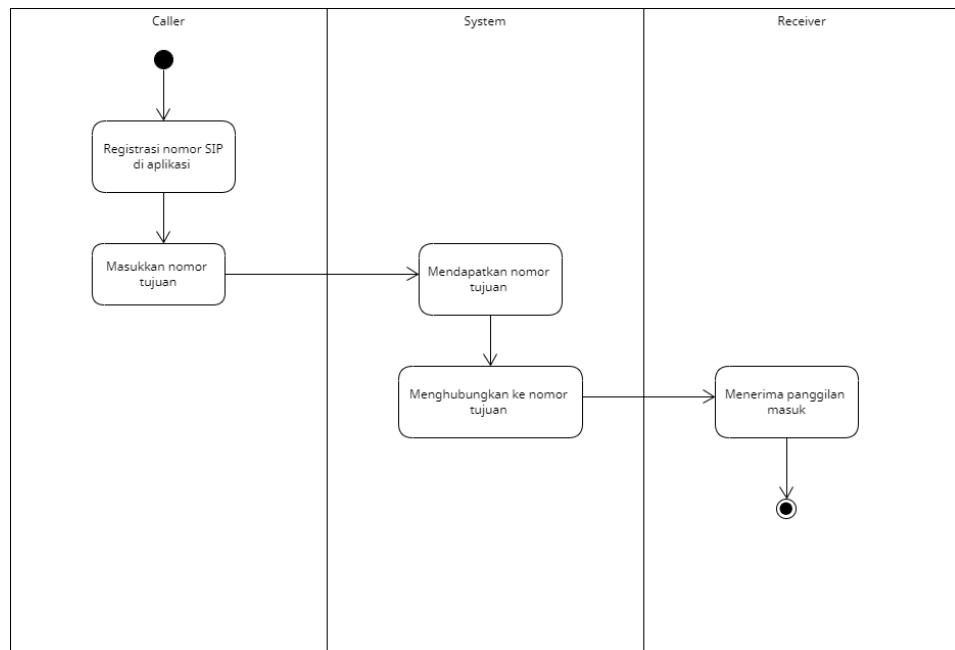


Gambar 3.2 *Use Case Diagram*

Gambar 3.2 menunjukkan bahwa kerjasama antara aktor yaitu *Caller* dan *Receiver* saling berhubungan untuk melakukan proses komunikasi menggunakan *Voice over Internet Protocol*.

3.2.3 Activity Diagram

Activity Diagram digunakan untuk menggambarkan proses yang terjadi dalam sistem secara visual. Dalam *diagram activity*, setiap komponen dihubungkan oleh sebuah tanda panah. Panah ini berfungsi mengurutkan tindakan yang dilakukan dari awal hingga akhir. *Diagram activity* yang digunakan dalam sistem ini disajikan pada Gambar 3.3, yakni:

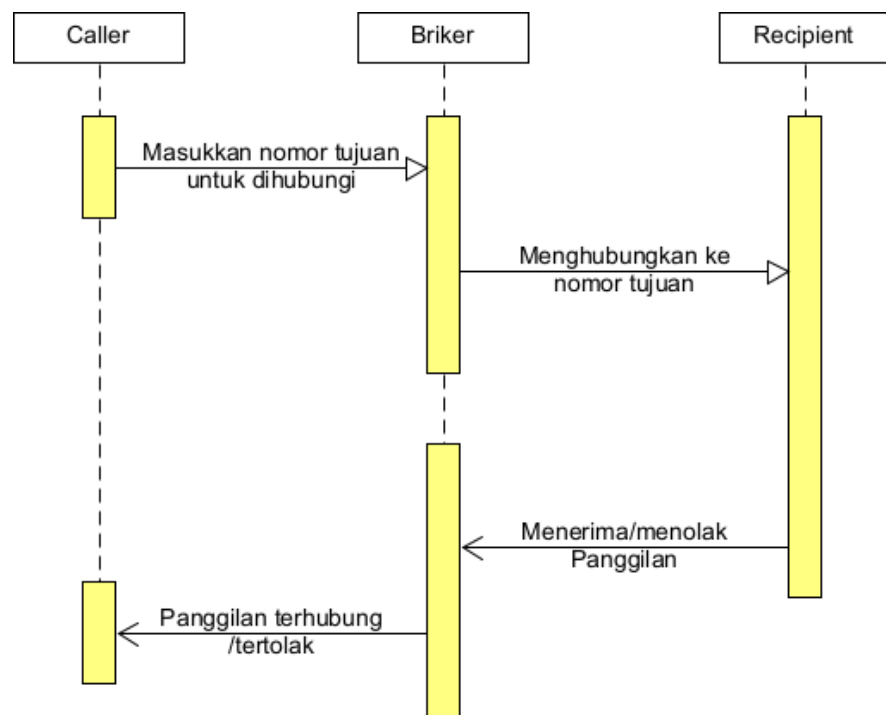


Gambar 3.3 *Activity Diagram*

Gambar 3.3 menjelaskan langkah awal dimana *Caller* akan terlebih dahulu melakukan registrasi nomor SIP di aplikasi yang digunakan yaitu PortSIP UC. Setelah melakukan registrasi maka *Caller* bisa memasukkan nomor tujuan yang akan dihubungi. Sistem akan mendapatkan nomor tujuan lalu menghubungkan panggilan sesuai nomor tujuan. *Receiver* kemudian menerima panggilan masuk. Lalu setelah melewati kondisi yang ada panggilan *video call* menggunakan server Briker VoIP akan berlangsung.

3.2.4 *Sequence Diagram*

Sebuah representasi grafis dalam format UML (*Unified Modeling Language*) yang dikenal dengan *sequence diagram* berfungsi untuk menunjukkan ikatan antara elemen-elemen sistem dalam periode waktu tertentu. Diagram ini menggambarkan urutan langkah-langkah dan komunikasi yang terjadi di antara setiap komponen dalam sistem, termasuk pengguna dan objek-objek yang terlibat. Berikut Gambar 3.4 menunjukkan *sequence diagram* yang diterapkan dalam sistem:



Gambar 3.4 *Sequence Diagram*

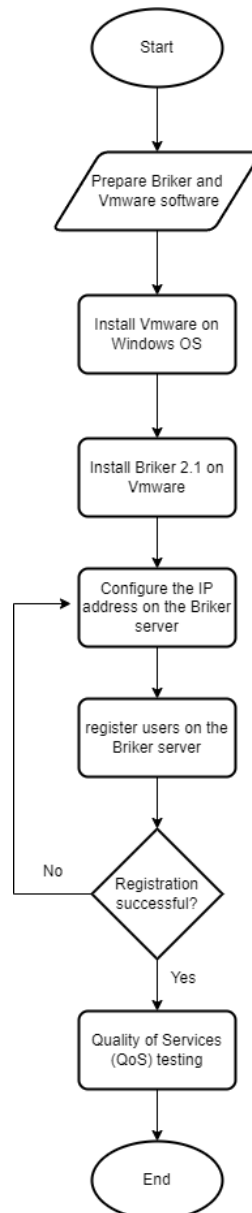
Pada Gambar 3.4 diatas, menunjukkan terdapat 3 komponen penyusun yang saling berkomunikasi satu dengan yang lain. Dimulai dari memasukkan nomor tujuan, menerima/menolak panggilan hingga panggilan terhubung.

3.3 *Flowchart* (Diagram Alir)

Representasi visual yang digunakan untuk menggambarkan urutan dari serangkaian proses, kondisi, dan alur logika dari suatu sistem dikenal dengan istilah *flowchart*. Setiap tahap dalam *flowchart* direpresentasikan dengan simbol-simbol khusus yang memiliki makna yang berbeda untuk menjelaskan alur atau langkah dalam sistem tersebut. Berikut adalah rancangan *flowchart* dalam sistem:

3.3.1 *Flowchart* Perancangan Sistem IPPBX

Pembangunan sistem IPPBX melibatkan pemanfaatan server, dimana Briker dipilih sebagai server dalam penelitian ini. Setelah proses instalasi server IPPBX selesai, *user* mencoba melakukan panggilan dari satu *user* ke *user* lainnya. Setiap *user* harus terdaftar pada server Briker di setiap perangkat. Berikut rincian perancangan sistem IPPBX ditunjukkan pada Gambar 3.5, sebagai berikut :

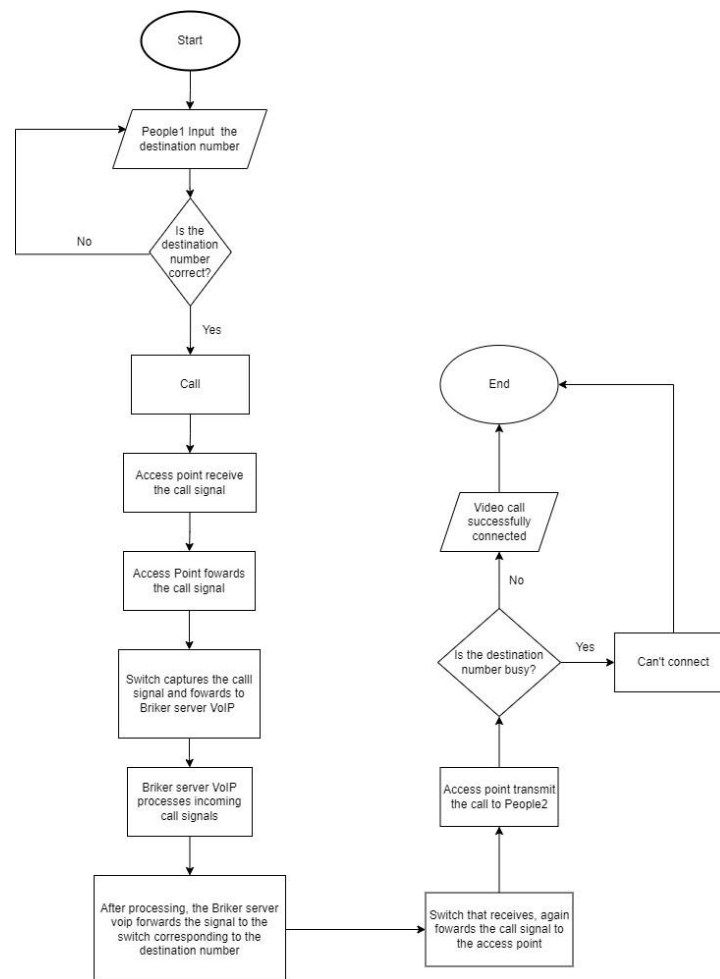


Gambar 3.5 Flowchart Perancangan Sistem IPPBX

Gambar 3.5 memperlihatkan bagaimana langkah-langkah dalam mengimplementasikan sistem jaringan IPPBX. Tahapan pertama sebelum menginstal VMware di sistem operasi (OS) *Windows* adalah menyiapkan perangkat lunak VMware dan CD Briker 2.1.1. Setelah instalasi selesai, *IP address server* dikonfigurasi dengan *dynamic IP address* dan ekstensi nomor dibuat untuk setiap *user*. Setelah itu, dilakukan pendaftaran *user*. Pendaftaran dianggap berhasil jika nomor ekstensi terdaftar di server. Setelah berhasil mendaftarkan nomor ekstensi, langkah selanjutnya adalah pengujian panggilan dan perhitungan parameter Quality of Service (QoS).

