

**PERANCANGAN APLIKASI CLIENT UNTUK JARINGAN VOIP  
(VOICE OVER INTERNET PROTOCOL) BERBASIS ANDROID**

**Yasdinul Huda<sup>1</sup>  
Muhammad Adri<sup>2</sup>  
Yoharmen Arnov<sup>3</sup>**

**ABSTRACT**

*Communication is a necessity in the operation of an agency or company. Inexpensive communications using VOIP will provide for the expense of the agency or company. In communication required a reliable network system with the concept of client for the user of a system and server as a network manager used by client. VOIP requires a server using a PC with Ubuntu Linux 12.04 LTE or Linphone as a VOIP (Voice Over Internet Protocol) server and an asterisk for managing voice traffic. android software on the client side to be the interface in communicating with other clients designed using Android Studio with android smartphone at least 4.4.2 (kitkat). And requires internet connection via WIFI\_UNP as the transmission medium (link) between server and client. WIFI\_UNP is an internet medium used by students, lecturers, staff and employees of State University of Padang to support the learning process and add insight about the latest technology or science. While tethering hotspot is an internet media that comes from internet connection from Smartphone which is shared or distributed to other users for internet. This VOIP Client application provides two main functions: Calling and Chat. On Calling will be granted permissions to make voice calls by entering the client username to be contacted or VOIP number that has been registered. While Chat is a feature to perform realtime (in accordance with the actual time) text chat .*

**Keywords :** *Client Application, Server, VOIP, Ubuntu Linux, Android.*

---

<sup>1</sup> Dosen Universitas Negeri Padang

<sup>2</sup> Dosen Universitas Negeri Padang

<sup>3</sup> Jurusan Teknik Elektronika UNP

## INTI SARI

Komunikasi adalah suatu keharusan dalam operasi suatu agen atau perusahaan. Komunikasi murah menggunakan VOIP akan menyediakan biaya agensi atau perusahaan. Dalam komunikasi diperlukan suatu sistem jaringan yang handal dengan konsep client untuk pengguna suatu sistem dan server sebagai pengelola jaringan yang digunakan oleh klien. VOIP memerlukan server menggunakan PC dengan Ubuntu Linux 12.04 LTS atau Linphone sebagai server VOIP (Voice Over Internet Protocol) dan tanda bintang untuk mengatur lalu lintas suara. perangkat lunak android di sisi klien untuk menjadi antarmuka dalam berkomunikasi dengan klien lain yang dirancang menggunakan Android Studio dengan smartphone android setidaknya 4.4.2 (kitkat). Dan membutuhkan koneksi internet melalui WIFI\_UNP sebagai media transmisi (tautan) antara server dan klien. WIFI\_UNP adalah media internet yang digunakan oleh mahasiswa, dosen, staf dan karyawan Universitas Negeri Padang untuk mendukung proses pembelajaran dan menambah wawasan tentang teknologi atau sains terbaru. Sedangkan tethering hotspot adalah media internet yang berasal dari koneksi internet dari Smartphone yang dibagikan atau didistribusikan ke pengguna lain untuk internet. Aplikasi VOIP Client ini menyediakan dua fungsi utama: Memanggil dan Mengobrol. On Calling akan diberikan izin untuk melakukan panggilan suara dengan memasukkan nama pengguna klien untuk dihubungi atau nomor VOIP yang telah didaftarkan. Sedangkan Obrolan adalah fitur untuk melakukan obrolan teks secara realtime (sesuai waktu yang sebenarnya).

**Kata Kunci :** Aplikasi Klien, Server, VOIP, Ubuntu Linux, Android

## PENDAHULUAN

Dewasa ini jaringan komunikasi telah menjadi suatu kebutuhan yang cukup penting bagi manusia. Hampir di setiap perusahaan memerlukan alat komunikasi yang ditujukan untuk mempermudah kegiatan operasional sehari-hari perusahaan. Jaringan komunikasi ini merupakan infrastruktur penting yang harus diperhatikan dan dikelola kinerjanya dengan baik.

