

# APLIKASI OPENSOURCE MULTIPLATFORM VOIP CLIENT MENGGUNAKAN FRAMEWORK SIP BERBASIS JAVASCRIPT

## OPENSOURCE MULTIPLATFORM VOIP CLIENT APPLICATION USING SIP FRAMEWORK BASED ON JAVASCRIPT

Muhammad Salman Abdul Qohar<sup>1</sup>, Hafidudin, ST., M.T.<sup>2</sup>, Dadan Nur Ramadan, S.Pd., M.T.<sup>3</sup>

<sup>123</sup>Fakultas Ilmu Terapan, Universitas Telkom

Jln. Telekomunikasi Dayeuhkolot Bandung 40257

<sup>1</sup>souhart@gmail.com, <sup>2</sup>hafidudins@gmail.com, <sup>3</sup>dadan.nr@gmail.com

---

### Abstrak

Teknologi Voice Over Internet Protocol (VOIP) berkembang pesat, teknologi ini memungkinkan manusia dapat berkomunikasi dalam bentuk pesan, suara, atau video melalui jaringan Internet Protocol (IP). Di sisi lain, sistem operasi yang digunakan untuk smartphone semakin banyak dan berkembang pesat, misalnya android, windows phone, iOS, dan sebagainya. Belakangan ini, aplikasi VoIP untuk smartphone semakin populer tetapi aplikasi VoIP yang bersifat opensource dan crossplatform masih belum begitu berkembang. Oleh karena itu, proyek akhir ini membahas tentang perancangan aplikasi VoIP client multiplatform yang bersifat opensource.

Perancangan aplikasi VoIP Client pada proyek akhir ini dimulai dengan perancangan API sebagai penghubung antara database dengan aplikasi web atau mobile, pengembangan aplikasi web client menggunakan bootstrap sebagai Framework CSS, angularJS sebagai web framework. Untuk menghubungkan aplikasi client dengan server, penulis memanfaatkan teknologi WebRTC dengan framework SIP berbasis javascript. Sedangkan pengembangan aplikasi mobile client dilakukan dengan cara code reuse dari aplikasi web menggunakan Crosswalk.

Hasil yang didapatkan setelah pembuatan proyek akhir ini adalah sebuah produk berupa aplikasi VoIP SIP Client yang dapat berjalan pada website dan android, dan bersifat opensource. Nilai MOS dari QoS untuk panggilan suara menggunakan aplikasi ini adalah 4.24 dan untuk panggilan video adalah 4.25 atau baik.

Kata Kunci : VOIP, SIP, Javascript, Multiplatform, Web RTC

---

### Abstract

Voice Over Internet Protocol (VOIP) Technology is growing so fast, this technology allow us to communicate using text, voice, or video call over an Internet Protocol (IP) network. Then, the smartphone operating system is being rapidly growing such as Android, WindowsPhone, iOS, and so on. In recent years, VoIP application for smartphones are increasingly popular but VoIP applications that are opensource and crossplatform still underdeveloped. Therefore, this final project is explaining the developement of Opensource Multiplatform VOIP Client using SIP Framework based on Javascript.

The application developement of this VoIP Client apps began with the design of the API as a bridge between the database server and an application, client web application is developed using bootstrap CSS Framework, angularJS as a web framework, and for the gateway from client to the VOIP server, author using the WebRTC technology with Javascript SIP Framework. While the developement of mobile client application uses Crosswalk for reusing code from web application.

The results obtained after the making of this final project is a VoIP SIP Client application that can run on website or android. The license of the source code will be opensource. The MOS QOS Value of voice call using this application is about 4.24 and for video is about 4.25 or has a good quality.