3.3.2 Flowchart of VoIP Network System Using Briker Server

Berikut *flowchart* dari server Briker VoIP :



Gambar 3.6 *Flowchart of VoIP*

Penjelasan Konsep Sistem jaringan VoIP menggunakan server Briker :

Berdasarkan Gambar 3.6. Orang-1 akan melakukan sebuah panggilan yang akan diterima oleh *access point*. Setelah *access point* menerima sinyal panggilan maka akan diteruskan ke *switch*, *switch* meneruskan sinyal panggilan ke *server* Briker VoIP yang kemudian akan dikirimkan lagi kepada *switch* yang dituju. *Switch* kembali meneruskan sinyal panggilan ke *access point* dan terakhir sinyal panggilan akan diterima oleh Orang2, sehingga panggilan *video call* berhasil terhubung.

Penjelasan flowchart berdasarkan Gambar 3.5 :

1. *Start* : berfungsi untuk memulai skema panggilan.

2. *People1 input the destination number* : *People1* memasukkan sebuah nomor tujuan.
3. *Is the destination number correct?* : Kondisi dimana melakukan pengecekan terhadap nomor tujuan

Kondisi *YES* :

- a. *Call* : Terjadi ketika nomor tujuan yang dimasukkan benar

Kondisi *NO* : kembali ke proses dimana *People1* kembali memasukkan sebuah nomor tujuan.

4. *Access point receive the call signal* : *Access point* menerima sinyal panggilan
5. *Access point forwards the call signal* : *Access point* meneruskan sinyal panggilan
6. *Switch captures the call signal and forwards to Briker server VoIP* : *Switch* menangkap sinyal panggilan kemudian meneruskannya ke *server Briker VoIP*
7. *Briker server VoIP processes incoming call signals* : *Server Briker VoIP* memproses sinyal panggilan yang masuk
8. *After processing, the Briker server voip forwards the signal to the switch corresponding to the destination number* : Setelah diproses, maka *server briker voip* meneruskan sinyal tersebut ke *switch* yang sesuai dengan nomor tujuan
9. *Switch that receives, again forwards the call signal to the access point* : *Switch* yang menerima, kembali meneruskan sinyal panggilan kepada *access point*
10. *Access point transmit the call to People2* : *Access point* menghantarkan sinyal panggilan kepada *People2*
11. *Is the destination number busy?* : Kondisi dimana melakukan pengecekan apakah nomor tujuan sedang sibuk

Kondisi *YES* :

- a. *Can't connect* : Jika ternyata nomor tujuan sedang sibuk maka panggilan *video call* tidak dapat terhubung

Kondisi *NO* :

- a. *Video call successfully connected* : Merupakan *output* dimana nomor tujuan tidak sibuk dan panggilan *video call* berhasil terhubung

12. *End* : Panggilan selesai.

BAB IV

IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi

Proses implementasi dan pengujian dimulai setelah analisis dan desain selesai. Tiga komponen utama yang saling terhubung diantaranya, *VoIP Server*, *Access Point*, dan *VoIP Client*, membentuk struktur komunikasi video call dalam penelitian ini. *Server VoIP* memiliki tanggung jawab dalam memproses, mendaftarkan, dan mengelola panggilan client VoIP. *VoIP client*, apakah itu laptop atau *smartphone* dengan aplikasi *softphone* terinstal, mendaftar ke server dan mengadakan panggilan ke *client* VoIP lain yang sudah terdaftar di server briker.

4.1.1 Implementasi Panggilan dari IP ke IP

Langkah-langkah untuk melakukan panggilan dari IP ke IP menggunakan sistem VoIP, yaitu :

1. Sistem operasi Briker 2.1.1 diinstal pada komputer *server*, yang meliputi persiapan konfigurasi yang diperlukan untuk implementasi sistem VoIP.
2. Perangkat lunak VoIP (*Softphone* PortSIP UC) diinstal pada *VoIP client*.
3. Ekstensi SIP yang digunakan oleh *client* diatur pada komputer *server* yang dikonfigurasi pada Briker 2.1.1.
4. Konfigurasi ekstensi SIP *softphone* PortSIP UC dilakukan pada *VoIP client* yang telah terdaftar di *server* VoIP Briker.

4.1.2 Perancangan *Server*

Server VoIP menggunakan sistem operasi Linux. Dalam implementasi ini, Linux yang digunakan adalah distribusi khusus Briker yang dirancang khusus untuk menangani layanan VoIP.

Berikut adalah penggunaan alat yang digunakan dan proses perancangan server yang terdiri atas beberapa tahap :

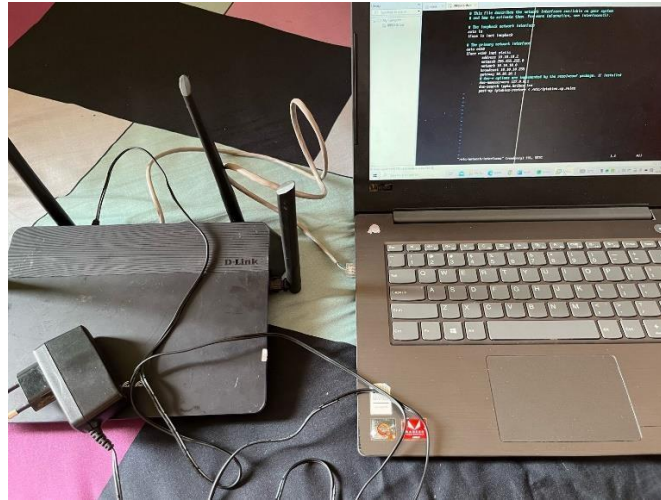
Tabel 4.1 Tabel Alat yang Digunakan

No	Nama Alat
1.	<i>Access Point</i>
2.	<i>Switch</i>
3.	AC Adapter
4.	1 buah PC
5.	Kabel UTP Cat 5
6.	<i>Connector Rj45</i>

**Gambar 4.1** Proses Perancangan *Server*

Terlihat pada Gambar 4.1, langkah pertama dalam perancangan *server* adalah memilih perangkat keras yang akan digunakan sebagai *server*. Briker 2.1.1 kemudian diinstall pada komputer/PC yang khusus digunakan sebagai *server* VoIP, setelah itu dilanjutkan proses *setting server* Briker mulai dari *setting* IP Address hingga *setting* IPPBX.

4.1.3 Bentuk Desain Sistem



Gambar 4.2 Tampilan Rangkaian Perancangan Sistem

Pada gambar 4.2 diatas, dapat terlihat bahwa terdapat susunan dan sambungan dari beberapa perangkat yaitu *access point*, kabel LAN, dll. Susunan tersebut merupakan rangkaian untuk menyelesaikan penelitian **“Penerapan Sistem Komunikasi Video Call Dengan Voice over Internet Protocol Menggunakan Briker Pada Ruang Lingkup Gedung Perkantoran”**.

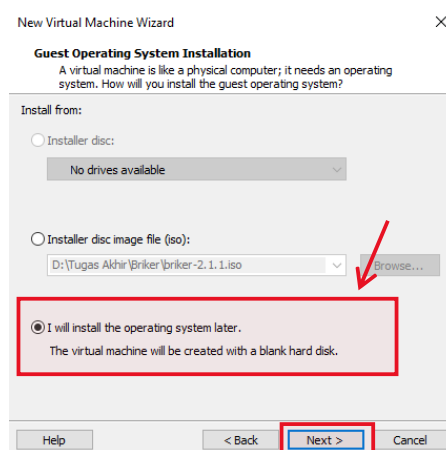
4.2 Pengujian

Untuk tahap pengujian ini akan dilakukan untuk memastikan apakah sistem yang dibangun sudah bisa berfungsi dengan baik dalam proses komunikasi *video call* di gedung perkantoran.

4.2.1 Konfigurasi Briker IPPBX Dalam Pembuatan Server VoIP

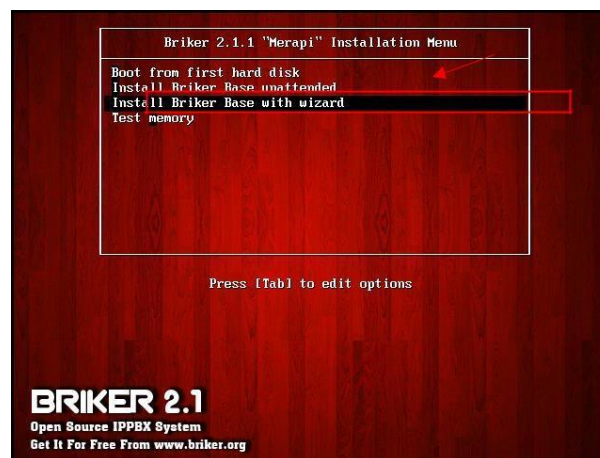
Berikut adalah langkah-langkah dalam konfigurasi Briker :

1. Masuk ke VMware dan ikuti penginstalannya. Untuk Gambar 4.3 merupakan tahapan awal perintah yang akan diberikan oleh Vmware.



Gambar 4.3 Tahap awal

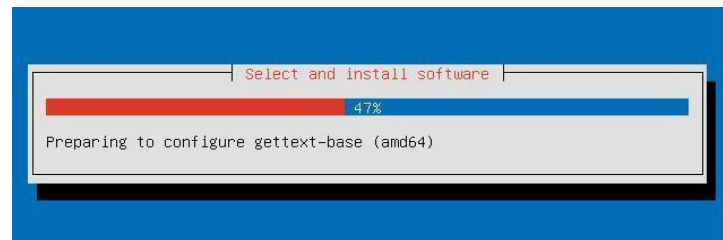
2. Penginstalan Briker ada beberapa pilihan mode install, yaitu :
 - a. *Install Briker Base Unattended*, dimana akan menggunakan seluruh kapasitas *harddisk* dan secara otomatis akan menghapus data yang ada.
 - b. *Install Briker Base with wizard*, mode instalasi ini untuk proses instalasi kostumisasi dengan tampilan grafis yang memudahkan proses instalasi.



Gambar 4.4 Pemilihan Mode Install

Untuk Gambar 4.4 menunjukkan pemilihan mode install Briker *Base with wizard* yang akan digunakan pada penelitian ini. Pemilihan mode install ini agar mempermudah dalam proses instalasi Briker.

3. Tunggu sampai proses instalasi selesai yang ditunjukkan seperti pada Gambar 4.5 dibawah ini.



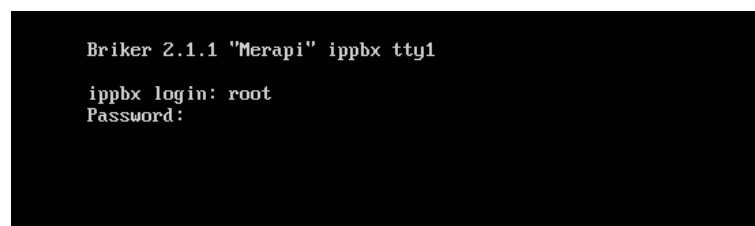
Gambar 4.5 Proses Instalasi

4. Setelah proses selesai, akan muncul seperti Gambar 4.6 dimana *virtual machine* akan *restart* kemudian muncul layar *login* yang menandakan bahwa briker telah berhasil diinstal.



