KNOWBASE

International Journal of Knowledge in Database

Edition July - December 2021

Application of Dropshipping Method in E-Commerce
to Create New Business Opportunities for Students
Dinda Syufradian Putra, Zulfikar

Design of New Student Registration Information System
at MTsN 4 Kab. Lima Puluh Kota Android Based Cities
Yulia Elsa Putri, Sarwo Derta

Application for Calculation of Islamic Sharia Inheritance
Based on Android for Mawaris Fiqh Courses
Artesia Anwar

Matrix Integration with Jitsi Conference Server for Online Learning
*Rini Widyastuti, Karmila Suryani, Ade Fitri Rahmadani,
Triadmoko Denny Fatrosa, Wandi Syahindra*

Design and Build a New Student Registration Information System
Supported by SMS Gateway at Alhira Middle School
Yosdarso Afero

Application of the Fuzzy Sugeno Method in a Decision Support System
for Teacher Performance Assessment
Muhammad Rezeki, Nursaka Putra

Heavy Equipment Rental Schedule Report Application
Using Visual Basic.Net and MySQL Database
Syerlie Annisa, Yurita Delvianti

Implementation of the Brute Force Algorithm
in the Design of Android-Based Thesis Catalogs
Rahmi Tirajana Hasibuan, Supratman Zakir

Reviewers

Taufik Sutanto, (Scopus ID : 56154729600), Universitas Islam Negeri Syarif Hidayatullah Jakarta, Indonesia

Mr. Suendri Suendri, (Scopus Author ID: 57215914818), Universitas Islam Negeri (UIN) Sumatera Utara, Indonesia

Zainal Arifin Hasibuan, (Scopus Author ID: 57203922071), Universitas Indonesia, Indonesia

Anitawati Mohd Lokman, (Scopus Author ID: 56962751100), Universiti Teknologi MARA (UiTM), Malaysia

Fajril Akbar, (Scopus ID : 56029641300), Universitas Andalas, West Sumatra, Indonesia

Amir Rizaan, (Scopus Author ID: 57194241656), Universiti Putra Malaysia, Malaysia

Ikhwan Arief, (Scopus Author ID: 57209453335), Universitas Andalas, West Sumatra, Indonesia

Handaru Jati, (Scopus Author ID: 25825016500), Universitas Negeri Yogyakarta, Indonesia

Budi Rahmadya, (Scopus Author ID: 57211640582), Universitas Andalas, Indonesia

Andri Pranolo, (Scopus Author ID: 56572821900), Universitas Ahmad Dahlan, Indonesia

Imam Marzuki Shofi, (Scopus Author ID: 55580293800, h-index=3), Universitas Islam Negeri Syarif Hidayatullah Jakarta, Indonesia

Sarjon Defit, (Scopus Author ID: 35758511600), Universitas Putra Indonesia "YPTK" Padang, Indonesia

Jufriadif Na'am, (Scopus Author ID: 57189371499), Universitas Putra Indonesia "YPTK" Padang, Indonesia

Saiful Akbar, (Scopus Author ID: 57193715510), Institut Teknologi Bandung, Indonesia

Rizal Munadi, (Scopus Author ID: 16022345800), Universitas Syiah Kuala, Indonesia

Rila Mandala, (Scopus Author ID: 6506600409), Institut Teknologi Bandung, Indonesia

Fahren Bukhari, (Scopus Author ID: 57201923147), Institut Pertanian Bogor (IPB University), Indonesia

Febi Yanto, (Scopus Author ID: 57204963122), Universitas Islam Negeri (UIN) Sultan Syarif Kasim Riau, Indonesia

Table of Contents

Matrix Integration with Jitsi Conference Server for Online Learning

Rini Widyastuti[1,], Karmila Suryani[2], Ade Fitri Rahmadani[3], Triadmoko Denny Fatrosa[4], Wandi Syahindra[5]*

[1,2,3,4]*Informatics and Computer Engineering Education, Bung Hatta University, Padang, Indonesia*
[5]*State Islamic Institute (IAIN), Curup, Indonesia*

Article Information

A B S T R A C T

The current online learning process requires virtual face-to-face media to interact between lecturers and students. However, currently available media requires lecturers to pay a fee to communicate indefinitely. Therefore, we need a media that can facilitate lecturers to do learning easily. This study aims to integrate jitsi with matrix conferences as a real-time online communication medium in learning. This type of research is development research (R&D) with the waterfall method. The steps taken are analysis, design, coding and testing. The analysis is carried out on the system development needs, then the design stage of the Jitsi architecture with a matrix, then coding using PHP and testing through QoS. QoS testing has very good throughput results, poor packet loss, very good delay, and excellent jitter for small-scale applications. It shows that picking video conferencing can be used as a medium of communication in online learning.