Keywords: VOIP, SIP, Javascript, Multiplatform, Web RTC

---

### 1. Pendahuluan

Teknologi Voice Over Internet Protocol (VOIP) merupakan sebuah teknologi pada bidang telekomunikasi yang saat ini sedang berkembang cepat, teknologi ini memungkinkan manusia dapat berkomunikasi dengan biaya yang sangat terjangkau melalui jaringan Internet Protocol (IP). Meskipun teknologi VOIP ini sudah sangat populer bahkan cukup banyak provider yang menyediakan Server VOIP yang dapat digunakan secara bebas dan

gratis, tetapi belum ada aplikasi VOIP Client yang bersifat opensource dan dapat dengan mudah digunakan melalui browser tanpa plugin apapun.

Di samping itu, sistem operasi yang dibuat untuk smartphone dengan prosesor ARM berkembang dengan sangat cepat dan jenisnya semakin banyak. Fenomena perkembangan teknologi ini disambut baik oleh masyarakat dengan banyaknya sistem operasi untuk ponsel maka menghasilkan pengalaman yang berbeda pada saat pemakaiannya, tetapi ini juga memaksa pihak industri yang bergerak di bidang application development untuk membuat aplikasi bukan hanya untuk diterapkan pada satu sistem operasi saja, jika developer dan pihak industri menginginkan aplikasinya populer, maka aplikasi tersebut harus tersedia pada berbagai platform.

Oleh karena itu berdasarkan latar belakang tersebut, untuk mendukung fenomena perkembangan teknologi VOIP seperti yang telah dijelaskan sebelumnya, penulis melalui proyek akhir ini melakukan perancangan sebuah produk berupa aplikasi VoIP Client multiplatform dengan protocol SIP yang dapat digunakan melalui web browser maupun smartphone Android, aplikasi ini bersifat opensource dan dapat digunakan oleh pihak manapun untuk kepentingan belajar, riset, atau bahkan untuk pihak industri yang ingin mengembangkan produknya.

## 2. Dasar Teori dan Perancangan

### 2.1 Voice over Internet Protocol (VoIP)

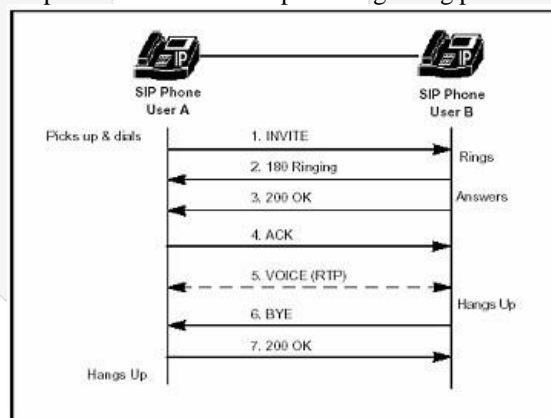
Voice over Internet Protocol (VoIP) merupakan teknologi transmisi data untuk melakukan komunikasi suara melalui jaringan TCP/IP seperti Internet ataupun jaringan packet-switched lainnya. Dikenal juga sebagai IP telephony, Internet Telephony, Broadband Telephony, ataupun Voice over BroadBand (VoBB), walau yang paling sering digunakan saat ini adalah Voice over IP (VoIP). Teknologi ini berbeda dengan telepon konvensional (PSTN) dimana data media percakapan ditransmisikan sebagai data digital. Layanan VoIP umumnya melibatkan sejumlah protokol, yang dapat dibagi ke dalam kelompok protokol pensinyalan dan protokol transport media.<sup>[2]</sup>

### 2.2 Session Initiation Protocol<sup>[8]</sup>

SIP adalah suatu signalling protocol pada layer aplikasi yang berfungsi untuk membangun, memodifikasi, dan mengakhiri suatu sesi multimedia yang melibatkan satu atau beberapa pengguna. Sesi multimedia adalah pertukaran data antara pengguna yang meliputi suara, video, atau text. Komunikasi pada SIP dilakukan dengan mengirimkan message yang berbasis HTTP. Setiap pengguna mempunyai alamat yang dinyatakan dengan SIP-URI (Uniform Resource Identification).