Gambar 4.6 Briker berhasil di Install

5. Untuk user *login* Briker menyediakan user *root* dan *support* dengan *password* yang sama yaitu “Briker”, yang ditunjukkan pada Gambar 4.7 sebagai berikut:



Gambar 4.7 Layar Tampilan *Login*

6. Briker IPPBX memiliki IP Address *default* yaitu 192.168.2.2/24. Seperti pada Gambar 4.8 menunjukkan proses untuk mengganti IP Address nya dilakukan dengan memasukkan perintah `vim /etc/network/interfaces`. Kemudian IP Address yang penulis buat menjadi 10.10.10.2/24.

```

# and how to activate them. For more information, see interfaces(5).

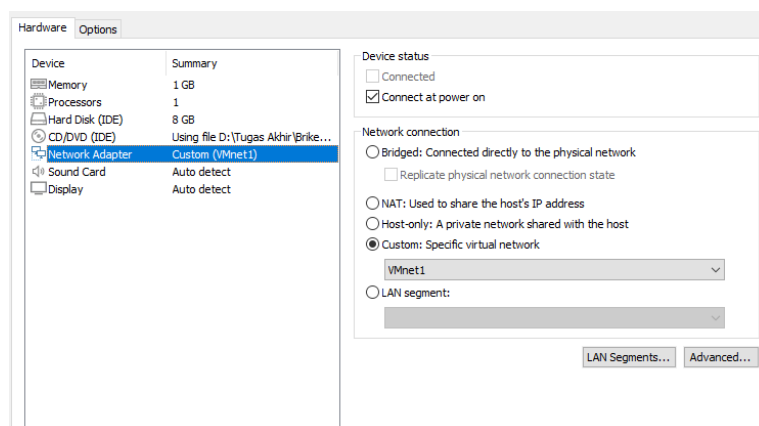
# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
    address 10.10.10.2
    netmask 255.255.255.0
    network 10.10.10.0
    broadcast 10.10.10.255
    gateway 10.10.10.1
    # dns-* options are implemented by the resolvconf package, if installed
    dns-nameservers 127.0.0.1
    dns-search ippbx.briker.lan
    post-up iptables-restore < /etc/iptables.up.rules
--
-- INSERT --
1,1

```

Gambar 4.8 Tampilan untuk mengganti IP *address* Briker

7. Untuk bisa mengakses Briker melalui *browser* kita perlu menyetting adapter VMnet nya menjadi VMnet1, yang dapat dilihat pada Gambar 4.9.



Gambar 4.9 Custom VMnet

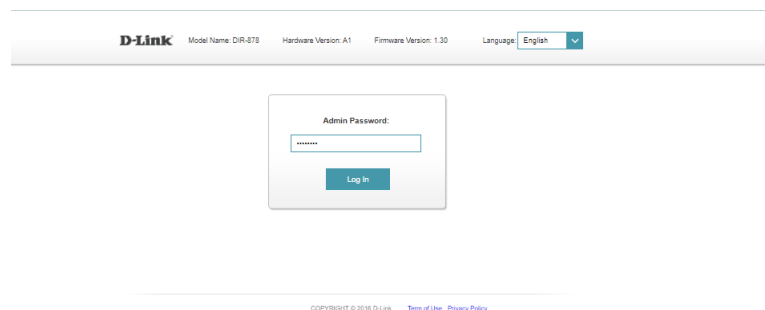
8. Pada Gambar 4.10 menunjukkan cara untuk mengakses Briker melalui *browser* dengan IP *server*.



Gambar 4.10 Akses Briker melalui Browser

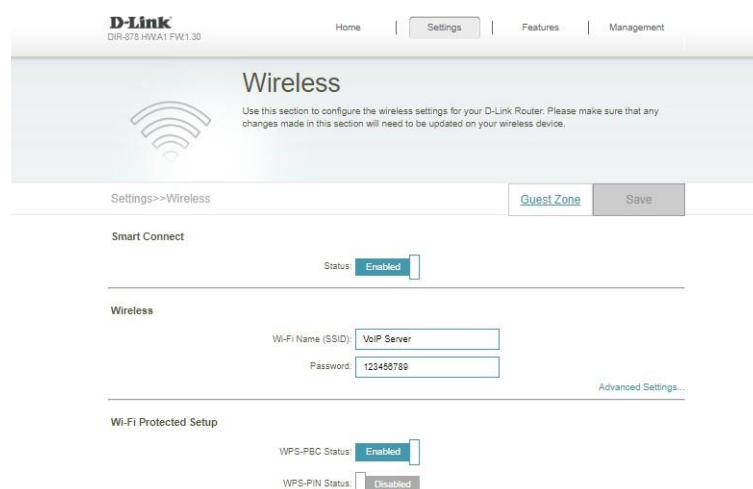
4.2.2 Konfigurasi Access Point

1. Masukkan alamat IP Address yaitu 10.10.10.3, lalu masukkan kata sandi dan klik tombol login untuk mengkonfigurasi *access point* seperti yang diperlihatkan pada Gambar 4.11 dibawah ini.



Gambar 4.11 Login Access Point

2. Untuk menu *Wireless – Wireless AP Setting*, atur *smart connect* menjadi *enabled* lalu isi *Wi-Fi Name* beserta *password* sesuai dengan yang diinginkan. Pada Gambar 4.12 merupakan tampilan menu yang telah diisi.

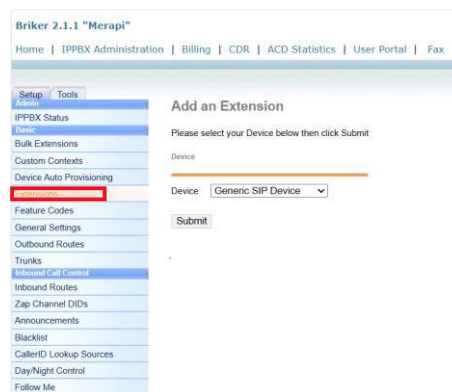


Gambar 4.12 Setting Wireless Access Point

4.2.3 Konfigurasi *Extension*

Pada fitur ini erat kaitannya dengan pengelolaan akun pada IPPBX. Fitur ini memungkinkan *user* untuk menambahkan, menghapus, dan mengubah data akun. Setiap akun yang ditambahkan akan berfungsi sebagai *extension* IPPBX. *Extension* dapat dianggap sebagai *user* yang menggunakan layanan Briker. Dibawah ini terdapat langkah-langkah untuk mengkonfigurasi *extension*:

1. Pilih opsi *Extensions* pada menu IPPBX Administration seperti yang ditunjukkan pada Gambar 4.13, sebagai berikut.



Gambar 4.13 Halaman *Extension*

2. Klik *Add Extension*
3. Pilih protokol yang akan digunakan seperti SIP, IAX2, ZAP, atau Custom. Pada penelitian ini penulis menggunakan protokol SIP
4. Tekan tombol *submit*
5. Kemudian akan muncul menu dimana dapat menambahkan *extension* dengan mengatur untuk nomor, nama, *password*, dan tambahan protokol agar bisa melakukan panggilan *video call*.

Briaer 2.1.1 "Merapi"

Home | IPPBX Administration | Billing | CDR | ACD Statistics | User Portal | Fax

Setup | Tools

IPPBX Status

Extensions

Bulk Extensions

Custom Contexts

Device Auto Provisioning

Extensions

Feature Codes

General Settings

Outbound Routes

Trunks

Inbound Call Control

Inbound Routes

Zap Channel DIDs

Announcements

Blacklist

CallerID Lookup Sources

Day/Night Control

Follow Me

IVR

Queues / ACD

Ring Groups

Time Conditions

Internal Options & Configuration

Callback

Conferences

DISA

Misc Applications

Misc Destinations

Music on Hold

PIN Sets

Paging and Intercom

Parking Lot

System Recordings

Add SIP Extension

Add Extension

User Extension: 11000

Display Name: hp4

CID Num Alias:

SIP Alias:

Extension Options

Direct DID:

DID Alert Info:

Music on Hold: acc_1

Outbound CID:

Ring Time: Default

Call Waiting: Enable

Emergency CID:

Device Options

This device uses sip technology:

calllimit: 1

calltimer: 0

accountcode: 11000

secret: 1234

dtmfmode: rfc2833

deny: 0.0.0.0/0.0.0.0

permit: 0.0.0.0/0.0.0.0

context: from-internal

disallow: all

allow: alaw&ulaw&h263p&h264&v

Gambar 4.14 Detail Halaman *Extension*

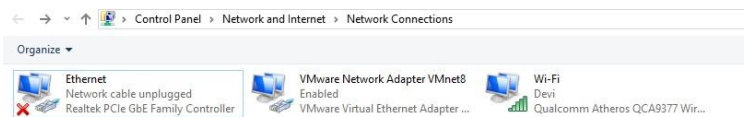
Keterangan untuk Gambar 4.14 :

- User Extension* : merupakan nomor *extension*, misal 11000, yang umumnya berbentuk nominal
- Display Name* : Nama yang digunakan pengguna sebagai *Caller ID* untuk panggilan
- Secret* : Kata sandi yang dibuat sistem untuk proses *login User Agent*
- Account Code* : Kode yang digunakan pada proses *billing*
- Allow* : Protocol yang digunakan dan untuk *video call* ditambahkan protocol H264.

4.2.4 Konfigurasi *Adapter Wi-Fi*

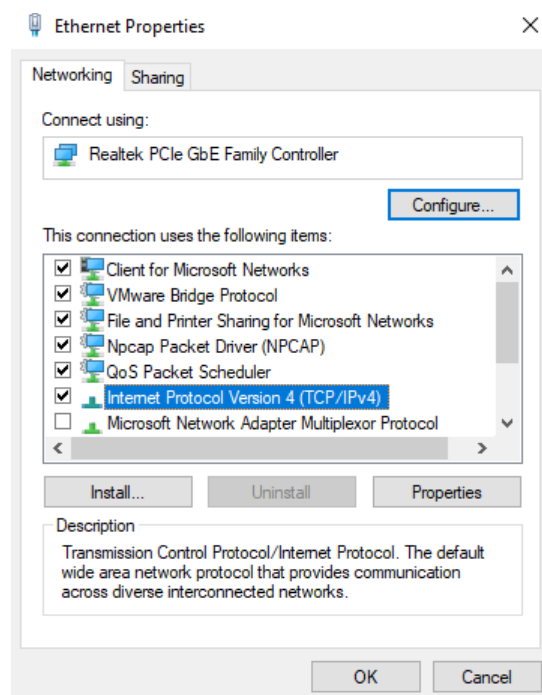
1. Pastikan bahwa koneksi Ethernet sudah terpasang ke server komputer yang dapat dilihat pada Gambar 4.15. Selanjutnya, lakukan pengaturan IP komputer server dengan mencari *Control Panel* pada kolom pencarian. Lalu tekan *Network and Internet*,

selanjutnya tekan *Network & Sharing Center* dan pilih *Change Adapter Setting*. Kemudian jendela *Network Connections* muncul.



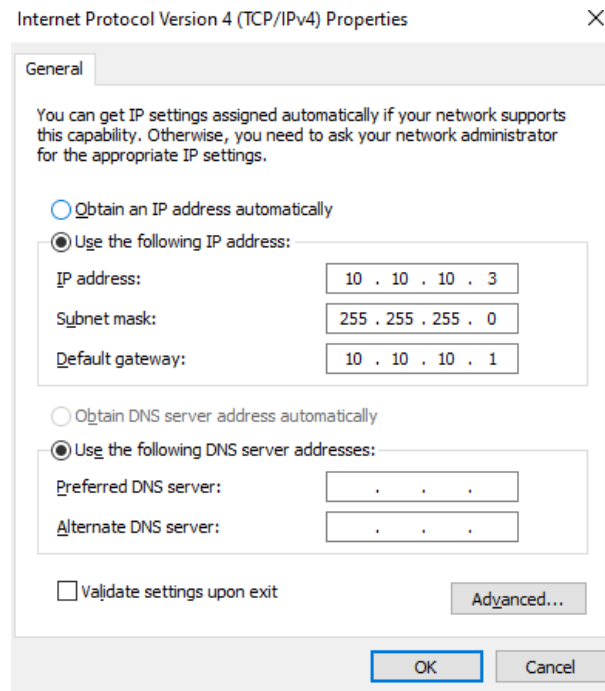
Gambar 4.15 *Network Connection*

2. Gambar 4.16 menampilkan pemilihan koneksi Ethernet, dengan membuka jendela *Ethernet Status* dan klik *Properties*. Selanjutnya, pilih *Internet Protocol Version 4 (TCP/IPv4)* dan klik *Properties*.