1. Introduction

The current development of Information and Communication Technology (ICT) requires all aspects of life to adapt both in the fields of economy, trade and education [1][2]. Especially when the Covid-19 pandemic entered Indonesia in mid-March 2020, the government issued a policy to temporarily eliminate face-to-face learning and replace it with online learning at both the school and college level [3]. Online learning (in the network) requires learning media to bring students and educators together directly through virtual face-to-face. One of the media that can be used is through a video conference platform to implement learning at both the school and college levels so that the educational process continues [4].

Several studies that have been conducted regarding video conferencing include, [5] using jitsi and raspberry PI 3 on home video conferencing devices for baby monitoring. In addition, [6] has collaborated with Jitsi Meet to plan home video conferencing tools for Harvie products. Meanwhile, [7] revealed that the resulting video conference could facilitate online learning through direct file transfer. However, the web-based video conference that has been produced has not yet collaborated between jitsi and the matrix. Therefore, the researcher developed a video conference by integrating jitsi and the matrix called a quote video conference.

Picked web conference provides a platform for all lecturers for free; its features can be retrieved and reviewed such as communication history, file transfer, search public rooms services and create groups. In addition, picking web conferences can be used in real-time by taking advantage of the features for free and

with no time limit. Learning using video conferencing technology is a more exciting and effective choice in carrying out online learning.

2. Method

The type of research used is development research (R&D), which in this study produces a new product [8] using the waterfall development model [9]. The waterfall procedure carried out consists of the following stages.

2.1. Analysis

At this stage, an analysis of the system is carried out , namely how the platform can be run properly. Requirements analysis is focused on hardware and software analysis.

2.2. Design

The system design stage can assist in determining the requirements of the system and defining the overall system architecture.

2.3. Coding

The coding stage compiles the design results into program lines that are compiled using the API (Application Programming Interface). The coding stage is carried out to implement the design results into program lines. The system will configure in the VPS that has been provided so that users can access the platform anywhere and anytime. The first configuration that must be considered is the need for the software that will be used to design the system; this stage must be carried out in stages, namely, the Web Server (NGINX) is an open-source web server, which can also function as an IMAP/POP3 proxy. Before installing NGINX server must be updated first. Then install the Python Cerbot Plugin. This plugin functions to change the http protocol to https, so that the existing communication permissions on the domain can be used for security.

2.4. Test

System testing is a software system execution process to determine whether the software system matches the system specifications and runs in the desired environment. System testing is often associated with finding bugs, program imperfections, program line errors that cause system software execution failures. The test uses the QoS method with the following parameters.

2.4.1. Throughput

Throughtput is the average value of successful delivery through a telecommunications channel in a delivery. Throughput is measured in bits per second (bps or bit/s). The formula for calculating throughput is shown in equation 1.

$$\text{Throughtput (bps)} = \frac{Received\ data\ packets}{Package\ delivery\ time\ (second)} \tag{1}$$

2.4.2. Delay

Delay is a common problem that occurs in telecommunications networks. Delay is the time it takes for a packet to travel from delivery to recipient. The formula for calculating delay is shown in equation 2.

$$\text{Delay } \bar{x} = \frac{Total\ delay}{Total\ Packages\ received} \tag{2}$$

The classification of delay standardization based on the TIPHON standard is shown in table 1.

Table 1. Standardization of Delay Based on TIPHON Standard

No	Category	Large (ms)
1	Very good	<150
2	Good	150 – 300
3	Pretty good	300 – 500
4	Not Recommended	>450

The delay value is categorized as very good if it gets less than 150 ms, good value if it gets between 150 ms to 300 ms, it is quite good if it gets between 300 ms to 450 ms and if it gets more than 450 ms then it is not recommended to use.

2.4.3. Jitter

Jitter is a variation of delay that occurs due to queue length in data processing and reassemble data packets at the end of delivery due to previous failures. The formula for calculating Jitter is shown in equation 3.

$$\text{Jitter} = \frac{(delay2 - delay1) + (delay3 - delay2) + \cdots + (delay\ n - delay\ (n-1))}{Total\ Packages\ received} \tag{3}$$

2.4.4. Packet Loss

Packet loss is a parameter that describes conditions that indicate the number of lost packets. The formula for calculating packet loss is shown in equation 4.

$$\text{Paket Loss} = \frac{(Data\ packets\ sent\ -\ Data\ packets\ received) x\ 100\%}{data\ packet\ sent} \tag{4}$$

The classification of packet loss standardization based on the TIPHON standard is shown in table 2.