Contoh SIP URI : sip:martin@bandung.com

Hubungan yang dibangun oleh SIP pada proses signalling bersifat client-server. Dengan demikian ada dua jenis message, yaitu request dan response. Berikut adalah proses signaling pada SIP.



Gambar 1 Signaling pada SIP

### 2.3 SIP Software Development Kit

SDK merupakan satu set perangkat lunak yang digunakan untuk pengembangan suatu perangkat lunak pada satu platform, biasanya SDK menyediakan satu atau beberapa Application Programming Interface (API). SIP SDK merupakan sebuah paket perangkat lunak yang menyediakan framework dan API dalam pembuatan aplikasi SIP Client, beberapa contoh SIP SDK yang berbasis javascript adalah :

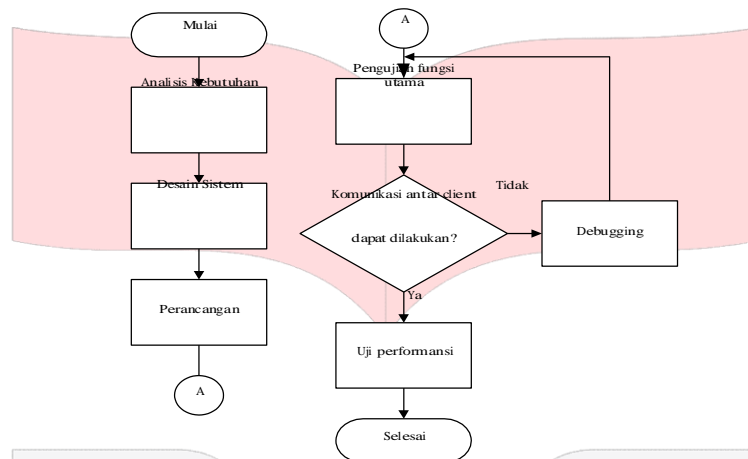
1. SipJS (sipjs.com)
2. JsSIP (jssip.net)
3. sipML5 (sipml5.org)
4. Linphone

## 2.4 Web RTC

WebRTC (Web Real-Time Communications) merupakan sebuah proyek open-source yang memungkinkan untuk dilakukannya komunikasi real-time lintas web browser<sup>[9]</sup>. Teknologi ini berjalan diatas backbone web browser modern. Komunikasi tersebut nantinya terdiri dari pemanfaatan suara, video dan konektivitas menggunakan Javascript API tanpa plugin tambahan.

## 2.5 Alur Pengerjaan

Agar pada pengerjaan proyek akhir ini lebih teratur, maka dibuatlah rencana pembuatan proyek akhir dengan alur pengerjaan sesuai dengan metode pengerjaan yang telah dijelaskan pada bab 1 yang digambarkan dengan diagram alir sebagai berikut.



Gambar 2 Alur Pengerjaan

Pengerjaan aplikasi VOIP Client ini dimulai dengan analisis kebutuhan sistem, dilanjutkan dengan tahap desain arsitektur jaringan dan arsitektur sistem, setelah itu dilakukan perancangan sampai fungsi utama dari aplikasi VOIP Client ini dapat berjalan dengan baik, pada tahap akhir setelah fungsi utama aplikasi dinilai cukup baik maka akan dilakukan pengujian performansi QoS untuk hubungan antar client dan diakhiri dengan publikasi source code pada internet.