Gambar 4.16 *Ethernet Properties*

3. Untuk Gambar 4.17 dilakukan pemilihan opsi *Use the following IP address*, dan mengisi bagian *IP address*, *Subnet mask*, *Default gateway* lalu klik *Ok*. Selanjutnya, keluar dari jendela *Ethernet Status*.



Gambar 4.17 TCP/Ipv4

4. Setelah berhasil mengkonfigurasi *IP address* pada komputer *server* dan *Client*, langkah berikutnya adalah menjalankan *software* PortSIP UC yang telah di install sebelumnya.

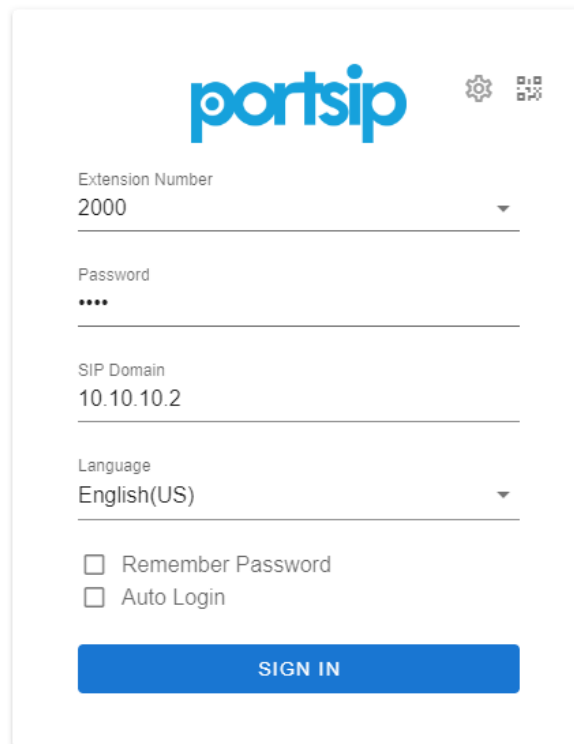
4.2.5 Konfigurasi PortSIP UC

Untuk mengatur *account* pada PortSIP, dengan memasukkan *username* pengguna yang telah terdaftar pada *server* Briker, dibawah ini langkah-langkah yang dipakai dan tampilan *login* dari PortSIP dapat dilihat pada Gambar 4.18, sebagai berikut:

1. Masukkan *Extension number*, *password*, SIP Domain (IP Briker) yang telah dibuat.
2. Isi *Account* seperti contoh dibawah ini :

Extention Number	2000
Password	1234
SIP Domain	: 10.10.10.2

Lalu kli tombol *Sign in*



portsip

Extension Number
2000

Password
....

SIP Domain
10.10.10.2

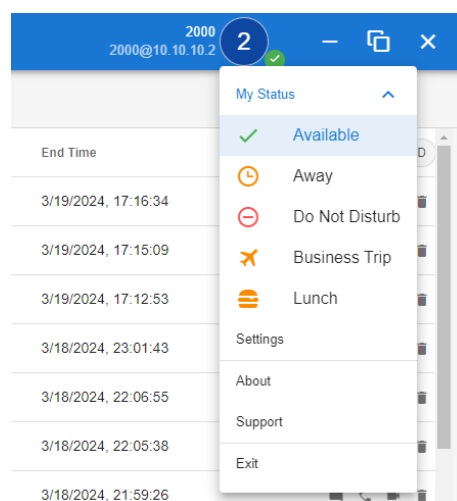
Language
English(US)

☐ Remember Password
☐ Auto Login

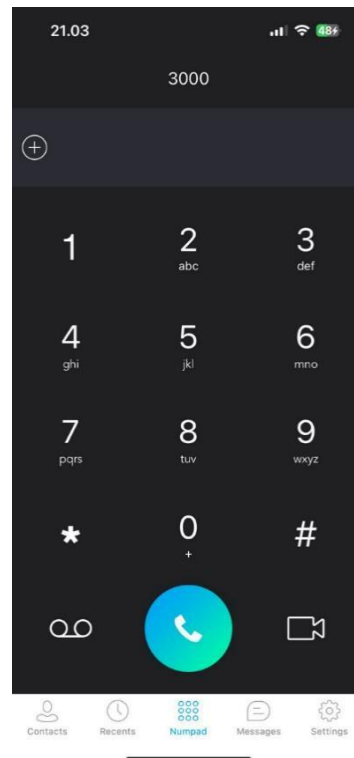
SIGN IN

Gambar 4.18 Login PortSIP

- Setelah menyelesaikan konfigurasi, langkah selanjutnya adalah mengecek apakah PortSIP berhasil terhubung ke *server* dengan meng klik opsi My Status. Pada Gambar 4.19, memperlihatkan bahwa PortSIP sudah *Available* kepada *server* Briker. Dan untuk Gambar 4.20, merupakan tampilan halaman *client* yang digunakan ketika melakukan panggilan.



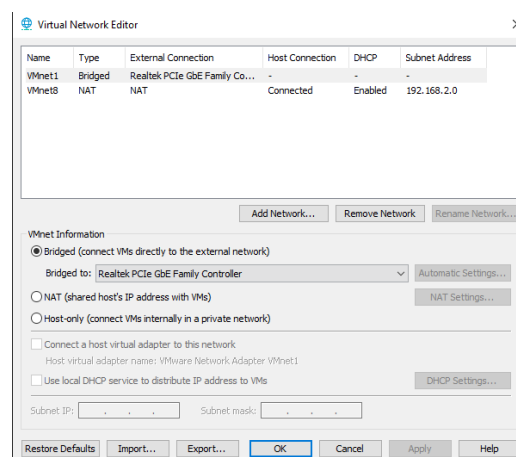
Gambar 4.19 Tampilan *Client* VoIP yang sudah teregistrasi pada Laptop



Gambar 4.20 Tampilan *Client* yang sudah teregistrasi di *Smartphone*

4.2.6 Uji Coba Koneksi Jaringan Lokal FASILKOM-TI USU

Pada Gambar 4.21, Gambar 4.22, dan Gambar 4.23 memperlihatkan bahwa *server* Briker VoIP yang dibangun berhasil untuk digunakan pada lingkungan Fakultas Ilmu Komputer dan Teknologi Informasi tepatnya pada Gedung-D menggunakan jaringan *Wide Area Network* (WAN).



Gambar 4.21 Tampilan *Bridge Adapter*

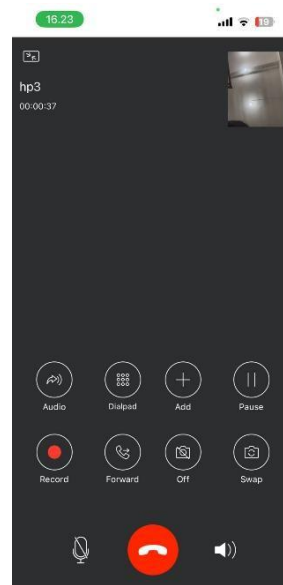
```

Ethernet adapter Ethernet:

Connection-specific DNS Suffix  . : 
Description . . . . . : Realtek PCIe GbE Family Controller
Physical Address. . . . . : 7C-8A-E1-38-D1-BB
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
Link-local IPv6 Address . . . . . : fe80::2fd0:b592:4c3d:3aa9%15(Preferred)
IPv4 Address. . . . . : 10.1.15.21(Preferred)
Subnet Mask . . . . . : 255.255.255.0
Lease Obtained. . . . . : 30 May 2024 15:41:46
Lease Expires . . . . . : 30 May 2024 16:21:47
Default Gateway . . . . . : 10.1.15.1
DHCP Server . . . . . : 10.1.15.1
DHCPv6 IAID . . . . . : 276597473
DHCPv6 Client DUID. . . . . : 00-01-00-01-26-37-1F-E8-7C-8A-E1-38-D1-BB
DNS Servers . . . . . : 10.100.100.1
                        10.200.200.2
NetBIOS over Tcpip. . . . . : Enabled

```

Gambar 4.22 *ipconfig /all* pada cmd



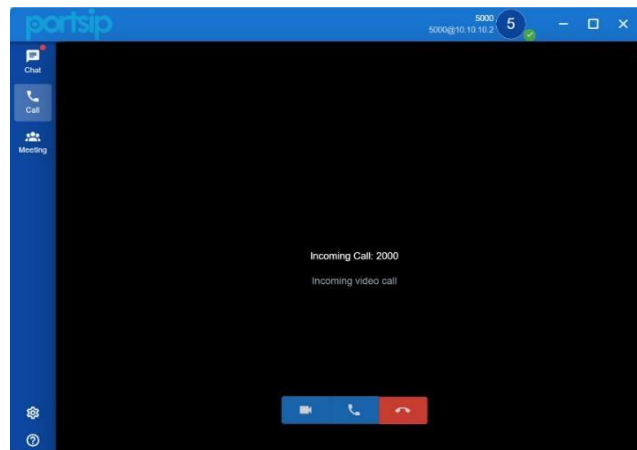
Gambar 4.23 Hasil Panggilan Masuk *Smartphone Client* ke *Smartphone Client*

4.2.7 Hasil Pengujian Panggilan

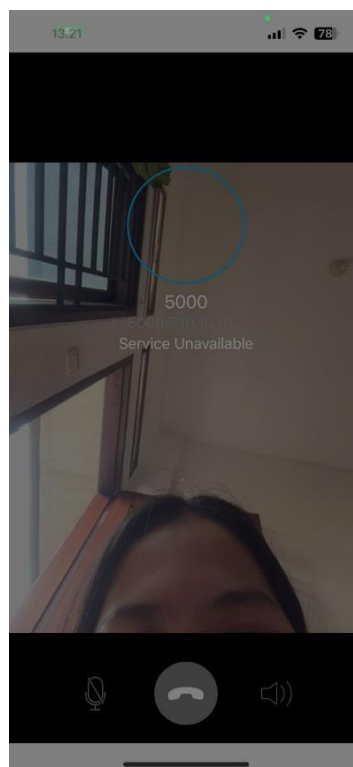
Hasil pengujian perangkat dalam sistem VoIP menunjukkan kemampuan yang baik dalam menerima, mengakhiri panggilan, dan berkomunikasi dengan lancar menggunakan *softphone* PortSIP UC, baik dari sisi *server* maupun *client*.

Tabel 4.2 Daftar Pengujian Panggilan

No.	Nama Perangkat	Panggilan Keluar	Panggilan Masuk
1	Komputer <i>Server</i>	Berhasil	Berhasil
2	Komputer <i>Client</i>	Berhasil	Berhasil
3	<i>Smartphone Client</i>	Berhasil	Berhasil



Gambar 4.24 Hasil Panggilan Masuk *PC Server ke PC Client*



Gambar 4.25 Hasil Panggilan Keluar *Smartphone Client ke PC Client*



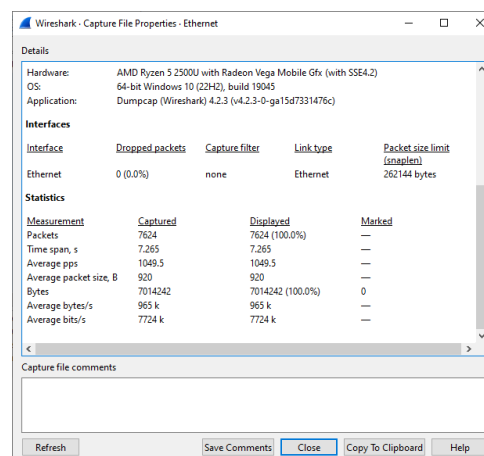
Gambar 4.26 Hasil Panggilan Masuk PC Server ke Smartphone Client

Untuk Gambar 4.24, Gambar 4.25, dan Gambar 4.26 menunjukkan bagaimana tampilan dari setiap *client* yang sudah mencoba untuk melakukan panggilan *video call*, baik itu panggilan masuk maupun panggilan keluar.

