

Rancang Bangun Aplikasi Web Teleconference Two-way Communication Menggunakan IP Kamera Sebagai Alat Pembelajaran Daring

Tia Varisha¹, Husaini², Mursyidah³

^{1,3} Jurusan Teknologi Informasi dan Komputer Politeknik Negeri Lhokseumawe
Jln. B.Aceh Medan Km.280 Buketrata 24301 INDONESIA

¹varisha2802@gmail.com

²husaini@pnl.ac.id

³mursyidah@pnl.ac.id

Abstrak— Pembelajaran secara daring merupakan salah satu kebutuhan bagi institusi dan lembaga pendidikan saat ini. Pada kelas praktikum secara daring, biasanya pengajar memerlukan seseorang untuk mengontrol dan mengarahkan kamera agar pelajar tetap dapat menjangkau materi dari alat yang dipraktikkan. Pada penelitian ini akan dibangun sebuah aplikasi berbasis web sebagai alat *teleconference*, berbeda dengan *platform teleconference* yang sudah ada saat ini, aplikasi web ini dibuat dengan mengutamakan fitur yang dapat mengontrol arah atau gerak (*motion*) dari kamera secara *real-time* menggunakan IP kamera dan berkomunikasi dua arah melalui input-output audio secara *real-time*. Penelitian ini bertujuan mengetahui waktu respon komunikasi audio dan perbandingan performa fitur *streaming* video, dan gerak IP kamera ketika dijalankan pada kondisi *online* dan lokal menggunakan metode QoS (*Quality of Service*). Berdasarkan analisis data yang di peroleh dari hasil pengujian untuk komunikasi audio dikategorikan baik dengan rata-rata pengujian *throughput* memiliki nilai 2,578 kbps, *packet loss* senilai 0,291%, *delay* 9,295 ms, dan *jitter* 8,603 ms. Hasil pengujian untuk fitur *streaming* data video dan gerak IP kamera dengan dua percobaan pada masing-masing kondisi *local* dan *online* menunjukkan bahwa kondisi lokal cenderung memberikan kinerja yang lebih baik dalam *throughput*, *packet loss*, *delay*, dan *jitter* dibandingkan dengan kondisi *online*. Hal ini menunjukkan bahwa dalam pengujian ini, penggunaan IP kamera dalam aplikasi web telekonferensi mengalami peningkatan kualitas ketika beroperasi dalam jaringan lokal.

Kata kunci— *Teleconference*, IP kamera, *real-time*, *motion*

Abstract— *Online learning is one of the essential needs for educational institutions and organizations nowadays. In virtual practical classes, instructors typically require someone to control and direct the camera so that students can access the materials related to the practiced tools. This research aims to develop a web-based application as a teleconference tool for virtual teaching. Different from existing teleconference platforms, this web application is designed with a priority on features that enable real-time control of camera direction or motion using IP cameras, along with two-way communication through real-time audio input-output. The objective of this study is to determine the audio communication response time and compare the performance of video streaming and IP camera motion features when operated under online and local conditions using the Quality of Service (QoS) method. Based on the data analysis obtained from the test results, the audio communication is categorized as good, with an average throughput testing value of 2,578 kbps, a packet loss rate of 0.291%, a delay of 9.295 ms, and a jitter of 8.603 ms. The test results for video streaming and IP camera motion features, conducted in both local and online conditions, indicate that the local condition tends to provide better performance in terms of throughput, packet loss, delay, and jitter compared to the online condition. This suggests that in this test, the utilization of IP cameras in the web teleconference application experiences an improvement in quality when operating within a local network.*

Keywords— *Teleconference*, IP camera, *real-time*, *motion*

I. PENDAHULUAN

Peran komunikasi secara langsung dalam proses belajar mengajar sangat mempengaruhi tingkat pemahaman peserta didik terhadap materi yang disampaikan. Oleh karena itu, untuk memenuhi hal tersebut dibutuhkan media atau layanan yang dapat memberikan akses kemudahan dalam berinteraksi dan berkomunikasi dalam pembelajaran secara daring [1]. Salah satu bentuk pengembangan teknologi daring yang digunakan dalam dunia pendidikan saat ini adalah layanan *teleconference*. Teknologi *teleconference* digunakan dalam konteks pendidikan untuk memfasilitasi pembelajaran jarak jauh dengan menggunakan video dan audio untuk menghubungkan pengguna secara *real-time* meskipun tidak berada ditempat dan diwaktu yang sama [2].

Seiring dengan kebutuhan lembaga pendidikan akan layanan *teleconference* yang semakin meningkat setiap kurun waktu tertentu, maka diperlukan juga tingkat kualitas layanan. Berdasarkan penelitian atau layanan *teleconference* yang sudah ada, masih ditemukan kekurangan seperti mengharuskan instalasi aplikasi di setiap perangkat berbeda dan tidak multi *platform*, kemudian belum ada yang menggunakan IP kamera dengan fitur yang dapat menggerakkan kamera ke arah yang diinginkan sehingga membuat pengajar atau mentor harus mengarahkan kamera secara manual atau memerlukan operator untuk mengarahkannya. Khususnya ketika pembelajaran daring bagi kelas praktikum yang biasanya memerlukan pengajar atau mentor untuk menjelaskan alat-alat besar yang sulit dijangkau oleh kamera. Maka pada penelitian ini penulis akan

mengembangkan fitur untuk mengontrol gerak kamera ke arah yang diinginkan dengan menggunakan IP kamera sebagai alat *teleconference*.

Untuk menyelesaikan masalah penulis melakukan penelitian untuk memberikan solusi yaitu membuat sebuah aplikasi web *multi-platform* dengan pemanfaatan perangkat keras IP kamera dengan fitur *streaming* video secara *real-time*, fitur menggerakkan arah kamera dan komunikasi dua arah melalui audio.

Penelitian ini berkaitan dengan penelitian sebelumnya dengan judul “Rancangan Sistem Video *Conference e-Learning* di Universitas Harapan Medan”. Penelitian ini menghasilkan aplikasi *e-learning* yang merupakan web *based* dan mempunyai fitur *video conference* untuk memudahkan interaksi antara dosen dan mahasiswa dalam perkuliahan online. Dari hasil penelitian, *video conference* masih terbatas dalam jumlah *room* yang tersedia dan kamera yang digunakan hanya merupakan kamera perangkat dari *user* [1].