Table 2. Standardization of Packet Loss Based on TIPHON Standard

No	Category	Large (ms)
1	Very good	0-2
2	Good	3-14
3	Pretty good	15-25

3. Results and Discussion

3.1. Analysis

This analysis includes analysis of hardware and software requirements. The hardware used in the design of this application are as follows: (1) Processor : Intel(R) Core(TM) i3-3217U CPU @ 1.80 GHz; (2) RAM : 4.00 GB; (3) Hard disk : 500 GB; (4) Standard optical and keyboard; (5) Monitor resolution of 1366 x 768 pixels. While the software used is: (1) Windows 10 Single Language 64-bit operating system; (2) Google Chrome Internet Browser; (3) Debian 10 64 bit VPS, 4) DNS; (5) NGINX; (6) Cerbot plugin; (7) Synapse-Matrix; and (8) Wireshark.

3.2. Design

3.2.1. Matrix Architectural Design

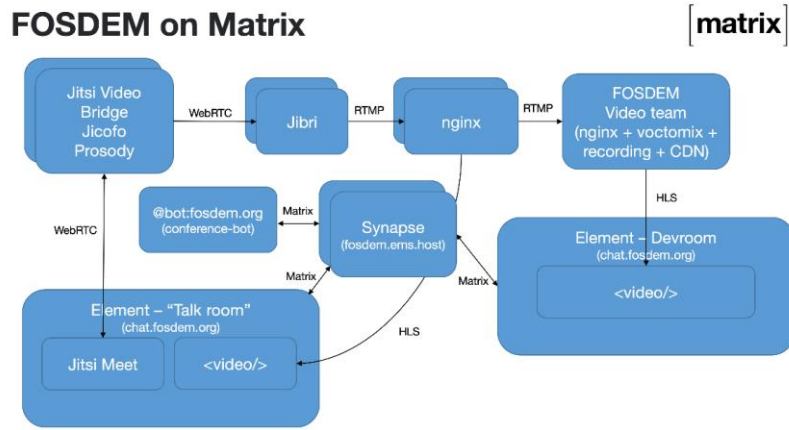The application is designed using the architecture as shown in Figure 1.

Figure 1. Matrix Architecture

Figure 1 is a video conference design scheme that will be integrated with jitsi [10] revealing that there are four main components on the matrix side, namely 1) Application of a horizontally scalable Matrix server (Synapse hosted on EMS), 2) Jitsi cluster for conferences video, used to host all Q&A sessions, hallway sessions, booth and other adhoc video conferencing, 3) The scalable, elastic Jibri Cluster is used for Jitsi conference streaming both for official FOSDEM live stream and to provide local previews of Matrix conferences (to avoid Jitsis getting overloaded with people who only want views), and 4) Conference bots Matrix that manages the entire conference in Matrix, written from scratch for FOSDEM by TravisR, uses the schedule from FOSDEM and maintains all necessary rooms with proper permissions, widgets, invitations, etc.

This design stage can be seen in Figure 2.
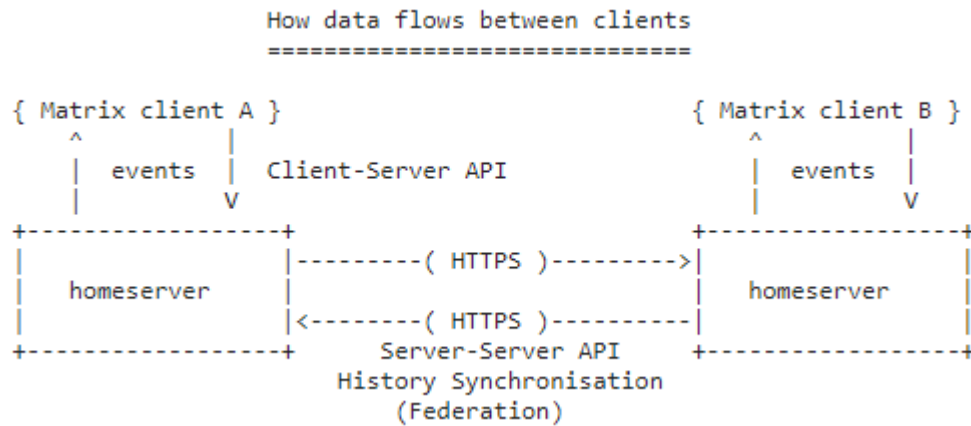
Figure 2. Example of User A Sending a Message to User B