4.2.8 Pengukuran dan Analisis Parameter QoS berdasarkan Jarak

a. Perhitungan Parameter Berdasarkan Jarak 10 meter :

Gambar 4.27 menampilkan data yang berhasil di *capture* oleh *Wireshark* sehingga bisa digunakan untuk mencari nilai *jitter*, *delay*, dan *throughput*.



Gambar 4.27 Jarak 10 Meter

Perhitungan untuk nilai *Delay* berdasarkan persamaan (2.2)

$$\begin{aligned} \text{Delay} &= \frac{7,6058}{8965} = 0,00084 \text{ s} \\ &= 0,84 \text{ ms} \end{aligned}$$

Perhitungan nilai *Throughput* menggunakan persamaan (2.1)

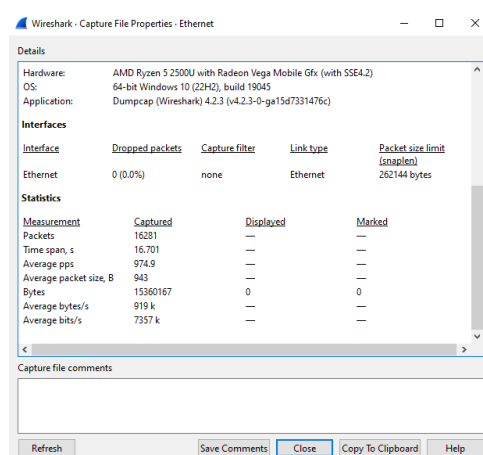
$$\begin{aligned} \text{Throughput} &= \frac{\text{Jumlah bit yang dikirim}}{\text{Waktu pengiriman paket}} = \frac{7014242}{7,265} \\ &= 965484,102 \text{ bytes/s} = 965484,102 \times 8 \\ &= 7723872,81 \text{ bits/s} = 7723,87 \text{ Kbits/s} \end{aligned}$$

Perhitungan nilai *Jitter* menggunakan persamaan (2.3)

$$\begin{aligned} \text{Rata-rata Jitter} &= \frac{7,606406}{8965} = 0,00085 \text{ s} \\ &= 0,85 \text{ ms} \end{aligned}$$

b. Perhitungan Berdasarkan Jarak 20 meter

Gambar 4.28 menampilkan data yang berhasil di *capture* oleh *Wireshark* sehingga bisa digunakan untuk mencari nilai *jitter*, *delay*, dan *throughput*.



Gambar 4.28 Jarak 20 meter

Perhitungan untuk nilai *Delay* berdasarkan persamaan (2.2)

$$\text{Delay} = \frac{8,071902}{8832} = 0,00091 \text{ s}$$

$$= 0,91 \text{ ms}$$

Perhitungan nilai *Throughput* menggunakan persamaan (2.1)

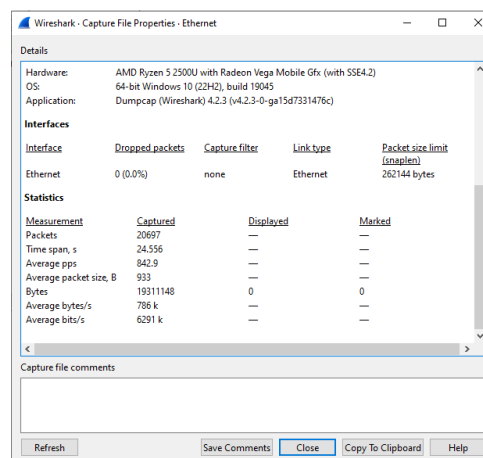
$$\begin{aligned} \text{Throughput} &= \frac{\text{Jumlah bit yang dikirim}}{\text{Waktu pengiriman paket}} = \frac{15360167}{16,701} \\ &= 919715,406 \text{ bytes/s} = 919715,406 \times 8 \\ &= 7357723,25 \text{ bits/s} = 7357,72 \text{ Kbits/s} \end{aligned}$$

Perhitungan nilai *Jitter* menggunakan persamaan (2.3)

$$\begin{aligned} \text{Rata-rata Jitter} &= \frac{8,072718}{8832} = 0,00091 \text{ s} \\ &= 0,91 \text{ ms} \end{aligned}$$

c. Perhitungan Berdasarkan Jarak 30 meter

Gambar 4.29 menampilkan data yang berhasil di *capture* oleh *Wireshark* sehingga bisa digunakan untuk mencari nilai *jitter*, *delay*, dan *throughput*.



Gambar 4.29 Jarak 30 meter

Perhitungan untuk nilai *Delay* berdasarkan persamaan (2.2)

$$\begin{aligned} \text{Delay} &= \frac{8,001877}{8160} = 0,00098 \text{ s} \\ &= 0,98 \text{ ms} \end{aligned}$$

Perhitungan nilai *Throughput* menggunakan persamaan (2.1)

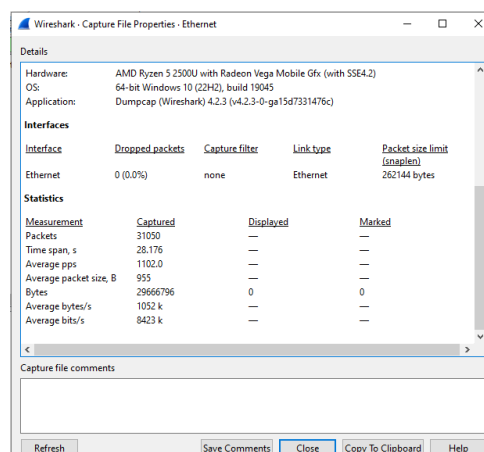
$$\begin{aligned}
 Throughput &= \frac{\text{Jumlah bit yang dikirim}}{\text{Waktu pengiriman paket}} = \frac{19311148}{24,556} \\
 &= 786412,608 \text{ bytes/s} = 786412,608 \times 8 \\
 &= 6291300,86 \text{ bits/s} = 6291,3 \text{ Kbits/s}
 \end{aligned}$$

Perhitungan nilai *Jitter* menggunakan persamaan (2.3)

$$\begin{aligned}
 \text{Rata-rata } Jitter &= \frac{8,001849}{8160} = 0,00098 \text{ s} \\
 &= 0,98 \text{ ms}
 \end{aligned}$$

d. Pengukuran Berdasarkan Jarak 40 meter

Gambar 4.30 menampilkan data yang berhasil di *capture* oleh *Wireshark* sehingga bisa digunakan untuk mencari nilai *jitter*, *delay*, dan *throughput*.



Gambar 4.30 Jarak 40 meter

Perhitungan untuk nilai *Delay* berdasarkan persamaan (2.2)

$$\begin{aligned}
 Delay &= \frac{11,7695}{11394} = 0,00103 \text{ s} \\
 &= 1,03 \text{ ms}
 \end{aligned}$$

Perhitungan nilai *Throughput* menggunakan persamaan (2.1)

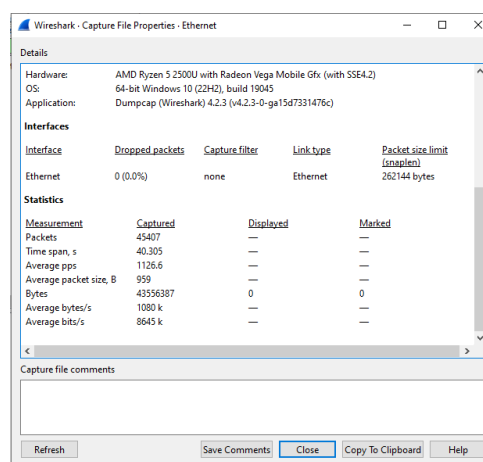
$$\begin{aligned}
 Throughput &= \frac{\text{Jumlah bit yang dikirim}}{\text{Waktu pengiriman paket}} = \frac{29666796}{28,176} \\
 &= 1052910,14 \text{ bytes/s} = 1052910,14 \times 8 \\
 &= 8423281,09 \text{ bits/s} = 8423,28 \text{ Kbits/s}
 \end{aligned}$$

Perhitungan nilai *Jitter* menggunakan persamaan (2.3)

$$\begin{aligned}\text{Rata-rata } Jitter &= \frac{12,106185}{11394} = 0,00106 \text{ s} \\ &= 1,06 \text{ ms}\end{aligned}$$

e. Perhitungan Berdasarkan Jarak 50 meter

Gambar 4.31 menampilkan data yang berhasil di *capture* oleh *Wireshark* sehingga bisa digunakan untuk mencari nilai *jitter*, *delay*, dan *throughput*.



Gambar 4.31 Jarak 50 meter

Perhitungan untuk nilai *Delay* berdasarkan persamaan (2.2)

$$\begin{aligned}\text{Delay} &= \frac{10,050052}{10285} = 0,00097 \text{ s} \\ &= 0,97 \text{ ms}\end{aligned}$$

Perhitungan nilai *Throughput* menggunakan persamaan (2.1)

$$\begin{aligned}\text{Throughput} &= \frac{\text{Jumlah bit yang dikirim}}{\text{Waktu pengiriman paket}} = \frac{43556387}{40,305} \\ &= 1080669,57 \text{ bytes/s} = 1080669,57 \times 8 \\ &= 8645356,56 \text{ bits/s} = 8645,36 \text{ Kbits/s}\end{aligned}$$

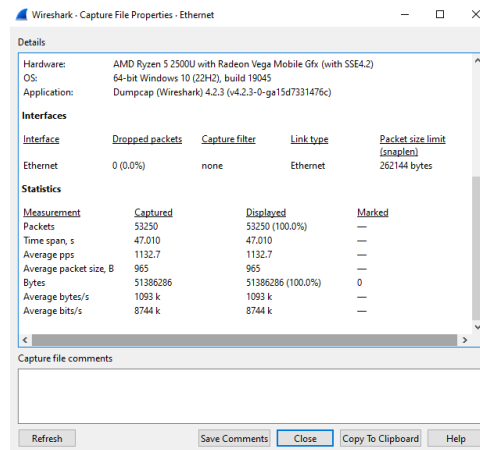
Perhitungan nilai *Jitter* menggunakan persamaan (2.3)

$$\text{Rata-rata } Jitter = \frac{10,050293}{10285} = 0,00098 \text{ s}$$

$$= 0,98 \text{ ms}$$

f. Perhitungan Berdasarkan Jarak 60 meter

Gambar 4.32 menampilkan data yang berhasil di *capture* oleh *Wireshark* sehingga bisa digunakan untuk mencari nilai *jitter*, *delay*, dan *throughput*.