Penelitian ini berkaitan dengan penelitian sebelumnya dengan judul “Implementasi Video *Conference* dengan File *Sharing* menggunakan Web *RTC*” penelitian ini bertujuan untuk mempermudah akses komunikasi rapat atau konsultasi secara daring. penelitian ini menghasilkan aplikasi web dengan menerapkan metode *peer to peer* yang mempunyai fungsi utama yaitu fitur *video conference* dan file *sharing* yang berjalan di semua browser dan perangkat dengan memanfaatkan WebRTC sebagai protokol komunikasi data audio, video, dan file *sharing* [2].

Penelitian ini berkaitan dengan penelitian sebelumnya dengan judul “Sistem Konsultasi dan Bimbingan *Online* Berbasis Web Menggunakan Web *RTC* (Studi Kasus : Fakultas Teknik Universitas Suryakencana)” untuk mempermudah kegiatan bimbingan dan konsultasi antara mahasiswa dan dosen. Sistem konsultasi dan bimbingan *online* ini memiliki beberapa fitur seperti *live video stream* dan *live chatting*, agar mahasiswa dan dosen dapat berkomunikasi secara langsung dan juga dapat mengurangi waktu dalam jarak dari rumah ke kampus untuk bertemu dosen maupun mahasiswa. Dalam penelitian ini, aplikasi merupakan web *based* dan menggunakan WebRTC yang bertindak sebagai protokol dalam komunikasi data berupa audio dan video [3].

Penelitian ini berkaitan dengan penelitian sebelumnya dengan judul “Perancangan Sistem Monitoring kamera CCTV Menggunakan *Mobile Phone*”. Studi ini menghasilkan sebuah aplikasi berbasis *mobile phone* yang bertujuan untuk dapat memonitoring IP kamera CCTV melalui aplikasi *mobile* dengan fitur utama yaitu *streaming* video dari IP kamera CCTV. Monitoring IP kamera ini menggunakan sebuah protokol yaitu RTSP (*real time streaming protocol*) untuk *streaming* data video. Aplikasi ini menggunakan metode *cloud computing* untuk penyimpanan data dan belum menambahkan komunikasi data audio [4].

Pada penelitian ini akan dibangun sebuah aplikasi web *teleconference* dengan menggunakan IP Kamera yang bertujuan untuk memudahkan pengajar dalam mengajar secara daring, dengan menggunakan IP kamera pelajar dapat lebih mudah mengerti materi terutama dalam kelas praktikum karena aplikasi web ini menyediakan fitur untuk dapat mengontrol arah kamera dan berkomunikasi dua arah melalui audio. Aplikasi web ini menggunakan IP kamera V380 Pro yang sudah mendukung protokol RTSP (*real time streaming*

protocol) untuk *streaming* video dari IP kamera, protokol ONVIF untuk fitur menggerakkan arah IP kamera, dan untuk audio aplikasi web ini menggunakan WebRTC sebagai protokol.

II. METODOLOGI PENELITIAN

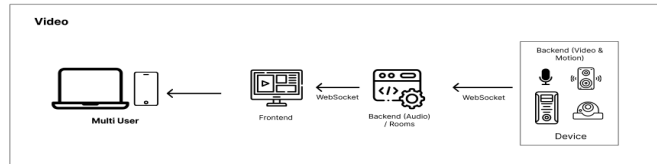
Pengumpulan data yang digunakan untuk penelitian rancang bangun aplikasi web *teleconference two-way communication* menggunakan IP kamera yaitu menggunakan jenis data sekunder yang meliputi jurnal penelitian dan publikasi penelitian. Teknik pengumpulan data yang digunakan dalam penelitian ini yaitu observasi yang merupakan suatu pengamatan dan pengujian secara langsung untuk mengumpulkan fakta, data, hingga nilai dari sistem yang dibangun.

A. Rancangan Sistem

Pada perancangan ini terdapat blok diagram sistem, perancangan konteks diagram dan user interface dari aplikasi web *teleconference* yang dibangun.

a. Rancangan sistem komunikasi data video

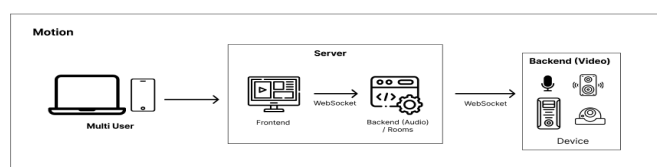
Sistem komunikasi data video hanya akan dilakukan satu arah yaitu dari IP kamera ke *user interface* web *teleconference* sebagai fitur *streaming* video, terlihat seperti pada gambar 1 berikut, data video yang didapatkan dari IP kamera V380 akan di-integrasi ke *frontend* audio yang sudah disalurkan dari server *local* ke internet menggunakan layanan ngrok agar dapat diakses oleh multi *user*. Komunikasi data video menggunakan RTSP (*real-time streaming protocol*) pada port 554 sebagai protokol.



Gambar 1. Rancangan sistem komunikasi data video

b. Rancangan komunikasi data *motion* (gerak)

Sama halnya dengan sistem komunikasi data video yang hanya satu arah, sistem komunikasi data *motion* atau gerak ini juga hanya satu arah. Namun perbedaannya adalah komunikasi *motion* hanya satu arah dari *user* ke IP kamera dengan *request* dari fitur *motion* yang ada pada *user interface* web *teleconference*. Komunikasi *motion* ini menggunakan protokol onvif untuk menggerakkan IP kamera. Gambar 2 berikut merupakan rancangan sistem komunikasi data *motion*.



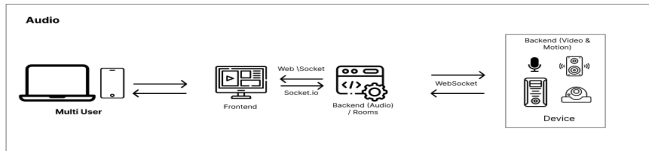
Gambar 2. Rancangan sistem komunikasi data *motion*

c. Rancangan sistem komunikasi data audio

Komunikasi audio menggunakan protokol WebRTC, Web *RTC* mengizinkan penjelajah *web* dapat melakukan pertukaran data aplikasi dan juga melakukan performa *teleconferencing*

suara/video secara *per-to-peer*, tanpa melakukan penginstalan *plug-in* atau perangkat lunak pihak ketiga [5].