Figure 2 shows that Matrix defines an API for synchronizing extensible JSON objects known as "events" between compatible users , servers and services. Typically users use messaging applications, VoIP (Voice over Internet Protocol), IoT (Internet of Things) and other communications to store communication history and account information for all their users, and share data with the wider matrix ecosystem by synchronizing the communication history with the origin server and other users.

Furthermore, the architecture of the room can be seen as in Figure 3.

```
{ @alice:matrix.org }                         { @bob:example.org }
         |                                              ^
         |                                              |
[HTTP POST]                                   [HTTP GET]
Room ID: !qporfwt:matrix.org                  Room ID: !qporfwt:matrix.org
Event type: m.room.message                    Event type: m.room.message
Content: { JSON object }                       Content: { JSON object }
         |                                              |
         V                                              |
+-------------------+                         +-------------------+
|    homeserver     |                         |    homeserver     |
|    matrix.org     |                         |    example.org    |
+-------------------+                         +-------------------+
         |                                              ^
         |          [HTTP PUT]                          |
         |          Room ID: !qporfwt:matrix.org        |
         |          Event type: m.room.message          |
         |          Content: { JSON object }            |
         `-------> Pointer to the preceding message ----`
                    PKI signature from matrix.org
                    Transaction-layer metadata
                    PKI Authorization header

                  ...................................
                  |            Shared Data          |
                  | State:                          |
                  |   Room ID: !qporfwt:matrix.org  |
                  |   Servers: matrix.org, example.org |
                  |   Members:                      |
                  |    - @alice:matrix.org          |
                  |    - @bob:example.org           |
                  | Messages:                       |
                  |    - @alice:matrix.org          |
                  |      Content: { JSON object }   |
                  |.................................|
```
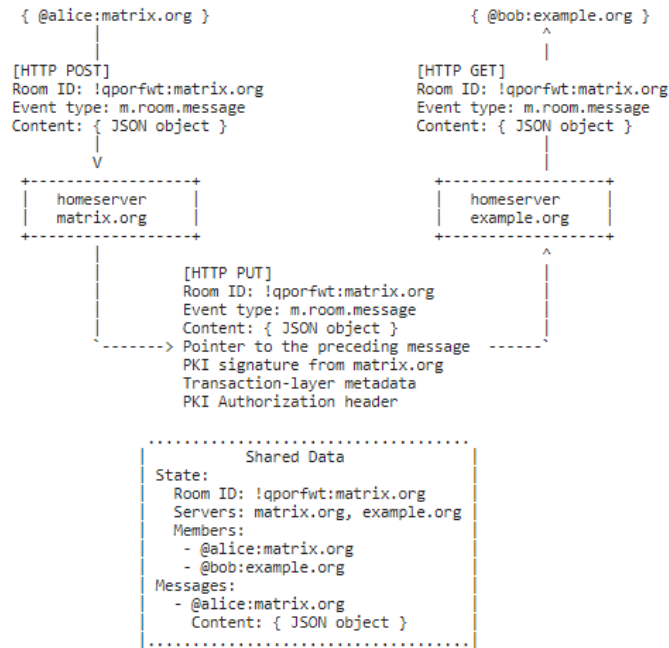
Figure 3. Room Architecture

Figure 3 illustrates that the matrix maintains a per-room data structure between multiple servers. The data is divided into message events and state events. The status of the room at a given point is calculated taking into account all previous events and including certain events in the graph. Where events describe the same state, a merge conflict algorithm is applied. The state resolution algorithm is transitive and independent of the server state because it must consistently select the same events regardless of the server or the order in which events are received. Events are signed or encrypted by the origin server (the signature includes the parent relation, type, depth and payload hash) and sent to the participating servers in a room, currently using a mesh topology. Servers can also request event reloads from other servers participating in a room via federation.