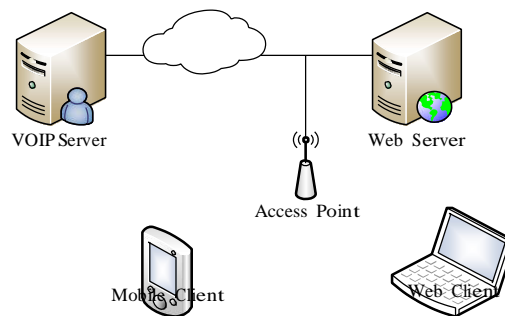
## 2.6 Analisis Kebutuhan

1. Kebutuhan perangkat lunak server dan client :
  - a. SIP Server + WebRTC Support
  - b. HTTP Server (Apache)
  - c. Relational Database Server (MySQL)
  - d. Sistem Operasi.
  - e. Web Browser (Google Chrome)
2. Kebutuhan perangkat lunak untuk perancangan aplikasi :
  - a. Phonegap / Cordova
  - b. Crosswalk
  - c. Node.js dan Node Package Manager
  - d. Android Software Development Kit
  - e. Text Editor
3. Kebutuhan framework untuk perancangan aplikasi
  - a. Slim Framework
  - b. Bootstrap
  - c. Ionic Framework
  - d. AngularJS
  - e. JQuery
  - f. SipML5 - Javascript SIP Framework

4. Kebutuhan perangkat keras
  - a. PC Server
  - b. Laptop + built in webcam dan microphone (client)
  - c. Smartphone Android (client)
  - d. Access Point (TL-MR3020)
  - e. Kabel Ethernet

## 2.7 Arsitektur Jaringan

Pada proyek akhir ini dibutuhkan arsitektur jaringan yang sesuai untuk mendukung skenario percobaan aplikasi VOIP ini agar komunikasi antar Mobile Client dan Web Client berjalan dengan baik, maka dari itu akan dibangun arsitektur jaringan lokal dengan beberapa perangkat yang akan dijelaskan dibawah ini.

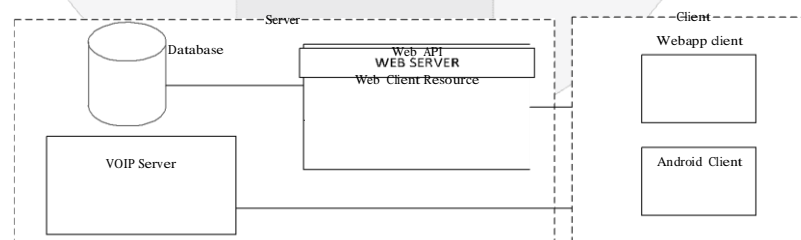


Gambar 3 Arsitektur jaringan

Skenario jaringan yang akan digunakan pada proyek aplikasi VOIP ini adalah seperti yang telah digambarkan diatas, yaitu komputer yang digunakan sebagai VOIP server dihubungkan langsung dengan access point melalui kabel ethernet, sedangkan client yang akan diujicobakan yaitu laptop dan smartphone akan dihubungkan dengan access point melalui jaringan nirkabel (wifi).

## 2.8 Arsitektur Sistem

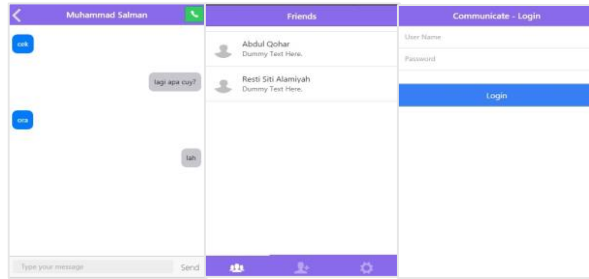
Pada perancangan arsitektur sistem akan digunakan elemen-elemen sistem yang telah disebutkan diatas dan akan dibagi pada dua bagian yaitu untuk sisi server dan sisi client, pada sisi server akan ditambahkan aplikasi-aplikasi seperti server SIP, Database Server, WebRTC Support, HTTP Server, Web API, Webapp Client. Sedangkan aplikasi mobile pada sisi client menggunakan code yang telah dipakai pada Webapp client. Penjelasan dan ilustrasi yang lebih detil dijelaskan dibawah ini.