Gambar 4.32 Jarak 60 meter

Perhitungan untuk nilai *Delay* berdasarkan persamaan (2.2)

$$\text{Delay} = \frac{10,895303}{11486} = 0,00094 \text{ s}$$

$$= 0,94 \text{ ms}$$

Perhitungan nilai *Throughput* menggunakan persamaan (2.1)

$$\text{Throughput} = \frac{\text{Jumlah bit yang dikirim}}{\text{Waktu pengiriman paket}} = \frac{51386286}{47,010}$$

$$= 1093092,66 \text{ bytes/s} = 1093092,66 \times 8$$

$$= 8744741,29 \text{ bits/s} = 8744,74 \text{ Kbits/s}$$

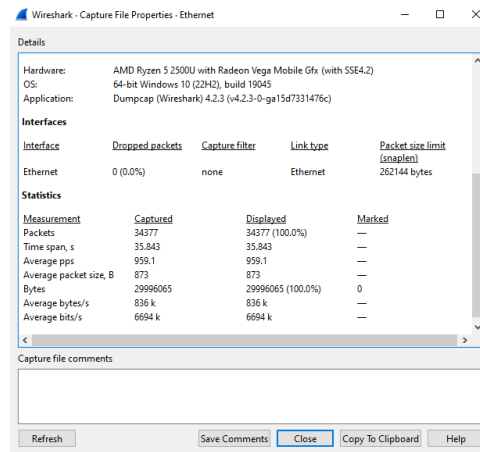
Perhitungan nilai *Jitter* menggunakan persamaan (2.3)

$$\text{Rata-rata Jitter} = \frac{10,895261}{11486} = 0,00094 \text{ s}$$

$$= 0,94 \text{ ms}$$

g. Perhitungan Berdasarkan Jarak 70 meter

Gambar 4.33 menampilkan data yang berhasil di *capture* oleh *Wireshark* sehingga bisa digunakan untuk mencari nilai *jitter*, *delay*, dan *throughput*.



Gambar 4.33 Jarak 70 meter

Perhitungan untuk nilai *Delay* berdasarkan persamaan (2.2)

$$\text{Delay} = \frac{13,867366}{14824} = 0,00093 \text{ s}$$

$$= 0,93 \text{ ms}$$

Perhitungan nilai *Throughput* menggunakan persamaan (2.1)

$$\text{Throughput} = \frac{\text{Jumlah bit yang dikirim}}{\text{Waktu pengiriman paket}} = \frac{29996065}{35,843}$$

$$= 836873,72 \text{ bytes/s} = 836873,72 \times 8$$

$$= 6694989,82 \text{ bits/s} = 6694 \text{ Kbits/s}$$

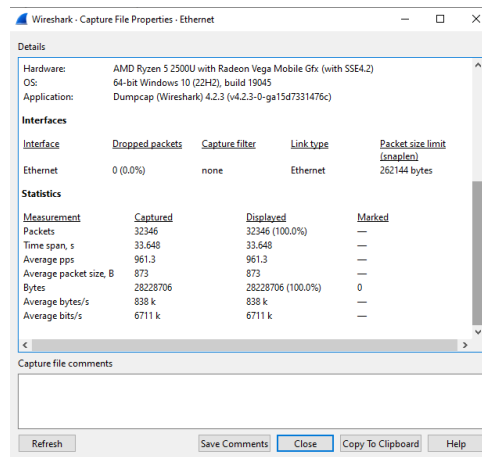
Perhitungan nilai *Jitter* menggunakan persamaan (2.3)

$$\text{Rata-rata Jitter} = \frac{14,355277}{14824} = 0,00096 \text{ s}$$

$$= 0,96 \text{ ms}$$

h. Perhitungan Berdasarkan Jarak 80 meter

Gambar 4.34 menampilkan data yang berhasil di *capture* oleh *Wireshark* sehingga bisa digunakan untuk mencari nilai *jitter*, *delay*, dan *throughput*.



Gambar 4.34 Jarak 80 meter

Perhitungan untuk nilai *Delay* berdasarkan persamaan (2.2)

$$Delay = \frac{19,415482}{21408} = 0,00090 \text{ s}$$

$$= 0,90 \text{ ms}$$

Perhitungan nilai *Throughput* menggunakan persamaan (2.1)

$$Throughput = \frac{\text{Jumlah bit yang dikirim}}{\text{Waktu pengiriman paket}} = \frac{28228706}{33,648}$$

$$= 838941,57 \text{ bytes/s} = 838941,57 \times 8$$

$$= 6711532,56 \text{ bits/s} = 6711,53 \text{ Kbits/s}$$

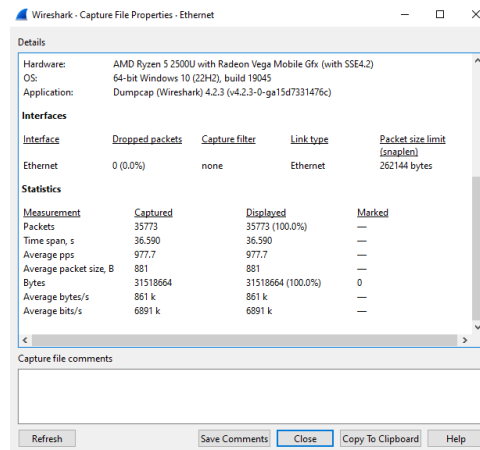
Perhitungan nilai *Jitter* menggunakan persamaan (2.3)

$$\text{Rata-rata } Jitter = \frac{19,425063}{21408} = 0,00090 \text{ s}$$

$$= 0,90 \text{ ms}$$

i. Perhitungan Berdasarkan Jarak 90 meter

Gambar 4.35 menampilkan data yang berhasil di *capture* oleh *Wireshark* sehingga bisa digunakan untuk mencari nilai *jitter*, *delay*, dan *throughput*.



Gambar 4.35 Jarak 90 meter

Perhitungan untuk nilai *Delay* berdasarkan persamaan (2.2)

$$Delay = \frac{19,750587}{21524} = 0,00091 \text{ s}$$

$$= 0,91 \text{ ms}$$

Perhitungan nilai *Throughput* menggunakan persamaan (2.1)

$$Throughput = \frac{\text{Jumlah bit yang dikirim}}{\text{Waktu pengiriman paket}} = \frac{31518664}{36,590}$$

$$= 861401,03 \text{ bytes/s} = 861401,03 \times 8$$

$$= 6891208,24 \text{ bits/s} = 6891,21 \text{ Kbits/s}$$

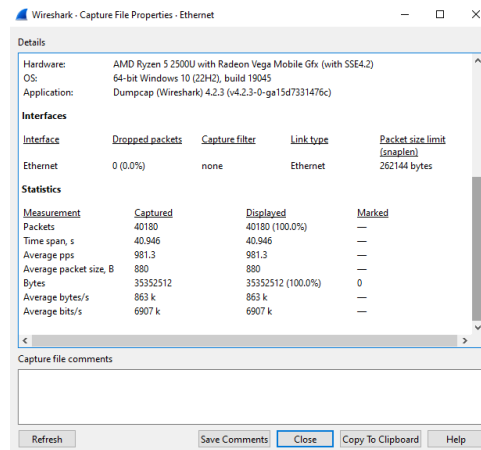
Perhitungan nilai *Jitter* menggunakan persamaan (2.3)

$$\text{Rata-rata } Jitter = \frac{19,752858}{21524} = 0,00092 \text{ s}$$

$$= 0,92 \text{ ms}$$

j. Perhitungan Berdasarkan Jarak 100 meter

Gambar 4.36 menampilkan data yang berhasil di *capture* oleh *Wireshark* sehingga bisa digunakan untuk mencari nilai *jitter*, *delay*, dan *throughput*.



Gambar 4.36 Jarak 100 meter

Perhitungan untuk nilai *Delay* berdasarkan persamaan (2.2)

$$Delay = \frac{22,584132}{23293} = 0,00096 \text{ s}$$

$$= 0,96 \text{ ms}$$

Perhitungan nilai *Throughput* menggunakan persamaan (2.1)

$$Throughput = \frac{\text{Jumlah bit yang dikirim}}{\text{Waktu pengiriman paket}} = \frac{35352512}{40,946}$$

$$= 863393,543 \text{ bytes/s} = 863393,543 \times 8$$

$$= 6907148,34 \text{ bits/s} = 6907,15 \text{ Kbits/s}$$

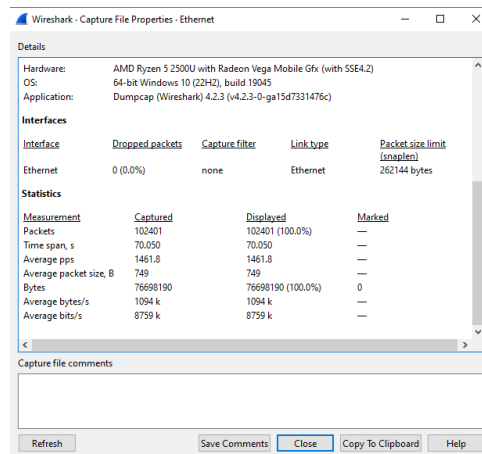
Perhitungan nilai *Jitter* menggunakan persamaan (2.3)

$$\text{Rata-rata Jitter} = \frac{22,602592}{23293} = 0,00097 \text{ s}$$

$$= 0,97 \text{ ms}$$

k. Perhitungan Berdasarkan Jarak 110 meter

Gambar 4.37 menampilkan data yang berhasil di *capture* oleh *Wireshark* sehingga bisa digunakan untuk mencari nilai *jitter*, *delay*, dan *throughput*.



Gambar 4.37 Jarak 110 meter

Perhitungan untuk nilai *Delay* berdasarkan persamaan (2.2)

$$Delay = \frac{70,050436}{102400} = 0,00068 \text{ s}$$

$$= 0,68 \text{ ms}$$

Perhitungan nilai *Throughput* menggunakan persamaan (2.1)

$$Throughput = \frac{\text{Jumlah bit yang dikirim}}{\text{Waktu pengiriman paket}} = \frac{76698190}{70,050}$$

$$= 1094906,35 \text{ bytes/s} = 1094906,35 \times 8$$

$$= 8759250,8 \text{ bits/s} = 8759,25 \text{ Kbits/s}$$

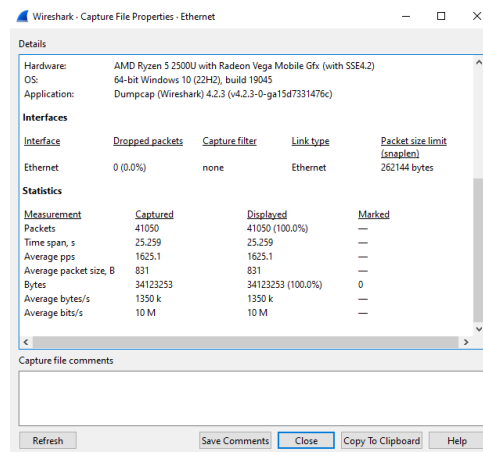
Perhitungan nilai *Jitter* menggunakan persamaan (2.3)

$$\text{Rata-rata Jitter} = \frac{70,050021}{102400} = 0,00068 \text{ s}$$

$$= 0,68 \text{ ms}$$

1. Perhitungan Berdasarkan Jarak 120 meter

Gambar 4.38 menampilkan data yang berhasil di *capture* oleh *Wireshark* sehingga bisa digunakan untuk mencari nilai *jitter*, *delay*, dan *throughput*.