3.2.2. *Jitsi Design*

When you first open the video conferencing application using a browser, it appears as shown in Figure 4.
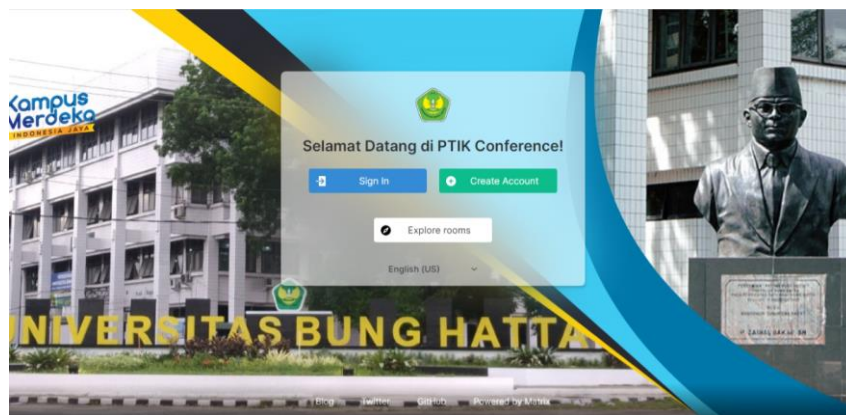

Figure 4. Initial View of the Web Conference

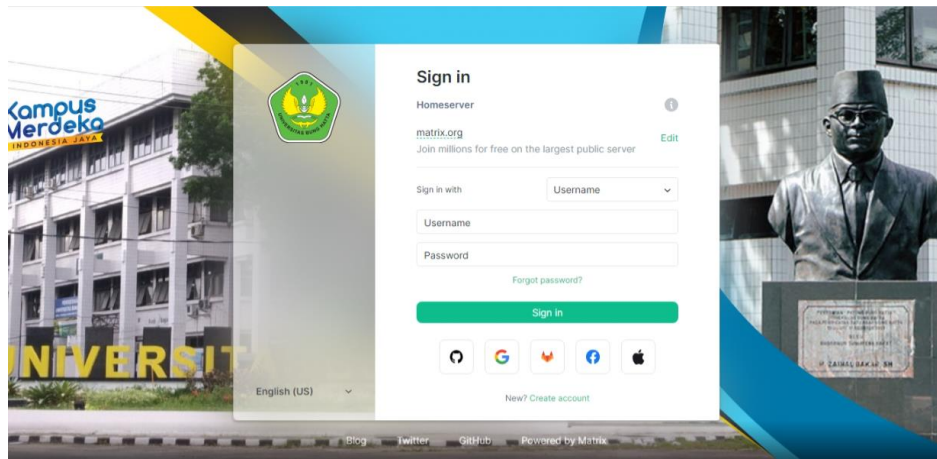Figure 5 is this page to enter the video conferencing application.

Figure 5. Display Sig In Application

Figure 5 shows that users can enter the application if they have a Github, Google, Gitlab, Facebook, Apple account and users can also sign in manually. To register to be able to log into the application, you can select the create Account button as shown in Figure 6. Users can also register into the application if the user has a Github , Google, Gitlab, Facebook, Apple account and users can also register manually.
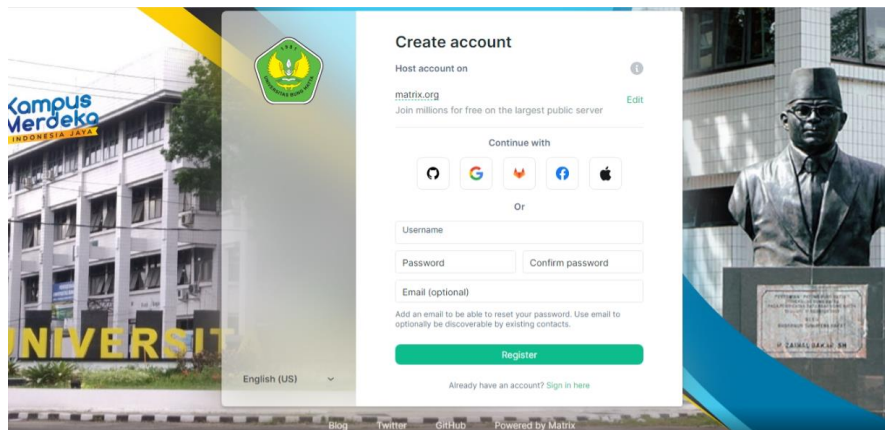