Komunikasi audio disini tidak akan berhubungan dengan IP kamera, karena hanya akan menggunakan perangkat audio eksternal dari komputer. Bentuk dari sistem komunikasi audio yaitu sebuah *room teleconference* yang dapat digunakan untuk komunikasi audio dua arah antar user sehingga untuk *backend* video juga akan dianggap sebagai *user*. Rancangan sistem komunikasi audio dapat dilihat pada gambar 3 berikut.



Gambar 3. Rancangan sistem komunikasi data audio

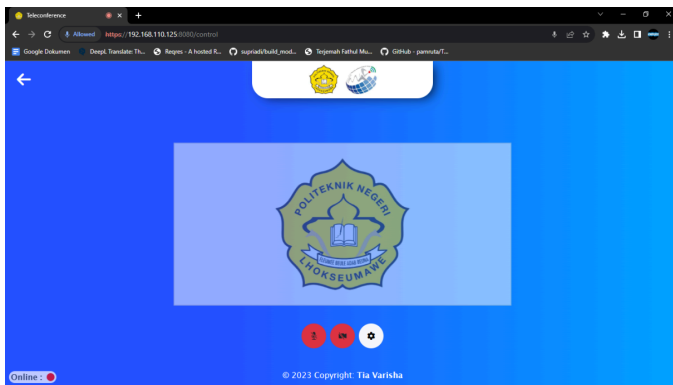
III. HASIL DAN PEMBAHASAN

A. Implementasi *User Interface*

Tahapan ini merupakan tahapan yang berisi tentang penjelasan tampilan *user interface* dari aplikasi web *teleconference*. Adapun tampilan *user interface* dari aplikasi web *teleconference* sebagai berikut

a. Halaman Kontrol

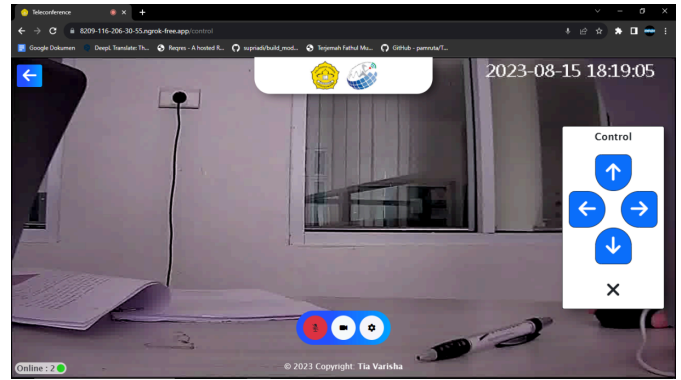
Halaman kontrol merupakan halaman yang memuat fitur-fitur yang disediakan oleh aplikasi web *teleconference*, yaitu tombol fitur video dan audio, dan tombol kontrol, tampilannya dapat dilihat pada gambar 3 berikut.



Gambar 3. Tampilan halaman kontrol pada tampilan web

Berikut merupakan gambar halaman kontrol dari aplikasi web *teleconference* jika fitur diaktifkan, terlihat bahwa jika sudah mengaktifkan fitur video maka akan video memenuhi layar yang sebelumnya hanya terdapat logo. Ketika tombol kontrol aplikasi web *teleconference* ditekan, terlihat muncul sebuah *pop-out* dengan empat tombol atas, kanan, kiri, bawah untuk mengontrol gerak atau arah dari kamera.

Fitur audio yang dapat diaktifkan untuk dapat komunikasi dua arah dengan sesama pengguna. Pada hasil video *streaming* akan terlihat juga keterangan untuk tanggal dan waktu secara *real-time*. Pada bagian pojok kiri bawah juga terdapat keterangan bahwa berapa pengguna yang sedang *online* atau menjalankan fitur yang ada dalam web *teleconference*.



Gambar 4. Tampilan halaman kontrol fitur arah kamera

B. Hasil Pengujian QoS *Streaming* Data Video dan Gerak IP Kamera dalam Kondisi *Online* dan Lokal

Pada tahap ini dilakukan pengujian menggunakan *Quality of Service* pada web *teleconference two-way communication* menggunakan IP kamera untuk percobaan *streaming* data video dan fitur gerak IP kamera yang dilakukan pada sisi pengguna sebanyak dua percobaan pada masing-masing kondisi *online* dan *local* menggunakan tiga *user*.

1. Pengujian *Throughput*

Pengujian *throughput* dilakukan dengan tujuan untuk mengidentifikasi efisiensi kecepatan pengiriman data pada sisi pengguna aplikasi web [5].

Berikut merupakan salah satu contoh perhitungan pada salah satu pengujian menggunakan persamaan *troughput* untuk mendapatkan hasil *troughput* dari hasil *packet* data seperti pada tabel berikut yang dihitung menggunakan *tools* wireshark.

TABEL I
HASIL DATA *TROUGHPUT* PADA WIRESHARK

Measurement	Captured	Displayed
Packets	24477	24477 (100.0%)
Time span, s	187,641	187,641
Average pps	130,4	130,4
Average packet size, B	697	697
Bytes	17050808	17050808 (100.0%)
Average bytes/s	90 k	90k

<i>Throughput</i>	=	Packet data yang diterima / Lama pengamatan
	=	17050808 bytes / 187,641 s
	=	91.180 bytes * 8
	=	729.446 ps

a. Hasil pengujian *throughput* pada kondisi *online*

Pada tabel berikut merupakan hasil dari pengujian *throughput* untuk percobaan *streaming* data video dan fitur gerak IP kamera pada kondisi *online*.