Gambar 4 Arsitektur sistem

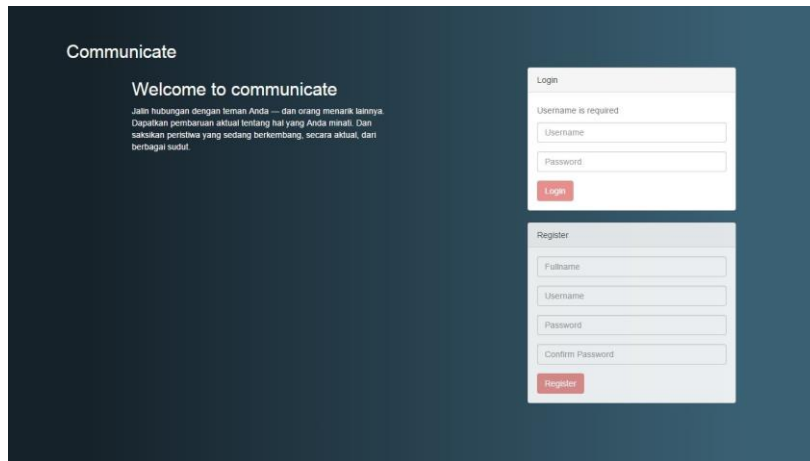
## 2.9 Perancangan Antarmuka

Berikut adalah interface mobile client aplikasi.

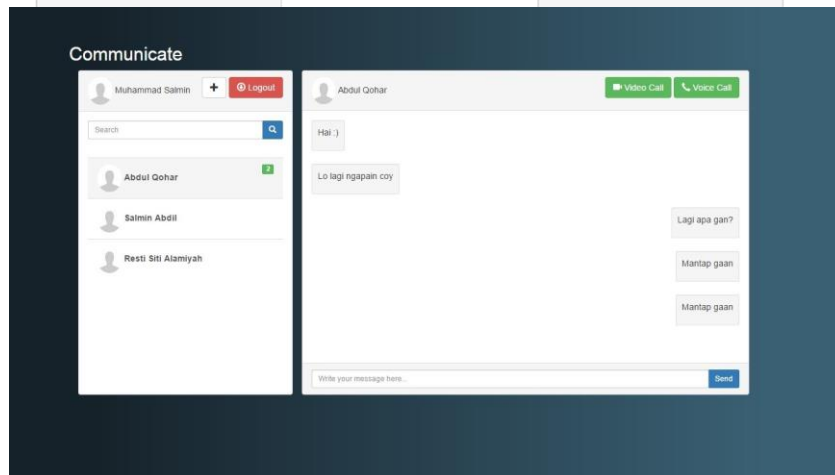


Gambar 5 Antarmuka Mobile Client

Berikut adalah interface dari webapp client.