Agar dapat bekerja dengan baik, maka jaringan tersebut harus dikelola dengan benar, baik dari segi performa, topologi, maupun keamanannya. Adanya gangguan kecil dan kurangnya stabilitas pada jaringan komunikasi suatu perusahaan dapat menyebabkan tidak lancarnya arus komunikasi perusahaan sehingga dapat menyebabkan kerugian pada perusahaan tersebut.

Manfaat berkomunikasi dengan *voice* menggunakan jaringan data menurut Winarno Sugeng (2007 : 5), antara lain "*Efisiensi alokasi bandwidth, kemampuan untuk menggunakan metode kompresi suara, menekan biaya penggunaan, kemampuan menggunakan single interface, meningkatkan keandalan (reliability) jaringan komputer, dan termasuk menekan biaya operasional komunikasi hingga mendekati gratis (Rp. 0,-) jika menggunakan LAN*"

Hal ini menjadikan persaingan pada *provider* untuk menekan harga jual paket 4G untuk dapat dinikmati pelanggannya, maka solusi komunikasi dengan via VOIP akan sangat murah untuk digunakan dibandingkan dengan menelpon pada jaringan suara konvensional dengan pulsa dari kartu provider. VOIP tidak dikenakan roaming untuk menelpon ke beda negara karena hanya menggunakan paket data internet yang dibeli, kompresi suara pada VOIP sangat kecil sehingga dapat menekan tingginya quota yang dipakai dalam komunikasi suara karena ada empat unsur yang membentuk jaringan VOIP menurut Onno W. Purbo dari buku

VOIP Cikal Bakal "Telkom Rakyat", yaitu: ***User Agent, Proxy, Protocol dan CODEX (Coder-Decoder)***.

Jaringan komputer merupakan salah satu cara dari penerapan teknologi informasi. Karena pentingnya jaringan komputer, dewasa ini hampir seluruh perusahaan menggunakan media jaringan komputer dalam komunikasi datanya. Karena melalui jaringan komputer diijinkan pengaksesan atau komunikasi data secara bersama. Untuk memungkinkan komunikasi data diantara komputer dalam sebuah jaringan, diperlukan sebuah perangkat pengontrol jaringan seperti *server*. Pengaturan dan pengontrolan yang dilakukan *server* dalam jaringan meliputi kontrol hak akses terhadap suatu bentuk sumber informasi dan kontrol komunikasi data. Karena itu VOIP membutuhkan sebuah *server* menggunakan PC dengan Linux Ubuntu 12.04 LTE atau Linphone sebagai *server* VOIP (*Voice Over Internet Protocol*) dan asterisk untuk mengelola lalu lintas suara. *software* android di sisi *client* untuk menjadi interface dalam berkomunikasi dengan *client* lain yang dirancang menggunakan Android Studio dengan *smartphone* android minimal 4.4.2 (kitkat). Dan membutuhkan koneksi internet melalui WIFI\_UNP sebagai media transmisi (penghubung) antara *server* dan *client*. WIFI\_UNP merupakan media internet yang digunakan mahasiswa, dosen, staff serta karyawan Universitas Negeri Padang untuk menunjang proses pembelajaran dan menambah wawasan tentang teknologi atau ilmu pengetahuan terbaru. Sedangkan tethering hotspot merupakan media internet yang berasal dari sambungan internet dari *Smartphone* yang di-*sharing* atau dibagikan kepada pengguna lain untuk internet.

Tujuan dari penelitian ini adalah merancang sebuah aplikasi *client* untuk komunikasi melalui panggilan suara dan chat. Sistem yang

dikembangkan mengikuti arsitektur *client-server*, dengan *client* terdiri dari aplikasi berbasis android. Aplikasi *mobile* dibangun menggunakan editor Android Studio v.3.1.0.