TABEL II
HASIL PENGUJIAN *TROUGHPUT* PADA KONDISI *ONLINE*

Percobaan	Pengguna	Perhitungan <i>Throughput</i> (kbps)	Rata-rata <i>Throughput</i> (kbps)
Percobaan ke 1	Pengguna 1	729	729
	Pengguna 2	655	
	Pengguna 3	805	

Percobaan ke 2	Pengguna 1	352	812
	Pengguna 2	720	
	Pengguna 3	1.366	
Rata-rata		770,5 kbps	

b. Hasil pengujian *throughput* pada kondisi *local*

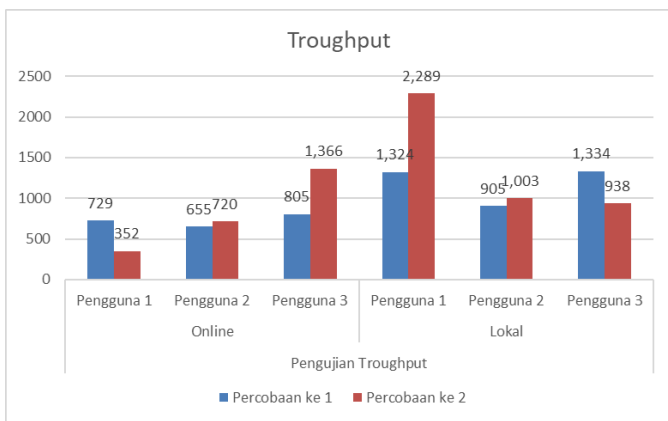
Pada tabel berikut merupakan hasil dari pengujian *throughput* pada percobaan *streaming* data video dan fitur gerak IP kamera dalam kondisi *local*.

TABEL III
HASIL PENGUJIAN *THROUGHPUT* PADA KONDISI *LOCAL*

Percobaan	Pengguna	Perhitungan <i>Throughput</i> (kbps)	Rata-rata <i>Throughput</i> (kbps)
Percobaan ke 1	Pengguna 1	1.324	3.563
	Pengguna 2	905	
	Pengguna 3	1.334	
Percobaan ke 2	Pengguna 1	2.289	1.410
	Pengguna 2	1.003	
	Pengguna 3	938	
Rata-rata		2.486 Kbps	

Dalam pengujian *throughput* pada tabel diatas, dilakukan pengiriman aliran data dari IP kamera ke aplikasi web *teleconference* melalui jaringan yang telah disiapkan. Jumlah data yang berhasil ditransmisikan selama periode waktu tiga menit menunjukkan bahwa rata-rata data yang berhasil dikirim mencapai 770,5 kbps dalam kondisi *online* dan 2.486 kbps pada pada kondisi *local*.

Hasil ini mengindikasikan bahwa jaringan mampu mentransmisikan data rata-rata sebesar 770,5 kb per detik dari IP kamera ke aplikasi web pada kondisi *online* yang dikategorikan sedang, dan 2.486 kb per detik pada pada kondisi *local* yang dikategorikan sangat baik berdasarkan standar *throughput*. Dari hasil tersebut dapat disimpulkan bahwa *throughput* pada kondisi *local* lebih baik dari *throughput* pada kondisi *online*. Gambar 5 berikut merupakan hasil perbandingan pengujian *throughput* dalam bentuk grafik untuk masing-masing pengguna pada dua percobaan.



Gambar 5. Grafik perbandingan hasil pengujian *throughput*

Hasil ini mengindikasikan bahwa jaringan mampu mentransmisikan data rata-rata sebesar 770,5 kb per detik dari IP kamera ke aplikasi web pada kondisi *online* yang dikategorikan sedang, dan 2.486 kb per detik pada pada kondisi *local* yang dikategorikan sangat baik berdasarkan standar *throughput*. Dari hasil tersebut dapat disimpulkan

bahwa *throughput* pada kondisi *local* lebih baik dari *throughput* pada kondisi *online*.

2. Pengujian *Packet loss*

Pengujian *packet loss* merupakan upaya untuk mendapatkan di mana *packet* data hilang dalam proses pengiriman akibat berbagai faktor seperti tabrakan (*collision*) atau kemacetan (*congestion*) dalam jaringan.

Berikut merupakan salah satu contoh implementasi perhitungan menggunakan persamaan *packet loss* untuk mendapatkan hasil dari hasil *packet* data seperti pada tabel berikut yang dihitung menggunakan *tools* wireshark.

TABEL IV
HASIL DATA *PACKET LOSS* PADA WIRESHARK

Measurement	Captured	Displayed
Packets	24477	3 (0.0%)
Time span, s	187,641	-
Average pps	130,4	-
Average packet size, B	697	1419
Bytes	17050808	4258 (0.0%)
Average bytes/s	90 k	-

$$Packet Loss = \frac{Packet Transmitted - Paket Received}{Packet Transmitted} \times 100\%$$

$$Packet Loss = \frac{24477 - 24474}{24477} \times 100\%$$

$$Packet Loss = \frac{3}{24477} \times 100\%$$

$$Packet Loss = 1,225640397107489e - 4 * 100\%$$

$$Packet Loss = 0,0122564039710749\%$$

a. Hasil pengujian *Packet loss* pada kondisi *online*

Tabel berikut merupakan hasil dari pengujian *packet loss* pada percobaan *streaming* data video dan fitur gerak IP kamera dalam kondisi *online*.

TABEL V
HASIL PENGUJIAN *PACKET LOSS* PADA KONDISI *ONLINE*

Percobaan	Pengguna	Perhitungan <i>Packet loss</i> (%)	Rata-rata <i>Packet loss</i> (%)
Percobaan ke 1	Pengguna 1	0,012	0,491
	Pengguna 2	0,264	
	Pengguna 3	0,215	
Percobaan ke 2	Pengguna 1	0,885	0,420
	Pengguna 2	0,000	
	Pengguna 3	0,376	
Rata-rata		0,455 %	

b. Hasil pengujian *packet loss* pada kondisi *local*

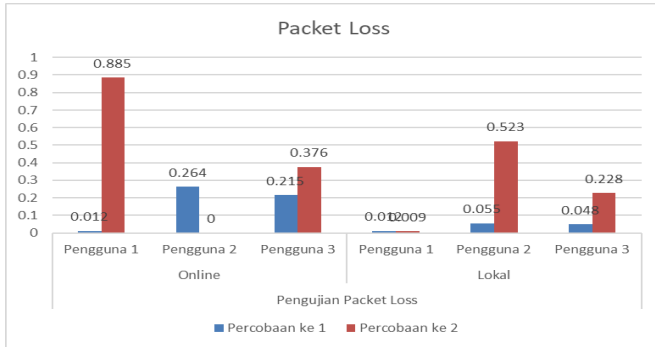
Pada tabel berikut merupakan hasil dari pengujian *packet loss* untuk percobaan *streaming* data video dan fitur gerak IP kamera dalam kondisi *local*.