Figure 6. Display Create Account

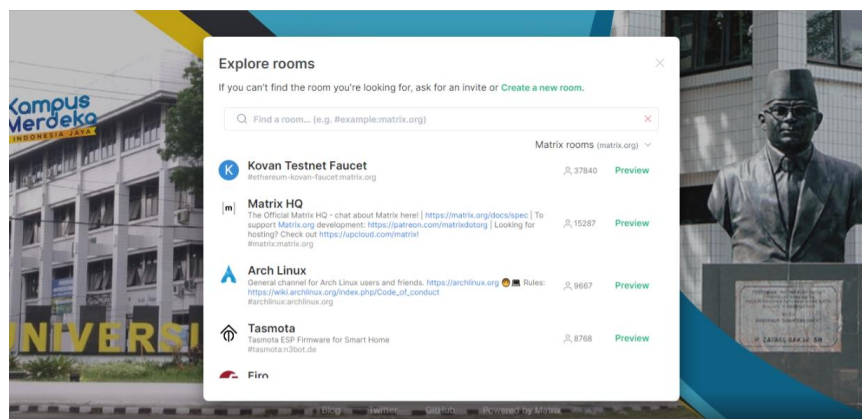In Figure 7, users can search for discussion rooms if the rooms are public.

Figure 7. Explore Rooms Display

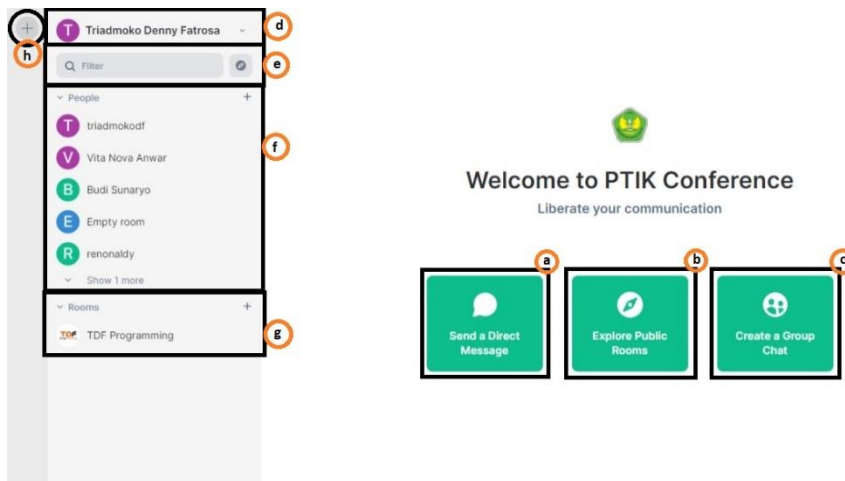When the user has registered and has successfully signed in, it will look like in Figure 8.

Figure 8. Display After Sign In

Figure 8 describes that the user can start to communicate with users of others, seek public rooms, create a group, see the settings of the profile of the user, can filter the conversation, and create a community that can group several rooms. People is a communication history that the user has not deleted and there is a list of rooms that have been joined.

Web conferencing online learning model can be with voice call chat [11]. The voice call chat page can see the user name , time or hour when chatting, and the chat date between users, as shown in Figure 9.
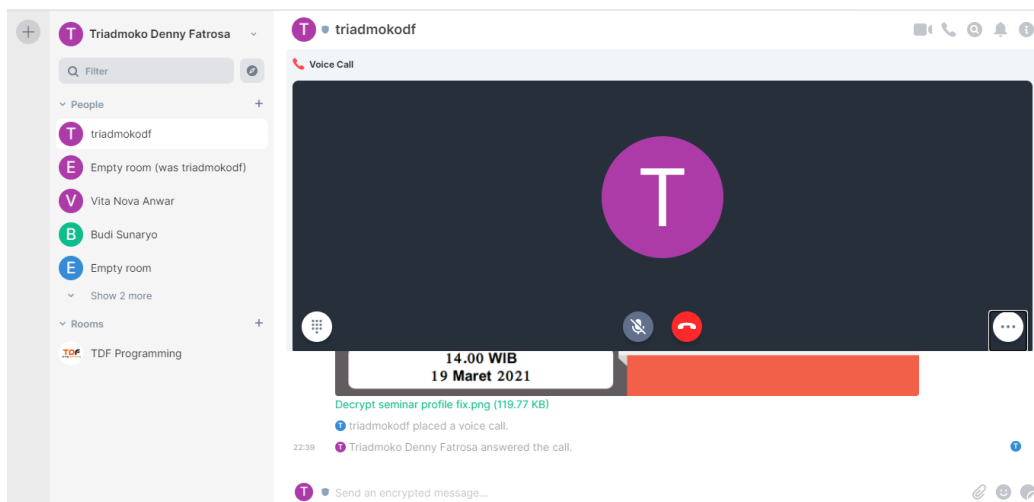