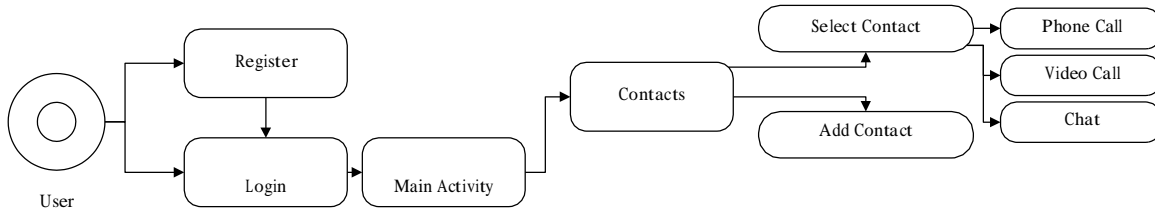
Gambar 6 Antarmuka Halaman Login



Gambar 7 Antarmuka Halaman Chat

### 2.10 Diagram Aktivitas Pengguna

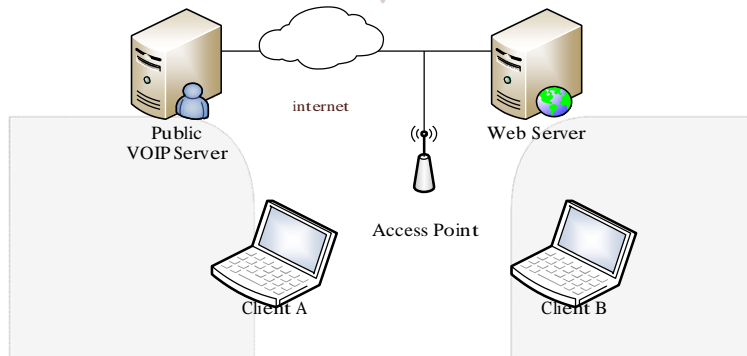
Pada sisi pengguna, untuk kemudahan dan kenyamanan dalam penggunaan, aplikasi ini akan dibuat sesederhana mungkin supaya dapat dengan mudah dimengerti oleh pengguna, berikut adalah aksi yang dapat dilakukan oleh pengguna yang digambarkan melalui diagram berikut.



Gambar 8 Diagram Aktivitas Pengguna

3 Pengujian dan Analisis  
3.1 Skenario Pengujian

Pengujian dilakukan dengan menggunakan 2 komputer, yaitu client A dan client B. dimana pengukuran dimulai pada saat kedua client sudah terhubung dengan server SIP melalui jaringan internet, lalu melakukan panggilan suara selama 120 detik dan panggilan video selama 120 detik. Untuk memperoleh nilai parameter dalam pengujian, dilakukan proses pengambilan data trafik paket pada sisi client A dengan menjalankan wireshark pada saat kedua client mulai melakukan panggilan suara dan video. Parameter pada pengujian kali ini adalah delay, packet loss, dan troughput. Berikut adalah model skenario pengujian yang dilakukan.



Gambar 9 Skenario Pengujian

3.2 Pengambilan data

Pengambilan data paket untuk perhitungan delay, loss, dan troughput dilakukan dalam 3 sesi untuk suara dan video, pengukuran dilakukan setiap 15 detik selama 300 detik (5 menit).

1. Trafik Panggilan Suara

Tabel 1 Trafik panggilan suara

Waktu	Sent Packets	Time between first and last packets (s)	Troughput (kbps)
Pagi (6.20)	16278	299,56	64
Siang (14.30)	16271	299.30	64
Malam (22.15)	16293	299,92	63

2. Trafik Panggilan Video

Tabel 2 Trafik panggilan video

Waktu	Sent Packets	Time between first and last packets (s)	Troughput (mbps)
Pagi (7.00)	92585	299,98	2,223

Siang (15.00)	97618	299.82	2.223
Malam (23.15)	81356	299,32	2.055

### 3.3 Analisis Perhitungan Delay

Dalam perhitungan delay berdasarkan hasil pengambilan data menggunakan wireshark dapat digunakan persamaan berikut.

$$\text{Delay Tx} = (\text{Time between first and last packet}) / (\text{Jumlah Paket})$$

Berikut adalah hasil perhitungan delay dari hasil capture wireshark

#### 1. Delay panggilan suara

Tabel 3 Delay panggilan suara

Waktu	Sent Packets	Time between first and last packets (s)	Troughput (kbps)	Delay (ms)
Pagi (6.20)	16278	299,56	64	18.40
Siang (14.30)	16271	299.30	64	18.39
Malam (22.15)	16293	299,92	63	18.40
Rata-rata			63.4	18.40

#### 2. Delay panggilan video

Tabel 4 Delay panggilan video

Waktu	Sent Packets	Time between first and last packets (s)	Troughput (mbps)	Delay (ms)
Pagi (7.00)	92585	299,98	2,223	3.24
Siang (15.00)	97618	299.82	2.223	3.07
Malam (23.15)	81356	299,32	2.055	3.67
Rata-rata			2.167	3.33

### 3.4 Analisis Perhitungan Packet Loss

Dalam menghitung packet loss berdasarkan hasil pengujian diatas dapat menggunakan persamaan dibawah ini.

$$\text{Packet Loss} = [(\text{PaketDiterima} - \text{PaketTer kirim}) / (\text{PaketDiterima})] \times 100\%$$

Dengan menggunakan rumus yang sama. Berikut hasil nilai packet loss untuk panggilan suara dan video.