TABEL VI
HASIL PENGUJIAN *PACKET LOSS* PADA KONDISI *LOCAL*

Percobaan	Pengguna	Perhitungan <i>Packet loss</i> (%)	Rata-rata <i>Packet loss</i> (%)
Percobaan ke 1	Pengguna 1	0,012	0,038
	Pengguna 2	0,055	
	Pengguna 3	0,048	
Percobaan ke 2	Pengguna 1	0,009	0,760

Pengguna 2	0,523
Pengguna 3	0,228
Rata-rata	0,399 %

Pengujian *packet loss* pada tabel diatas, dihasilkan dari pengiriman aliran data dari IP kamera ke aplikasi web *teleconference* melalui jaringan yang telah disiapkan dengan jumlah *packet* data yang dikirimkan dan jumlah *packet* data yang diterima di tujuan. Gambar 6 berikut merupakan hasil perbandingan pengujian *packet loss* dalam bentuk grafik.



Gambar 6. Grafik perbandingan hasil pengujian *packet loss*

Hasil pengujian pengujian mengindikasikan bahwa terdapat nilai rata-rata *packet loss* sebesar 0,455 % selama pengujian dengan kondisi *online* dan 0,399 % pada kondisi *local* yang dikategorikan sangat baik menurut standar *packet loss*. Persentase ini menunjukkan bahwa besar proporsi *packet* data yang hilang selama proses transmisi dari IP kamera ke layanan telekonferensi pada kondisi *online* lebih besar dari pada ketika kondisi lokal.

3. Pengujian Delay

Pengujian *delay* dilakukan pada sisi pengguna aplikasi web untuk mengetahui waktu yang diperlukan oleh data untuk melakukan perjalanan dari sumber awal hingga sumber tujuan, jika nilai *delay* rendah, hal ini menandakan bahwa jaringan dan aplikasi web efisien dalam mengirimkan data.

Berikut merupakan salah satu contoh implementasi perhitungan menggunakan persamaan *delay* untuk mendapatkan hasil *delay* dari hasil *packet* data seperti pada tabel yang dihitung menggunakan *tools* wireshark.

TABEL VII
HASIL DATA DELAY PADA WIRESHARK

Measurement	Captured	Displayed
Packets	24477	3 (0.0%)
Time span, s	187,641	-
Average pps	130,4	-
Average packet size, B	697	1419
Bytes	17050808	4258 (0.0%)
Average bytes/s	90 k	-

Rata-rata <i>delay</i>	=	Total <i>delay</i> / Total <i>packet</i> diterima
	=	187,641186 / 24474
	=	0,00766696028438343
	=	7,66696028438343

a. Hasil pengujian *delay* pada kondisi *online*

Pada tabel berikut merupakan hasil dari pengujian *delay* pada aplikasi web *teleconference* untuk percobaan *streaming* data video dan fitur gerak IP kamera pada kondisi *online*.

TABEL VIII
HASIL PENGUJIAN DELAY PADA KONDISI ONLINE

Percobaan	Pengguna	Perhitungan <i>Delay</i> (ms)	Rata-rata <i>Delay</i> (ms)
Percobaan ke 1	Pengguna 1	7,666	5,245
	Pengguna 2	4,269	
	Pengguna 3	3,801	
Percobaan ke 2	Pengguna 1	19,582	9,763
	Pengguna 2	4,282	
	Pengguna 3	5,426	
Rata-rata			7,504 ms

b. Hasil pengujian *delay* pada kondisi *local*

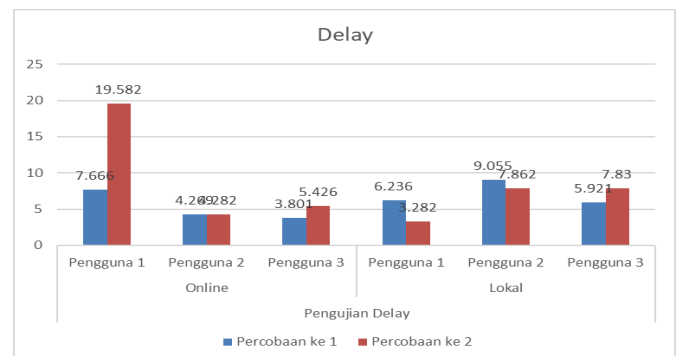
Pada tabel berikut merupakan hasil dari pengujian *delay* untuk percobaan *streaming* data video dan fitur gerak IP kamera pada kondisi *local*.

TABEL IX
HASIL PENGUJIAN DELAY PADA KONDISI LOCAL

Percobaan	Pengguna	Perhitungan <i>Delay</i> (ms)	Rata-rata <i>Delay</i> (ms)
Percobaan ke 1	Pengguna 1	6,236	7,070
	Pengguna 2	9,055	
	Pengguna 3	5,921	
Percobaan ke 2	Pengguna 1	3,282	6,324
	Pengguna 2	7,862	
	Pengguna 3	7,830	
Rata-rata			6,697 ms

Hasil pengujian pada tabel diatas menunjukkan bahwa setelah masing-masing kondisi yaitu *online* dan *local* melakukan dua percobaan dengan 3 pengguna di dapatkan nilai rata-rata *delay* pada setiap percobaan, yaitu pada kondisi *online* dengan nilai rata-rata 7,504 ms, dan pada percobaan kondisi *local* dengan nilai rata-rata 6,697 ms.

Berdasarkan pengujian dan nilai yang sudah didapatkan, masing-masing kondisi pada pengujian dikategorikan sangat baik menurut standar, namun nilai *delay* yang lebih kecil terjadi pada percobaan dengan kondisi *local* dengan selisih 0,807 ms yang berarti hasil pengujian *delay* didapati lebih baik ketika kondisi *local* dibanding dengan kondisi *online*. Gambar 7 berikut merupakan hasil perbandingan pengujian *delay* dalam bentuk grafik dari masing-masing pengguna pada dua percobaan.



Gambar 7. Grafik perbandingan hasil pengujian *packet loss*

4. Pengujian Jitter

Jitter mengacu pada variasi waktu yang terjadi dalam pengiriman *packet* data antara sumber dan tujuan. Dalam

telekonferensi melalui aplikasi web, *jitter* dapat mempengaruhi pengalaman pengguna.