## **PENDEKATAN PEMECAHAN MASALAH**

### **Perancangan**

Sistem yang akan dibangun memakai metode *client-server* dalam arsitekturnya, arsitektur dalam pada sistem ini terdiri 3 komponen utama yaitu, android *application*, *server application*, dan *database server*. Android *client application* merupakan aplikasi yang digunakan oleh *user* sebagai alat komunikasi VOIP. Pada android *client application* terdapat fitur untuk memanggil dan *chat* dengan *user* lain menggunakan DNS (*Domain Name Server*) yang telah terdaftar pada server.

*Server* VOIP dibangun menggunakan Sistem Operasi Linux Ubuntu 12.04. Dan *software* Asterisk untuk mengelola *user* VOIP yang akan berkomunikasi. *Database server* menggunakan SQL Lite yang berfungsi sebagai media penampung data dari keseluruhan data yang diolah di *web admin*.

1) Linux Ubuntu 12.04LTS  
Canonical, perusahaan dibalik distro Linux populer Ubuntu telah meluncurkan edisi LTS (*Long Term Support*) versi 12.04 dengan nama kode “Precise Pangolin” tepat waktu sebagaimana dijadwalkan sebelumnya, pada tanggal 26 April 2012 lalu. Edisi khusus LTS yang diluncurkan dua tahun sekali menjanjikan dukungan perbaikan selama 5 tahun baik untuk *server* maupun *desktop*. Pada versi-versi LTS sebelumnya (10.04 dan 8.04), hanya dukungan 5 tahun hanya berlaku untuk *server*, sedangkan *desktop* didukung selama 3 tahun.

2) Asterisk  
Asterisk adalah implementasi perangkat lunak dari “*telephone private branch exchange (PBX)*”, diciptakan pada tahun 1999 oleh Mark Spencer dari Digium. Seperti PBX lainnya, dimungkinkan memasang pesawat telepon dan melakukan panggilan ke satu dengan lainnya, termasuk tersambung ke layanan telepon pribadi dan publik, termasuk layanan jaringan telepon umum (PSTN) dan *Voice Over Internet Protocol (VOIP)*. Nama Asterisk berasal dari \* (tanda bintang). Asterisk dirilis menganut model lisensi ganda, menggunakan GNU/GPL sebagai lisensi perangkat lunak bebas dan lisensi perangkat lunak berpaten untuk mengizinkan pemegang lisensi untuk mendistribusikan komponen sistem propietari yang tidak perlu dipublikasikan.

3) Android Studio  
Android Studio menurut Alfa Satyaputra dan Eva Maulina Aritonang (2016 : 63) adalah “*IDE resmi untuk membangun aplikasi Android berdasarkan IntelliJ IDEA*”. IntelliJ IDEA sendiri adalah Java *Integrated Development Environment (IDE)* yang dikembangkan oleh JetBrains, untuk mengembangkan perangkat lunak (*software*) komputer. IntelliJ IDEA berfungsi dalam membantu Anda dalam dunia per-*coding*-an baik dari segi navigasi, penyokong produktivitas, hingga *code editor* yang cerdas. Android Studio dibangun dengan tujuan mempercepat proses pembangunan maupun pengembangan aplikasi Android yang berkualitas tinggi untuk setiap *device* Android. Google

mengakuisisi Android Inc pada bulan Agustus 2005 sebagai anak perusahaan yang sepenuhnya dimiliki dari tujuan utama Google Inc. Google adalah untuk menyediakan *platform* terbuka penuh, yang didukung oleh teknologi Google, baik untuk pengguna dan pengembang aplikasi. Pada bulan November 2007, Open Handset Alliance didirikan sebagai konsorsium untuk mengembangkan standar terbuka untuk perangkat *mobile*. Open Handset Alliance memulai perjalanannya dengan mengumumkan *platform* Android. Dalam waktu kurang dari satu tahun, anggota baru mulai bergabung konsorsium ini (Nazruddin Safaat, 2015).