Gambar 4.38 Jarak 120 meter

Perhitungan untuk nilai *Delay* berdasarkan persamaan (2.2)

$$Delay = \frac{25,259267}{41049} = 0,00061 \text{ s}$$

$$= 0,61 \text{ ms}$$

Perhitungan nilai *Throughput* menggunakan persamaan (2.1)

$$Throughput = \frac{\text{Jumlah bit yang dikirim}}{\text{Waktu pengiriman paket}} = \frac{34123253}{25,259}$$

$$= 1350934,43 \text{ bytes/s} = 1350934,43 \times 8$$

$$= 10807475,44 \text{ bits/s} = 10807,48 \text{ Kbits/s}$$

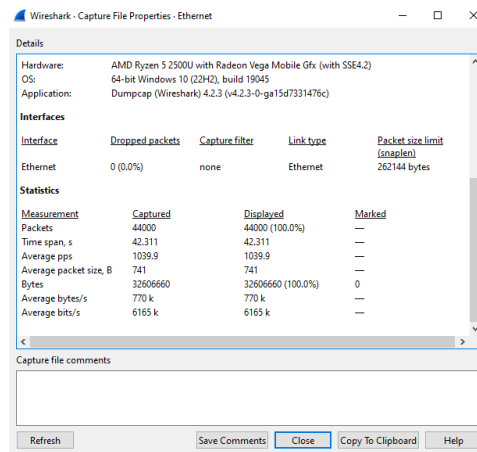
Perhitungan nilai *Jitter* menggunakan persamaan (2.3)

$$\text{Rata-rata Jitter} = \frac{25,259416}{41049} = 0,00062 \text{ s}$$

$$= 0,62 \text{ ms}$$

m. Perhitungan Berdasarkan Jarak 130 meter

Gambar 4.39 menampilkan data yang berhasil di *capture* oleh *Wireshark* sehingga bisa digunakan untuk mencari nilai *jitter*, *delay*, dan *throughput*.



Gambar 4.39 Jarak 130 meter

Perhitungan untuk nilai *Delay* berdasarkan persamaan (2.2)

$$Delay = \frac{42,311114}{43999} = 0,00096 \text{ s}$$

$$= 0,96 \text{ ms}$$

Perhitungan nilai *Throughput* menggunakan persamaan (2.1)

$$Throughput = \frac{\text{Jumlah bit yang dikirim}}{\text{Waktu pengiriman paket}} = \frac{32606660}{42,311}$$

$$= 770642,62 \text{ bytes/s} = 770642,62 \times 8$$

$$= 6165140,98 \text{ bits/s} = 6165,14 \text{ Kbits/s}$$

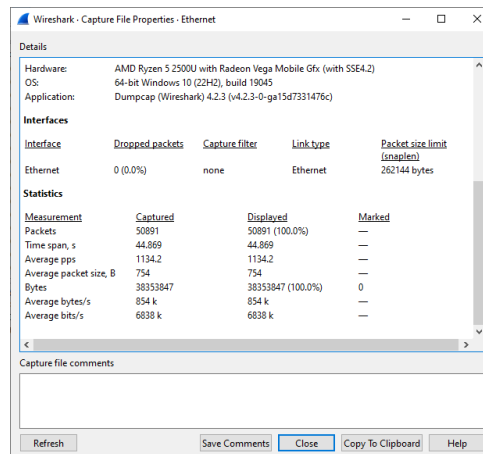
Perhitungan nilai *Jitter* menggunakan persamaan (2.3)

$$\text{Rata-rata } Jitter = \frac{42,300736}{43999} = 0,00096 \text{ s}$$

$$= 0,96 \text{ ms}$$

n. Perhitungan Berdasarkan Jarak 140 meter

Gambar 4.40 menampilkan data yang berhasil di *capture* oleh *Wireshark* sehingga bisa digunakan untuk mencari nilai *jitter*, *delay*, dan *throughput*.



Gambar 4.40 Jarak 140 meter

Perhitungan untuk nilai *Delay* berdasarkan persamaan (2.2)

$$Delay = \frac{44,869169}{50890} = 0,00088 \text{ s}$$

$$= 0,88 \text{ ms}$$

Perhitungan nilai *Throughput* menggunakan persamaan (2.1)

$$Throughput = \frac{\text{Jumlah bit yang dikirim}}{\text{Waktu pengiriman paket}} = \frac{38353847}{44,869}$$

$$= 854796,118 \text{ bytes/s} = 854796,118 \times 8$$

$$= 6838368,94 \text{ bits/s} = 6838,37 \text{ Kbits/s}$$

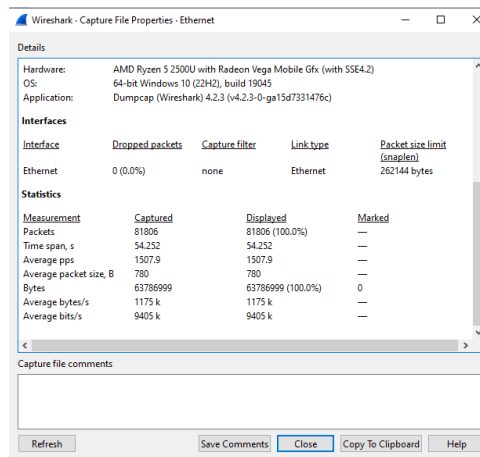
Perhitungan nilai *Jitter* menggunakan persamaan (2.3)

$$\text{Rata-rata Jitter} = \frac{44,873263}{50890} = 0,00088 \text{ s}$$

$$= 0,88 \text{ ms}$$

o. Perhitungan Berdasarkan Jarak 150 meter

Gambar 4.41 menampilkan data yang berhasil di *capture* oleh *Wireshark* sehingga bisa digunakan untuk mencari nilai *jitter*, *delay*, dan *throughput*.



Gambar 4.41 Jarak 150 meter

Perhitungan untuk nilai *Delay* berdasarkan persamaan (2.2)

$$Delay = \frac{54,252431}{81805} = 0,00066 \text{ s}$$

$$= 0,66 \text{ ms}$$

Perhitungan nilai *Throughput* menggunakan persamaan (2.1)

$$Throughput = \frac{\text{Jumlah bit yang dikirim}}{\text{Waktu pengiriman paket}} = \frac{63786999}{54,252}$$

$$= 1175753,87 \text{ bytes/s} = 1175753,87 \times 8$$

$$= 9406030,97 \text{ bits/s} = 9406,03 \text{ Kbits/s}$$

Perhitungan nilai *Jitter* menggunakan persamaan (2.3)

$$\text{Rata-rata Jitter} = \frac{54,252795}{81805} = 0,00066 \text{ s}$$

$$= 0,66 \text{ ms}$$

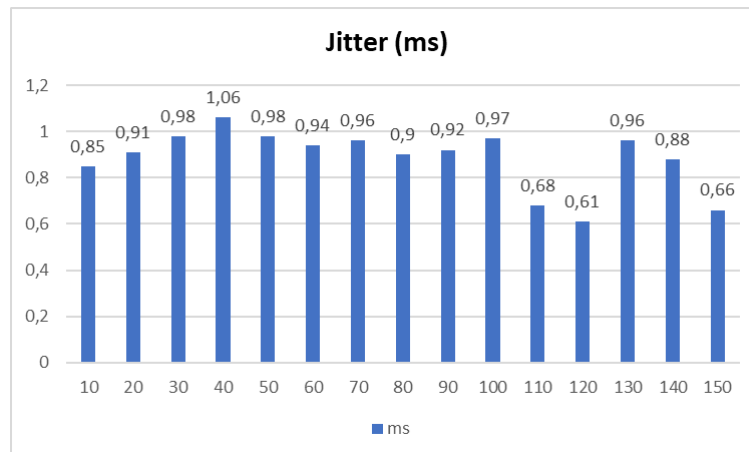
Tabel 4.3 Hasil Perhitungan Setiap Parameter QoS

Jarak (m)	<i>Jitter</i> (ms)	<i>Throughput</i> (Kbit/sec)	<i>Delay</i> (ms)	Kualitas suara
10	0,85	7723,87	0,84	Baik
20	0,91	7357,72	0,91	Baik
30	0,98	6291,3	0,98	Baik
40	1,06	8423,28	1,03	Baik
50	0,98	8645,36	0,97	Baik
60	0,94	8744,74	0,94	Baik
70	0,96	6694	0,93	Baik
80	0,90	6711,53	0,90	Baik
90	0,92	6891,21	0,91	Baik
100	0,97	6907,15	0,96	Baik
110	0,68	8759,25	0,68	Baik
120	0,62	10807,48	0,61	Baik
130	0,96	6165,14	0,96	Baik
140	0,88	6838,37	0,88	Baik
150	0,66	9406,03	0,66	Baik

1. Pengukuran dan Analisis *Jitter*

Jitter adalah variasi waktu kedatangan paket data yang diukur dengan selisih waktu antara kedatangan paket yang diterima saat ini dengan waktu kedatangan paket sebelumnya. *Jitter* ini disebabkan karena waktu tempuh perjalanan paket yang berbeda, variasi panjang antrian, dan waktu pemrosesan data.

Tabel 4.3 dan Gambar 4.39 menunjukkan pengukuran parameter *jitter*, yang digunakan untuk melakukan pengujian *jitter*. Perhitungan dilakukan lewat aplikasi *wireshark* dengan meng-*capture* waktu rata-rata antara kedatangan beberapa paket VoIP.



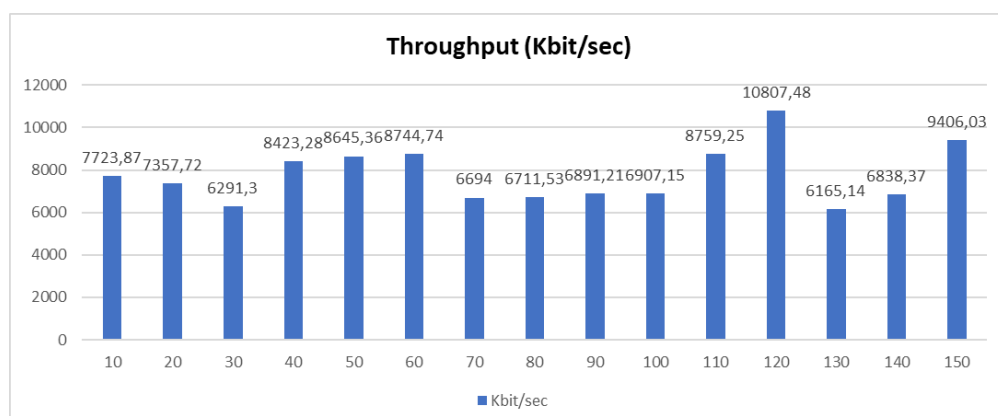
Gambar 4.42 Hasil Pengukuran *Jitter* dengan Jarak

Terlihat dari Tabel 4.3 dan Gambar 4.39 terdapat perubahan nilai *jitter* yang cukup signifikan. *Jitter* memiliki nilai terendah pada jarak 120 meter, yaitu sebesar 0,61 ms, namun mengalami kenaikan yang paling besar pada jarak 40 meter, mencapai 1,06 ms.

Rentang nilai *jitter* dari pengamatan pada berbagai jarak adalah antara 0,61 ms hingga 1,06 ms. Hasil pengujian ini menunjukkan bahwa nilai *jitter* memenuhi standar kualitas yang ditetapkan untuk *Quality of Service*.

2. Pengukuran dan Analisis *Throughput*

Informasi yang dipaparkan dalam Tabel 4.3 dan Gambar 4.40 diperoleh dari pengukuran yang dilakukan dengan *Wireshark*.



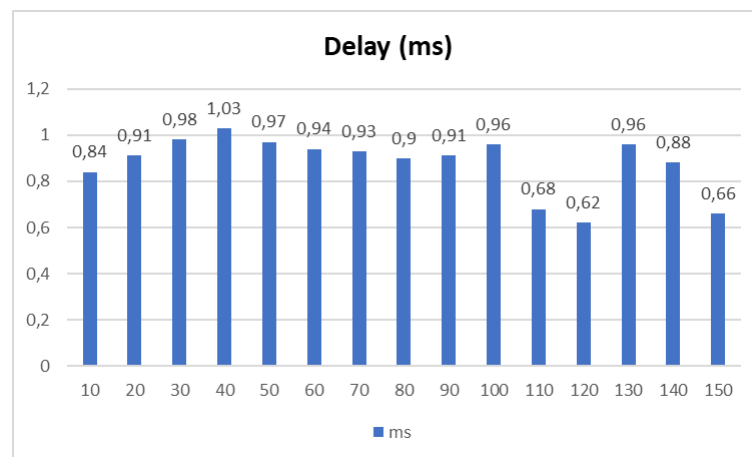
Gambar 4.43 Hasil Pengukuran *Throughput* dengan Jarak

Dari grafik di atas, terlihat bahwa *throughput* mencapai nilai tertinggi pada jarak 120 meter dengan panggilan VoIP mencapai 10807,48 Kbps.