Pada pengujian ini, dilakukan pengukuran *jitter* untuk dua scenario yaitu *local* dan *online* dengan masing-masing dua kali percobaan menggunakan 3 pengguna guna menilai sejauh mana fluktuasi waktu tiba data mempengaruhi kualitas layanan. Berikut adalah hasil pengujian untuk dua skenario yang berbeda. Berikut merupakan salah satu contoh implementasi perhitungan menggunakan persamaan *jitter* mendapatkan hasil dari hasil *packet* data seperti pada tabel berikut yang dihitung menggunakan *tools* wireshark.

TABEL X
HASIL *PACKET* DATA PERHITUNGAN *JITTER* PADA WIRESHARK

Measurement	Captured	Displayed
Packets	24477	3 (0.0%)
Time span, s	187,641	-
Average pps	130,4	-
Average packet size, B	697	1419
Bytes	17050808	4258 (0.0%)
Average bytes/s	90 k	-

<i>Jitter</i>	=	Total variasi <i>delay</i> / Total <i>packet</i> diterima - 1
	=	187,556674 / 24473 - 1
	=	0,007663507
	=	7,66350715

a. Hasil pengujian *jitter* pada kondisi *online*

Pada tabel berikut merupakan hasil dari pengujian *jitter* pada aplikasi web *teleconference* pada kondisi *online*.

TABEL XI
HASIL PENGUJIAN *JITTER* PADA KONDISI *ONLINE*

Percobaan	Pengguna	Perhitungan <i>Jitter</i> (ms)	Rata-rata <i>Jitter</i> (ms)
Percobaan ke 1	Pengguna 1	7,663	5,244
	Pengguna 2	4,270	
	Pengguna 3	3,801	
Percobaan ke 2	Pengguna 1	19,572	9,761
	Pengguna 2	4,282	
	Pengguna 3	5,429	
Rata-rata			7,504 ms

b. Hasil pengujian *jitter* pada kondisi *local*

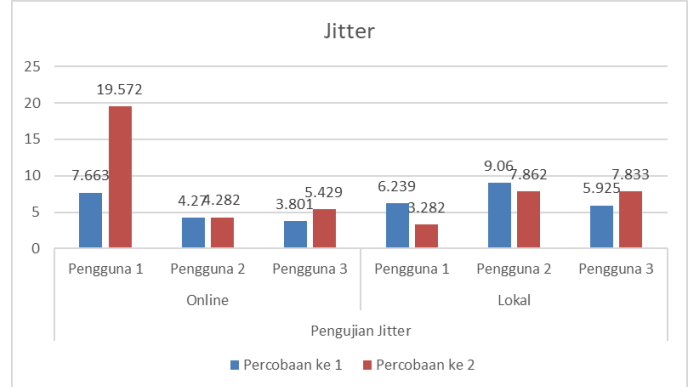
Pada tabel berikut merupakan hasil dari pengujian *jitter* untuk percobaan *streaming* data video dan fitur gerak IP kamera dalam kondisi *local*.

TABEL XII
HASIL PENGUJIAN *JITTER* PADA KONDISI *LOCAL*

Percobaan	Pengguna	Perhitungan <i>Jitter</i> (ms)	Rata-rata <i>Jitter</i> (ms)
Percobaan ke 1	Pengguna 1	6,239	7,074
	Pengguna 2	9,060	
	Pengguna 3	5,925	
Percobaan ke 2	Pengguna 1	3,282	6,325
	Pengguna 2	7,862	
	Pengguna 3	7,833	
Rata-rata			6,699 ms

Hasil pengujian pada tabel diatas menunjukkan bahwa setelah masing-masing kondisi yaitu *online* dan *local* melakukan dua percobaan dengan 3 pengguna didapatkan nilai rata-rata *jitter* pada setiap percobaan, yaitu pada kondisi *online* dengan nilai rata-rata 7,504 ms, dan pada percobaan kondisi *local* dengan nilai rata-rata 6,699 ms.

Berdasarkan nilai pengujian yang sudah didapatkan, masing-masing kondisi pada pengujian dikategorikan baik menurut standar *jitter*, namun nilai rata-rata *jitter* yang lebih kecil terjadi pada percobaan dengan kondisi *local* dengan selisih 0,505 ms yang berarti hasil pengujian *jitter* didapati lebih baik ketika kondisi *local* dibanding dengan kondisi *online*. Gambar berikut merupakan hasil perbandingan pengujian *jitter* dalam bentuk grafik.



Gambar 8. Grafik perbandingan hasil pengujian *jitter*

C. Hasil Pengujian QoS Pada Fitur Komunikasi Audio

Pada tahap ini dilakukan pengujian menggunakan *Quality of Service* pada web *teleconference two-way communication* menggunakan IP kamera untuk percobaan fitur komunikasi audio dengan empat parameter yaitu *throughput*, *packet loss*, *delay* dan *jitter* yang dilakukan sebanyak dua percobaan dengan masing-masing menggunakan tiga *user*.

1. Hasil Pengujian *Throughput* fitur komunikasi audio

Throughput mengacu pada jumlah data audio yang dapat dikirim atau diterima melalui jaringan. Pengujian ini melibatkan pengiriman sejumlah besar data audio melalui jaringan dan mencatat berapa banyak data yang berhasil ditransfer, dan diukur dalam satuan kilobits per detik (kbps).

Tabel berikut merupakan hasil dari pengujian *throughput* fitur komunikasi audio pada aplikasi web *teleconference*.