4) SQLite

Aplikasi Android umumnya perlu menyimpan dan mengolah data-data yang beragam jenisnya. Contohnya, sebuah game RPG (*Role Playing Game*) yang menyimpan data-data karakter, musuh, peta dunia, inventori, *magic*, *items* dan sebagainya. Atau aplikasi yang menyimpan data diri berupa data nama, alamat, nomor telepon, *e-mail*, dan sebagainya. Untuk menyimpan data-data tersebut secara aman, rapi dan terstruktur, Anda dapat menggunakan fitur *database* yang sudah tersedia dalam Android, yaitu SQLite.

5) Linphone WebServer

Linphone adalah server VOIP yang memungkinkan pengguna untuk memanggil teman-teman mereka dari mana saja di dunia tanpa membayar lebih bahwa biaya sebenarnya dari koneksi internet mereka.

Server ini memungkinkan pengguna untuk melakukan konfigurasi panggilan suara,

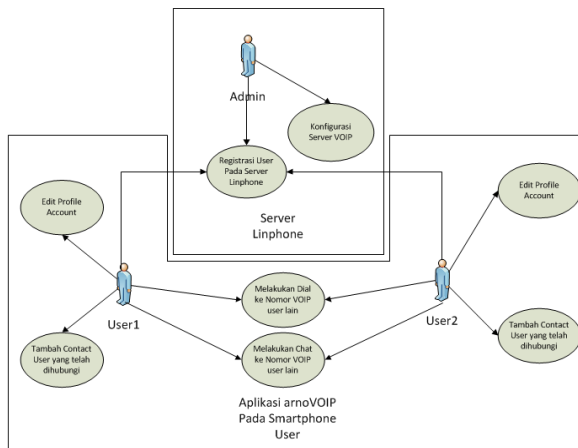
memulai konferensi video, serta teks teman-teman mereka menggunakan (*instant messaging*) protokol IM. Untuk sistem telepon internet, aplikasi yang menggunakan standar terbuka dan protokol SIP terkenal.

Untuk memanggil seseorang, pengguna harus menyediakan SIP (*Session Initiation Protocol*) URL yang terlihat seperti alamat *email*. Anda harus diketahui SIP URL sebelum memulai panggilan telepon, tetapi Anda juga dapat menggunakan alamat IP standar.

Dalam istilah yang lebih teknis, aplikasi dilengkapi dengan SIP UA (*User Agent*) yang sesuai dengan RFC 3261, mendukung protokol IPv6, mendukung SIP / UDP, SIP / TCP dan protokol SIP / TLS, serta *digest* otentikasi. Mendukung beberapa panggilan dan SIP proxy. Selain itu, aplikasi ini mendukung beberapa panggilan pada waktu yang sama dan beberapa *proxy* SIP (*proxy outbound*, registrar, dan lain-lain). Hal ini juga termasuk dukungan pemberitahuan pengiriman untuk komponen *instant messaging*. Fitur manajemen panggilan standar disediakan, termasuk berpegang pada dengan musik, *transfer* panggilan atau melanjutkan DTMF (nada telepon) dukungan juga disertakan, berdasarkan RFC 2833 atau spesifikasi SIP INFO. Ini adalah sebuah *server* yang *ultra portable*, yang telah berhasil diuji dengan Linux, Windows, MacOS X, Android, iOS, BlackBerry dan WebOS sistem operasi. 64-bit, 32-bit dan ARM arsitektur yang didukung. Dapat disimpulkan bahwa, Linphone terbukti menjadi alternatif yang sangat baik untuk aplikasi

telepon Internet (VOIP). Hal ini secara aktif dikembangkan, berjalan pada berbagai platform dan menyediakan koleksi *plugin codec*.