Figure 9. Voice Call Chat Display

3.3. Coding

The coding made on the web conference application uses a web server (NGINX). Before installing NGINX, the server must be updated. Broadly speaking, such as the following program listings.

```
$ apt-get update
```

After updating the server, NGINX will be installed using a script such as the following program listing:

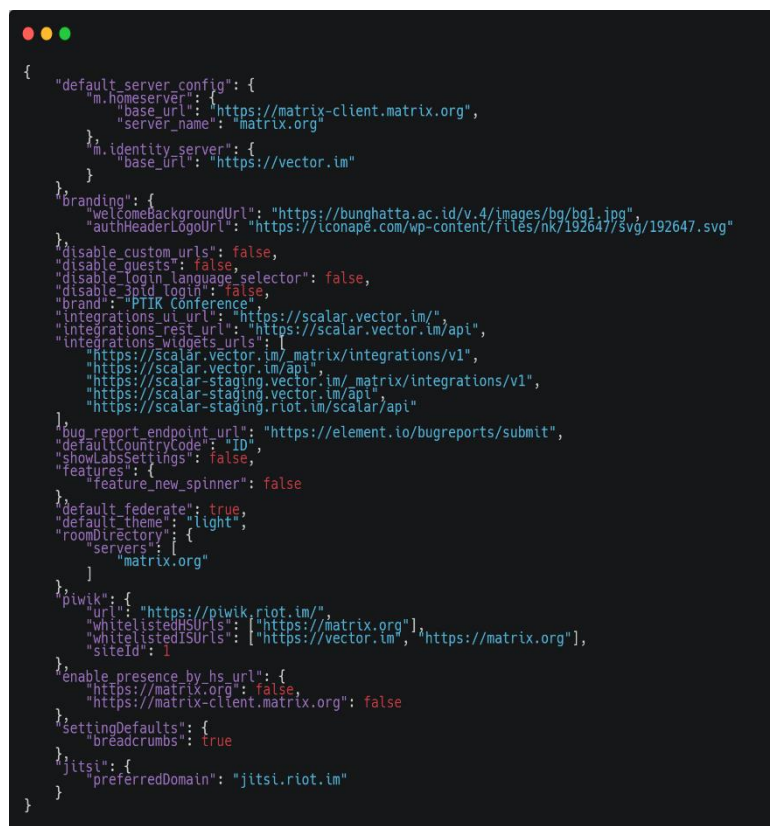```
$ apt-get install nginx
```
Configure the nginx web server according to the needs of the sub-domain to be used.

```
$ cd /etc/nginx/sites-enabled
$ nano ptikconf.xyz
```

Then type a script like the following for the pticconf.xyz domain