Sebaliknya, nilai *throughput* terendah terjadi pada jarak 130 meter, yakni sebesar 6165,14 Kbps. Untuk rentang ukuran *throughput* yang stabil berada pada jarak 70 hingga 100 meter. Jumlah paket dan ukuran yang dikirim per detik memengaruhi *throughput*. Semakin besar ukuran data dan jumlah paket yang dikirim, maka nilai *throughput* akan lebih besar, begitu pula sebaliknya.

3. Pengukuran dan Analisis Delay

Delay adalah durasi yang diperlukan dari sebuah paket data untuk beralih dari *caller* ke *receiver*. *Delay* ialah komponen penting dalam menilai mutu performa jaringan VoIP. Data pengukuran delay terdokumentasikan di Tabel 4.3 dan Gambar 4.41.



Gambar 4.44 Hasil Pengukuran *Delay* dengan Jarak

Pada data Tabel 4.3 serta diagram grafik pada Gambar 4.41, nilai delay maksimum terjadi pada jarak 40 meter yaitu 1,03 ms. Hasil pengujian ini menunjukkan nilai *delay* memenuhi standar kualitas *Quality of Service*.

BAB V

PENUTUP

5.1 Kesimpulan

Dari setiap proses yang telah dilakukan dalam penelitian penggunaan sistem komunikasi VoIP dengan Briker, dapat disimpulkan bahwa :

1. Kabel UTP RJ-45 dan *Access Point* dapat digunakan sebagai media penghubung yang membuktikan bahwa briker berhasil dikonfigurasi sebagai server VoIP dan bekerja secara efektif. Hal ini memungkinkan seluruh perangkat untuk berkomunikasi satu sama lain dan melakukan panggilan masuk dan keluar dengan lancar.
2. Sistem VoIP yang dirancang memperlihatkan bahwa komunikasi dapat dilakukan baik antara laptop ke laptop, laptop ke *smartphone*, dan *smartphone* ke *smartphone* yang terhubung ke jaringan LAN yang telah dibangun dalam penelitian ini.
3. VoIP *server* Briker berperan menangani panggilan SIP dari seluruh *client* yang terdaftar kedalam *server* Briker.
4. Pengujian parameter QoS dengan jarak 10-150 meter menunjukkan bahwa sistem VoIP yang dirancang menghasilkan jaringan dengan kualitas baik berdasarkan perhitungan *throughput*, *delay*, dan *jitter*.

5.2 Saran

Berikut ini adalah saran untuk penelitian selanjutnya :

1. Disarankan kepada peneliti selanjutnya untuk melakukan *testing* sistem dengan memperbanyak penempatan lokasi *server*.
2. Disarankan untuk memperkuat keamanan dengan menambahkan enkripsi pada data suara, seperti menggunakan IP Security, untuk mencegah akses tidak sah dan memastikan kerahasiaan percakapan di dalam sistem VoIP.

DAFTAR PUSTAKA

- Abualhaj, M. M.-K. (2020). Effective Voice Frame Pruning Method to Increase VoIP Call Capacity. *TEM Journal*.
- Abualhaj, M. M.-S.-K.-M. (2024). An innovative approach for enhancing capacity utilization in point-to-point voice over internet protocol calls. *International Journal of Electrical & Computer Engineering*.
- Adibasyah, M. (2023). Design and Analyze the Quality of Services (QoS) of a VoIP Server Using Briker by Utilizing Trunk and Interactive Voice Response (IVR) at the AirNav Padang. *Telekontran: Jurnal Ilmiah Telekomunikasi, Kendali dan Elektronika Terapan*, 115-128.
- Ali, B. Q. (2022). Covert voip communication based on audio steganography. *International Journal of Computing and Digital Systems*, 821-830.
- Andrianto, H. S. (2021). Evaluasi Kinerja GSM VoIP Gateway pada Sistem IP PBX. *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*.
- Arief, M. T. (2023). Implementasi Internet Protocol Versi 6 Pada Jaringan VoIP Berbasis Session Initiation Protocol. *JiTEKH*, 51-56.
- Berlian, B. (2020). Membangun Server VOIP Berbasis Asterisk. *Jurnal Media Infotama*.
- El Azzaoui, A. C. (2022). Scalable lightweight blockchain-based authentication mechanism for secure VoIP communication. *Hum.-Cent. Comput. Inf. Sci*.
- Fauzi, A. S. (2022). Perancangan Jaringan Voice Over Internet Protokol (Voip) Menggunakan Platform Asterisknow. *Computer Science (CO-SCIENCE)*, 19-25.
- Gawarle, A. S. (2017). Design a Free Voice Calling System Using Raspberry Pi. *International Journal of Research in Engineering and Applied Sciences (IJREAS)*, 10-14.
- Khomar, R. &. (2021). Implementasi dan Optimalisasi Voice Order Internet Protocol (VOIP) Menggunakan NDLC Untuk Komunikasi Antar Lantai. *TeknoIS: Jurnal Ilmiah Teknologi Informasi dan Sains*, 51-60.
- Kusuma, G. H. (2020). Implementasi Voip Elastix Server Pada PT XYZ. *Journal of Informatics and Advanced Computing (JIAC)*, 1-7.
- Lyimo, J. M. (2023). Implementing a campus VoIP intercom VLAN: A technology review, system requirements and architecture. *International Journal of Science and Research Archive*, 716-726.

- Marcel, M. L. (2020). Integration of a Voice Over Internet Protocol (VoIP) Solution with Internet Protocol Version 6 (IPv6) in an Internet Protocol Version 4 (IPv4) Data Network to Increase Employee Productivity. *ICT-U Byrd Research Center*.
- Muniappan, A. D. (2024). Improvement of Packet Delivery Ratio in Wireless Mesh Network Using Hybrid Routing Protocol in VoIP Application. *International Journal of Intelligent Systems and Applications in Engineering*, 207-222.
- Muthoharo, M. &. (2022). Analysis Of Voice Quality on Aircraft Telephone Through Internet Telephony Gateway in Voice Over Internet Protocol. *Journal of Telecommunication Network* , 80-84.
- Naik, S. G. (2021). Codec-based Qos Analysis of Voip Over Conventional Network and Sdn. *International Journal of Innovations in Engineering Research and Technology*, 11-17.
- Nwaogwugwu, I. B. (2021). A Low-Cost Telephony System for Small/Medium Scale Businesses in Nigeria: An Implementation of VoIP on a LAN Using Mini SIP Server. *International Journal of Innovative Research & Development*.
- Prasetyo, J. A. (2020). Comparison of Voice over Internet Protocol (VoIP) Performances in Various Network Topologies. *Buletin Pos Dan Telekomunikasi*.
- Pratiwi, N. I. (2017). Penggunaan Media Video Call Dalam Teknologi Komunikasi. *Jurnal ilmiah dinamika sosial*, 202-224.
- Pribadi, A. (2023). VoIP (Voice over Internet Protocol) Based on Asterisk Using Trixbox at SMKN 1 Bangkinang. *International Journal Software Engineering and Computer Science (IJSECS)*, 65-70.
- Putra, D. P. (2021). Analisis Keamanan Voice Over Internet Protocol (Voip) Over Virtual Private Network (Vpn). *Jurnal Informatika Dan Rekayasa Perangkat Lunak*, 324-333.
- Putra, T. A. (2020). Implementasi Server Voip Menggunakan Asterisk Pada Jurusan Teknik Informatika Universitas Halu Oleo. *SemanTIK: Teknik Informasi*, 1-5.
- Putra, W. A. (2019). Perancangan Server VoIP Briker Pesantren Tahfidz Daarul Qur'ān Sebagai Media Komunikasi Dengan Protokol SIP. *Jurnal Teknik Informatika*, 108-115.
- Putro, R. E. (2022). Analisis Keamanan Komunikasi VoIP Server Portable Dilengkapi OpenVPN Menggunakan Linux Asterisk. *Jurnal Media Informatika Budidarma*, 943-951.
- Rendra, D. B. (2022). Rancang Bangun Sistem Voice Over Internet Protocol Pada Local Area Network Berbasis Software Mini Sip Server. *Jurnal Teknik AMATA*, 36-50.
- Roy, O. P. (2021). A survey on voice over internet protocol (VoIP) reliability research. *IOP Publishing*.

- Tricahya, S. W. (2023). Perancangan Voice Over Internet Protocol Menggunakan VPN Gateway-To-Gateway Berbasis Server. *Journal of Informatics and Computer Science (JINACS)*, 358-368.
- Youness, O. (2023). Secure and Efficient Authentication Scheme for VoIP Communications. *IJCI. International Journal of Computers and Information*, 73-81.
- Yunarti, W. &. (2018). Analisis dan Rancang Bangun Jaringan Komunikasi VOIP Dengan Server Briker Pada SMKN 1 Tanjung Jabung Timur. *Jurnal Manajemen Sistem Informasi*, 1272-1281.

CURRICULUM VITAE

Devi Megarita Br Sibarani

Mahato, 19 September 2002
 Jadi Mulya KM, Kelurahan Mahato, KecamaTan Tambusai Utara
 +62 823 8631 5479 | devimegaritasibarani@gmail.com



PENDIDIKAN :

Perguruan Tinggi

Universitas Sumatera Utara
Fakultas Ilmu Komputer dan Teknologi Informasi
Jurusan Ilmu Komputer
2020 - Sekarang
 IPK 3,67 / 4,00.

Sekolah Menengah Atas

SMA Swasta Katolik Cahaya Medan
2017 – 2020

KEMAMPUAN KOMPUTER

Pemrograman: C/C++, Pascal, HTML, CSS.
 Database : MySQL.
 IDE : Visual Studio Code, DevC++.
 OS : Windows.
 Software : Microsoft Excel, VMWare.

PROYEK

Math Game

2022

- Membuat semua desain untuk setiap game yang merupakan proyek dari Mobile Programming .

PENGALAMAN KERJA

Praktik Kerja Lapangan di PT. Bank SUMUT | Divisi Dana dan Jasa

Juni 2023 – Juli 2023

- Mengerjakan data nasabah dari sebuah website ke microsoft excel.
- Mengerjakan penginputan rekap data rekap sp.Rate prioritas dan korporasi.
- Membuat lampiran Program KEJAR mulai dari media publikasi, capaian dalam rangka Hari Indonesia Menabung, Satuan Pendidikan Implementasi serta Awards KEJAR 2023.

Pengabdian Desa Simempar

Juli 2023 – Agustus 2023

- Mengerjakan konten tentang Desa Simempar.
- Menjadi admin Twitter Desa Simempar.

PENGALAMAN ORGANISASI

Panitia Imilkom Sport Competition

2022

Anggota Dana dan Usaha

- Bendahara.

Panitia ILGANI

2022

Anggota konsumsi

- Mengurus bagian konsumsi selama acara berlangsung.

Panitia Penerimaan Mahasiswa Baru

2022

Ketua Panitia

- Mengkoordinasikan seluruh anggota dalam pencarian dana yaitu rutin melakukan jualan.

Panitia Kesatuan Mahasiswa Kristen

September 2022 – Maret 2023

Anggota Humsos

- Mengurus bagian upload flyer di instagram.