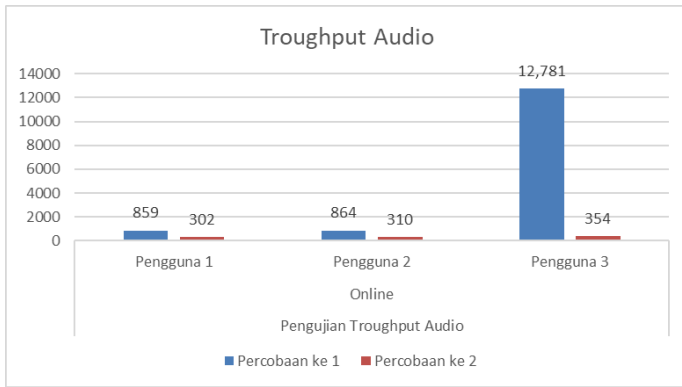
TABEL XIII
HASIL PENGUJIAN *THROUGHPUT* FITUR KOMUNIKASI AUDIO

Percobaan	Pengguna	Perhitungan <i>Throughput</i> (kbps)	Rata-rata <i>Throughput</i> (kbps)
Percobaan ke 1	Pengguna 1	859	4.834
	Pengguna 2	864	
	Pengguna 3	12.780	
Percobaan ke 2	Pengguna 1	302	322
	Pengguna 2	310	
	Pengguna 3	354	
Rata-rata			2.578 kbps

Dalam pengujian *throughput* pada tabel diatas, dilakukan pengiriman aliran data dari pengguna ke pengguna lainnya pada aplikasi web *teleconference* melalui jaringan yang telah disiapkan. Jumlah data yang berhasil ditransmisikan selama periode waktu tiga menit menunjukkan bahwa rata-rata data yang berhasil dikirim mencapai 2.578 kbps.

Hasil ini mengindikasikan bahwa jaringan mampu mentransmisikan rata-rata data sebesar 2.578 kb per detik dengan sesama pengguna aplikasi web yang dikategorikan sangat baik berdasarkan standar *throughput*. Dari hasil tersebut dapat disimpulkan bahwa sistem komunikasi mampu

menangani beban data yang dihasilkan oleh percakapan audio secara dua arah dalam web *teleconference*. Gambar 9 berikut merupakan hasil pengujian *throughput* dalam grafik.



Gambar 9. Grafik hasil pengujian *throughput*

2. Pengujian *Packet loss* fitur komunikasi audio

Pengujian *packet loss* pada komunikasi audio dalam web telekonferensi bertujuan untuk mengidentifikasi sejauh mana kehilangan *packet* data yang terjadi selama transmisi audio melalui jaringan. Dalam pengujian ini, sejumlah data audio dikirimkan melalui jaringan yang digunakan untuk telekonferensi, dan kemudian dilacak apakah semua *packet* data tiba dengan sukses di tujuan.

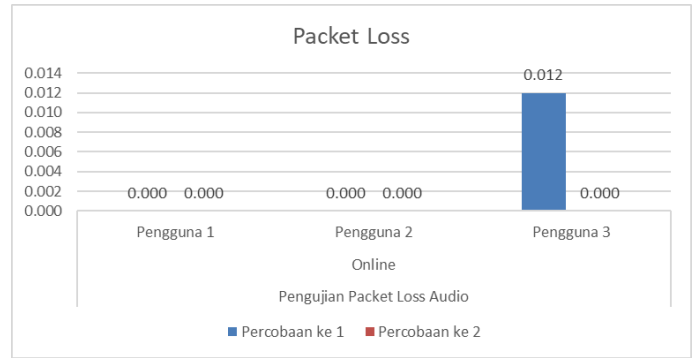
Tabel berikut merupakan hasil dari pengujian *Packet loss* pada fitur komunikasi audio pada aplikasi web *teleconference*.

TABEL XIV
HASIL PENGUJIAN *PACKET LOSS* FITUR KOMUNIKASI AUDIO

Percobaan	Pengguna	Perhitungan <i>Packet loss</i> (%)	Rata-rata <i>Packet loss</i> (%)
Percobaan ke 1	Pengguna 1	0,000	0,004
	Pengguna 2	0,000	
	Pengguna 3	0,012	
Percobaan ke 2	Pengguna 1	0,000	0,000
	Pengguna 2	0,000	
	Pengguna 3	0,000	
Rata-rata			0,002 %

Pengujian *packet loss* pada tabel diatas, dihasilkan dari pengiriman aliran data dari sesama pengguna layanan telekonferensi melalui jaringan yang telah disiapkan dengan jumlah *packet* data yang dikirimkan dan jumlah *packet* data yang diterima di tujuan, di mana suara *real-time* dikirimkan antar pengguna yang terhubung melalui internet.

Gambar 10 berikut merupakan hasil pengujian *packet loss* dalam bentuk grafik untuk masing-masing percobaan dan pengguna.



Gambar 10. Grafik hasil pengujian *throughput*

Hasil pengujian mengindikasikan bahwa terdapat rata-rata *packet loss* sebesar 0,002 % selama pengujian dengan kondisi *online*. Persentase ini menunjukkan bahwa besar proporsi *packet* data yang hilang selama proses transmisi antar pengguna pada layanan telekonferensi sangat kecil dan dikategorikan sangat baik.

3. Pengujian *Delay* fitur komunikasi audio

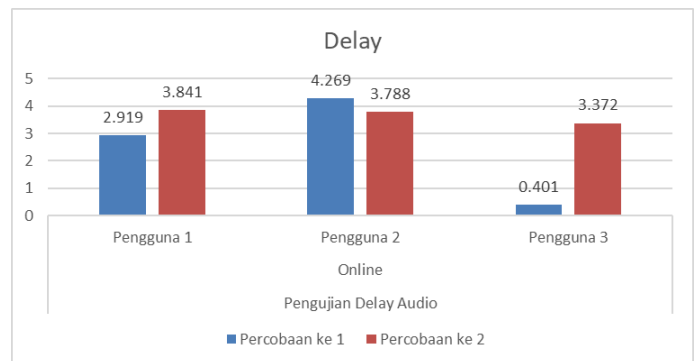
Pengujian ini dilakukan untuk mengevaluasi sejauh mana keterlambatan atau *delay* yang terjadi dalam transmisi audio *real-time*.

Tabel berikut merupakan hasil dari pengujian *delay* untuk percobaan komunikasi audio pada aplikasi web *teleconference*.