```
Server {
    listen 80;
    listen[::]:80;
    server_name ptikconf.xyz;
    root /var/www/ptikconf.xyz;
    index.html index;
    location / {
            try_files $uri $uri/ =404;
    }
}
 $ nano meet.ptikconf.xyz
Server {
    listen 80;
    listen[::]:80;
    server_name meet.ptikconf.xyz;
    root /var/www/meet.ptikconf.xyz/meet;
    index.html index;
    location / {
            try_files $uri $uri/ =404;
    }
}
$ nano matrix.ptikconf.xyz
Server {
    listen 80;
    listen[::]:80;
    server_name matrix.ptikconf.xyz;
    root /var/www/ptikconf.xyz;
    index.html index;
    location / {
            proxy_pass http://localhost:8008
    }
}
```

For setting the config.json settings can be seen as in Figure 10.

Figure 10. Config.json settings

3.4. Test

The software execution process in system testing is needed to determine whether the software matches the system specifications and can run with the desired environment. This test is associated with finding bugs, program imperfections, program line errors that cause system software execution failures. The system testing used is QoS which will be carried out in stages, namely Throughput, Delay, Jitter, Packet Loss.

Wireshark software testing is used to capture all passing packets and analyze each packet type. The process of capturing all packets by Wireshark will be recorded, then analyzed by calculating the distance between packets (jitter) and bandwidth usage on RTMP packets according to the type of packet used in video conferencing applications [12].

Wireshark testing is done by recording packets sent and packets received using the indihome network on a Telkom provider with an internet bandwidth of 20 Mbps. Internet speed testing using the Measurement Lab (M-Lab) with a period of 30 seconds. This test typically sends packets of less than 40 MB, but can transfer larger data on fast internet connections.

The results of the QoS test can be seen in table 3.

Tabel 3. Hasil Pengujian QoS

No	Parameter	Mark	Qualification
1	Throughput	5065.472 bps	Very good
2	Packet Loss	0,003 %	Very good
3	Delay	1,268 ms	Very good
4	Jitter	3,803 ms	Very good

Table 3 shows that the test results for all parameters such as throughput, packet loss, delay and jitter have very good qualifications. Therefore, the learning process through video conferencing can run well. It is in line with research conducted by [13] that if the throughput produced is at a minimum in the range of 250-600 kbps, the video conferencing process is stable.

4. Conclusion

Based on the study results, it was concluded that jitsi had been integrated with the matrix as a new learning platform in the form of video conferencing. The video conferencing application is given a quote to make it easier to access and has gone through the testing stages using QoS with very good qualifications. The picking web conference is directly connected to the Bung Hatta University website so that all activities can be recorded directly on the portal of each lecturer.

References

[1] M Sya'rani Machrizzandi, "Dampak Teknologi Informasi terhadap Kondisi Ekonomi Mahasiswa di Masa Pandemi Covid," *J-ALIF J. Penelit. Huk. Ekon. Syariah dan Sos. Budaya Islam*, vol. 5, no. 2, pp. 1–13, 2020.

[2] yunus Yunus, "Pengaruh Ti Media Sosial Terhadap Prestasi Belajar Pendidikan Agama Islam Peserta Didik Di Smp Negeri 3 Mappadeceng Kabupaten Luwu Utara," *J. Teknol. Inf. ESIT*, vol. M, no. 1, pp. 16–25, 2019.

[3] E. Novianti, "Strategi Sumber Daya Manusia Di Masa Pandemi Covid19 Melalui Kebijakan PJJ Oleh Pemerintah," *Researchgate.Net*, vol. 1, no. 2, 2011.

[4] I. B. A. I. Iswara and I. P. P. K. Yasa, "Analisis Dan Perbandingan Quality of Service Video Conference Jitsi Dan Bigbluebutton Pada Virtual Private Server," *J. Resist. (Rekayasa Sist. Komputer)*, vol. 4, no. 2, pp. 192–203, 2021.

[5] A. Purnamajati, I. N. B. A. K, and D. Perdana, "Desain Produk Harvie Untuk Home Video Conference Pada Smart Home Menggunakan Raspberry Pi 3 Product Design Harvie in Home Video Conference for Smart Home Using Raspberry Pi 3," vol. 7, no. 3, pp. 9019–9027, 2020.

[6] R. Z. Zulkarnaen, F. T. Elektro, U. Telkom, and B. Monitoring, "Analisis Performansi Produk Harvie Untuk Home Video Conference Performance Analysis of Harvie Products for Home Video," vol. 7, no. 3, pp. 8911–8918, 2020.

[7] H. Sandra, "Integrasi PHP Native Dengan Sistem Jitsi Converence Server Menggunakan API," vol. IV, no. 1, pp. 8–15, 2021.

[8]   Sugiyono, *Metode Penelitian Kuantitatif, Kualitatif, dan R&D*. Bandung, 2018.

[9]   M. S. A. S., Rosa, *Rekayasa Perangkat Lunak Terstruktur Dan Berorientasi Objek*. Bandung: Informatika, 2018.

[10] D. Laksmiati, "Implementasi Cloud Based Video Conference System Menggunakan Jitsi," *J. Akrab Juara*, vol. 4, no. 1, pp. 219–225, 2019.

[11] E. Kurtarto, "Keefektifan Model Pembelajaran Daring Dalam Perkuliahan Bahasa Indonesia Di Perguruan Tinggi," *J. Indones. Lang. Educ. Lit.*, vol. 1, no. 2, pp. 207–220, 2017.

[12] Y. Pujilestari, "Dampak Positif Pembelajaran Online Dalam Sistem Pendidikan Keperawatan Pasca Pandemi Covid 19," *J. Perawat Indones.*, vol. 4, no. 3, p. 432, 2020.

[13] J. P. Tarigan *et al.*, "Pengaruh Metode Classful Queuing Disciplines terhadap Efisiensi Penggunaan Bandwidth Aplikasi Video Conference," pp. 4–5, 2021.