Tabel 5 Packet Loss Suara

Waktu	Sent Packets	Delivered Packets	Packet Loss (%)
Pagi (6.20)	16278	16275	0.0184
Siang (14.30)	16271	16259	0.0721

Malam (22.15)	16293	16273	0.1228
Rata-rata			0.071

Tabel 6 Packet Loss Video

Waktu	Sent Packets	Delivered Packets	Packet Loss (%)
Pagi (7.00)	92585	92517	0.073
Siang (15.00)	97618	97560	0.059
Malam (23.15)	81356	81266	0.110
Rata-rata			0.081

### 3.5 Nilai MOS

Berikut adalah nilai MOS dari panggilan menggunakan aplikasi ini.

	Panggilan Suara			Panggilan Video		
	Pagi	Siang	Malam	Pagi	Siang	Malam
MOS	4.25	4.24	4.23	4.25	4.25	4.24
Rata-rata	4.24 (Baik)			4.25 (Baik)		

## 4 Kesimpulan

Dari perancangan dan pengujian performansi aplikasi VOIP Client ini, didapatkan beberapa kesimpulan sebagai berikut :

- Aplikasi dapat digunakan untuk komunikasi suara, video, dan teks dengan akun SIP publik dan dapat berjalan pada Smartphone android.
- Troughput yang didapat untuk panggilan suara adalah 63.4kbps sedangkan untuk panggilan video adalah 2167kbps.
- Delay yang didapat dari pengujian aplikasi ini adalah 18.4ms untuk panggilan suara, dan 3.33ms untuk panggilan video.
- Packet Loss yang didapat adalah 0.071% untuk panggilan suara, dan 0.081% untuk panggilan video. Keduanya dapat dikategorikan baik.
- Nilai MOS untuk panggilan suara adalah 4.24 dan untuk panggilan video adalah 4.25, keduanya dikategorikan baik.

## Daftar Pustaka

- [1] Azfar A., Kim-Kwang R., Liu L. 2014. A study of ten popular Android mobile VoIP applications: Are the communications encrypted?. 47th Hawaii International Conference on System Science.
- [2] Hariady, Ricky. 2011. Implementasi Protokol Keamanan TLS pada Aplikasi Layanan Komunikasi Suara Melalui Jaringan TCP/IP (VoIP). USU Institutional Repository.
- [3] Hulwa, Muhmilatul. 2008. Perancangan Prototype Sistem Pengalihan Nomor Panggilan Voip Ke Nomor Lokal Pstn Menggunakan Asterisk Softswitch. Bandung. Institut Teknologi Telkom
- [4] Irawan. 2008. Javascript untuk orang awam. Palembang: Penerbit Maxikom
- [5] Manuel P., Singh I., Cicchetti A. 2014. Comparison of Cross-Platform Mobile Development Tools. IEEE Computer Society.
- [6] Martinus I. 2004. Mekanisme dan Implementasi Keamanan pada Session Initiation Protocol (SIP). Bandung. Institut Teknologi Bandung.
- [7] Modul Praktikum Bengkel Internet dan Pemrograman Web. 2014. Universitas



Telkom : Prodi D3 Teknik Telekomunikasi

- [8] Modul Praktikum Teknik Switching. 2014. Bandung. Telkom University.
- [9] Raharja A. 2006. Session Initiation Protocol. Presentasi VoIP Rakyat.
- [10] Sunar Frihantono, Bimo. 2002. PHP dan MySQL untuk WEB. Yogyakarta : ANDI
- [11] SIP : Session Initiation Protocol, RFC 2543. pada Juni 2015, <https://www.ietf.org/rfc/rfc2543.txt>
- [12] Web RTC. Pada juni 2015. <http://webrtc.org>