TABEL XV
HASIL PENGUJIAN *DELAY* PADA FITUR KOMUNIKASI AUDIO

Percobaan	Pengguna	Perhitungan <i>Delay</i> (ms)	Rata-rata <i>Delay</i> (ms)
Percobaan ke 1	Pengguna 1	2,919	7,589
	Pengguna 2	4,269	
	Pengguna 3	0,401	
Percobaan ke 2	Pengguna 1	3,841	7,589
	Pengguna 2	3,788	
	Pengguna 3	3,372	
Rata-rata			9,295 ms

Hasil pengujian pada tabel diatas menunjukkan bahwa setelah masing-masing melakukan dua percobaan dengan 3 pengguna di dapatkan nilai rata-rata *delay* pada setiap percobaan, yaitu dengan nilai rata-rata 9,295 ms. Gambar 13 berikut merupakan hasil pengujian *delay* dalam bentuk grafik untuk masing-masing percobaan dan pengguna.



Gambar 11. Grafik hasil pengujian *delay*

Berdasarkan rata-rata nilai yang sudah didapatkan dari pengujian, *delay* dikategorikan sangat baik menurut standart dan tidak ada gangguan selama komunikasi audio secara *real-time* berlangsung.

4. Hasil Pengujian *Jitter* fitur komunikasi audio

Pengujian *jitter* pada komunikasi audio dalam aplikasi web telekonferensi merupakan proses yang mengukur variasi waktu kedatangan *packet-packet* data audio di tujuan akhir selama transmisi melalui jaringan.

Pada pengujian ini, dilakukan pengukuran *jitter* dengan masing-masing dua kali percobaan menggunakan 3 pengguna guna menilai sejauh mana fluktuasi waktu tiba data mempengaruhi kualitas layanan. Pada tabel berikut merupakan hasil dari pengujian *jitter* untuk fitur komunikasi audio pada aplikasi web *teleconference*.

TABEL XVI

HASIL PENGUJIAN *JITTER* PADA FITUR KOMUNIKASI AUDIO

Percobaan	Pengguna	Perhitungan <i>Jitter</i> (ms)	Rata-rata <i>Jitter</i> (ms)
Percobaan ke 1	Pengguna 1	2,919	6,206
	Pengguna 2	2,886	
	Pengguna 3	0,401	
Percobaan ke 2	Pengguna 1	3,841	11,001
	Pengguna 2	3,788	
	Pengguna 3	3,372	
Rata-rata			8,603 ms

Hasil pengujian pada tabel diatas menunjukkan bahwa setela melakukan dua percobaan dengan 3 pengguna didapatkan nilai rata-rata *jitter* pada setiap percobaan, yaitu dengan nilai rata-rata 8,603 ms. Dari nilai yang sudah didapatkan, masing-masing kondisi pada pengujian dikategorikan baik menurut standar *jitter* dimana komunikasi audio sudah dapat menghasilkan percakapan audio yang lancar.

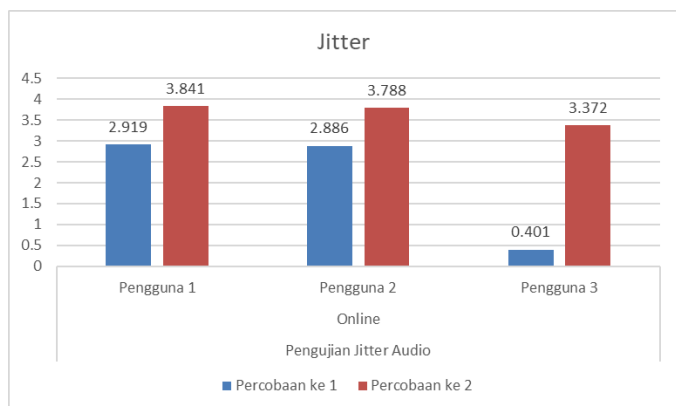
Dengan kurangnya variasi waktu tiba *packet* data audio di tujuan, suara yang diterima oleh peserta sudah lebih konsisten dan bebas dari gangguan seperti suara terputus-putus atau jeda yang mengganggu. Gambar 12 berikut merupakan hasil pengujian *packet loss* dalam bentuk grafik untuk masing-masing percobaan dan pengguna.

IV. KESIMPULAN

Berdasarkan hasil penelitian yang dilakukan, dapat disimpulkan bahwa hasil pengujian *streaming* data video dan fitur gerak IP kamera, didapati kondisi lokal lebih unggul dengan kategori sangat memuaskan menurut standar QoS, dimana rata-rata pengujian *throughput* memiliki selisih 1.715,5 kbps lebih tinggi pada kondisi lokal, pengujian *packet loss* memiliki selisih 0,056% lebih tinggi pada kondisi *online*, pengujian *delay* memiliki selisih 0,807 ms lebih tinggi pada kondisi *online* dan pengujian *jitter* memiliki selisih 0,805 ms lebih tinggi pada kondisi *online*. Berdasarkan hasil pengujian fitur komunikasi audio didapat bahwa performa komunikasi audio mencapai kategori kualitas yang baik, dimana rata-rata pengujian *throughput* memiliki nilai 2.578 kbps, *packet loss* senilai 0,291%, *delay* 9,295 ms, dan *jitter* 8,603 ms.

REFERENSI

- [1] Anggraini, N., & Syahputra, E. R. (2019). Rancangan sistem video conference *e-learning* di Universitas Harapan Medan. *Jurnal Penelitian Teknik Informatika Universitas Prima Indonesia (UNPRI) Medan* , 467-473.
- [2] Azzam, F. N., Kartikasari, D. P., & Bakhtiar, F. A. (2019). Implementasi Video Conference dengan *file sharing* menggunakan WebRTC. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer* , 10102-10109.
- [3] Abdulghani, T., & Gozali, M. M. (2019). Sistem konsultasi dan bimbingan *Online* berbasis Web menggunakan WebRTC (Studi kasus: Fakultas Universitas Suryakencana). *Media Jurnal Informatika* , 2477-2542
- [4] Jumiran, & Fitri, A. (2020). Perancangan Sistem Monitoring kamera CCTV Menggunakan Mobile Phone. *Jurnal IPSIKOM* , 2338-4093.
- [5] Pelco. (2023). *What is ONVIF, a complete guide to ONVIF cameras and profile*. Diakses 11 Juli 2023, dari Pelco: www.pelco.com/blog/onvif-guide
- [6] Rachmawati, Munawar, & Tari, S. (2023). Analisis *quality of service* (QoS) radio streaming menggunakan icecast pada Wi-Fi 802.11n. *Jurnal Listrik telekomunikasi elektronika* , 18-24.



Gambar 12. Grafik hasil pengujian *jitter*