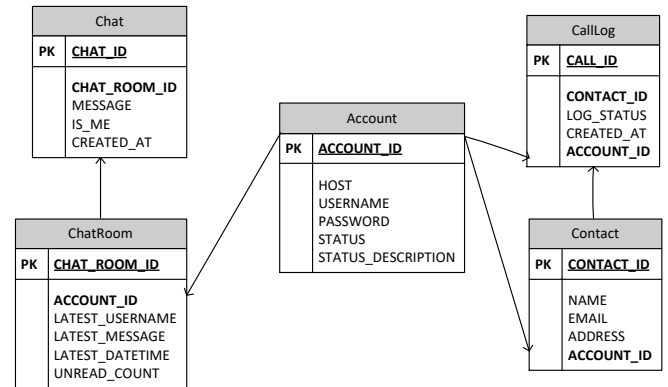
Use Case pada diagram Arsitektur *Client-Server* dijelaskan pada Gambar 1.



Gambar 1. Use Case Diagram Arsitektur Client-Server

Gambar 1 menggambarkan bahwa *server* Linphone berbasis *web* digunakan untuk registrasi *user* dan konfigurasi *server* VOIP dan aktor yang mengelola *server* Linphone adalah Admin. Pada Aplikasi VOIP yang diberi nama "arnoVoip" ini digunakan oleh 2 *user* yang berbeda untuk berkomunikasi suara atau pesan teks. Setiap *user* dapat melakukan *edit profile account* dan tambah *contact user* yang telah dihubungi.

*Class diagram* menggambarkan keadaan sistem (attribut), dan memberikan pelayanan untuk menyelesaikan keadaan tersebut (metoda). Pada gambar *class diagram* dari sistem Aplikasi *client* VOIP. Untuk *Class Diagram database* dari aplikasi dijelaskan pada Gambar 2.

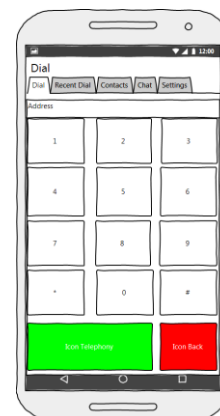


Gambar 2. Class Diagram

Gambar 2 menunjukkan bahwa tabel yang digunakan pada database aplikasi sebanyak 5 tabel yaitu :

- Chat*, untuk informasi dari *chat* yang masuk dari *user* lain.
- ChatRoom*, untuk data *personal chat* dengan *user* lain.
- Account*, untuk data aktivasi akun dari aplikasi
- CallLog*, untuk informasi dari semua panggilan
- Contacts*, untuk data kontak yang akan dihubungi.

Perancangan antarmuka dibutuhkan untuk mewakili keadaan sebenarnya dari aplikasi yang akan dibangun. Berikut tampilan antar muka aplikasi VOIP yang digunakan pada sisi *client* Pada Gambar 3.



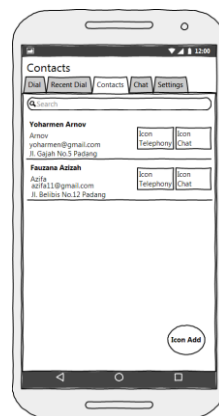
Gambar 3. Disain Tampilan Menu Dial

Disain tampilan menu *Dial* pada Gambar 3 merupakan tampilan awal aplikasi yang dilihat oleh *user*. *User* menekan tombol nomor VOIP atau memasukkan username yang akan dihubungi kemudian *user* menekan tombol hijau untuk menelpon. Tombol merah untuk menghapus nomor atau username yang telah dimasukkan kedalam *textbox address*. Untuk Menu Recent Dial *user* cukup menekan tab menu Recent Dial seperti Gambar 4.



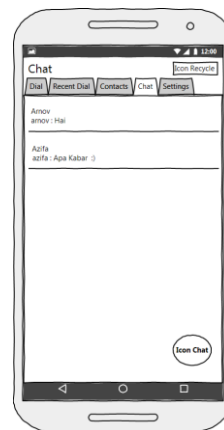
Gambar 4. Disain Tampilan Menu *Recent Dial*

Gambar 4 merupakan Menu *Recent Dial* adalah menu yang akan menampilkan seluruh panggilan yang telah masuk, keluar atau tidak terjawab. Menu *Contacts* dapat dijelaskan pada Gambar 5.



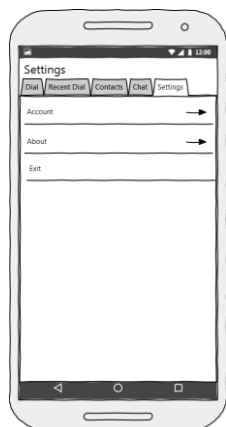
Gambar 5. Disain Tampilan Menu *Contacts*

Setelah memilih menu tab *Contacts* maka akan muncul *list* kontak yang telah disimpan pada *database* aplikasi seperti Gambar 5. Pada halaman ini *user* juga dapat langsung melakukan panggilan suara atau *chat* dengan menekan *icon* yang ada di sebelah kanan nama kontak tersebut. Untuk menu *Chat* dapat dijelaskan pada Gambar 6.



Gambar 6. Disain Tampilan Menu *Chat*

Halaman *chat* pada Gambar 6 digunakan *user* untuk melakukan obrolan teks dengan *user* lain. Untuk memulai chat maka *user* harus menekan tombol *new chat* yang ada pada pojok kanan bawah dari aplikasi ini. Untuk tampilan menu *Settings* dilihat pada Gambar 7.



**Gambar 7.** Disain Tampilan Menu *Settings*

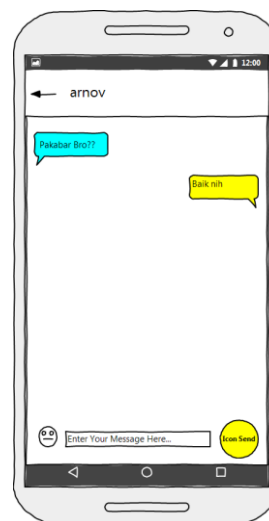
Menu *Settings* pada Gambar 7 terdapat 3 pilihan sub menu yaitu: *Account*, *About*, dan *Exit*. *Account* digunakan untuk mengaktifkan *account* yang akan digunakan pada aplikasi ini. *About* adalah informasi mengenai aplikasi ini. Untuk mengetahui *user* sedang melakukan panggilan VOIP maka akan muncul halaman *Calling* seperti Gambar 8.



**Gambar 8.** Disain Tampilan *Calling*

Halaman *Calling* pada Gambar 8 tampil ketika *user* melakukan panggilan VOIP, untuk mengakhiri panggilan *user* harus menekan tombol merah.

Tampilan *Personal Chat* dijelaskan pada Gambar 9.



**Gambar 9.** Disain Tampilan *Chat*

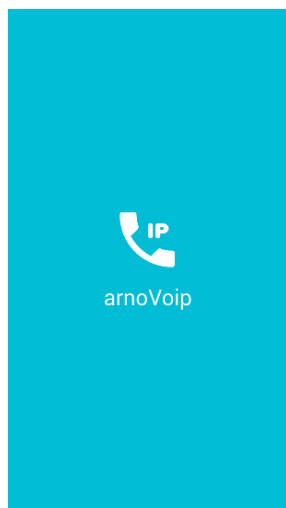
Halaman *chat* pada Gambar 9 tampil ketika *user* sudah memasukkan username yang akan di-*chat*. Halaman ini adalah *chat* yang real time. *Chat* berwarna kuning adalah *chat* dari *user* pengguna aplikasi sedangkan *chat* berwarna biru adalah *chat* lawan bicaranya.

## HASIL DAN PEMBAHASAN

### Hasil Rancangan

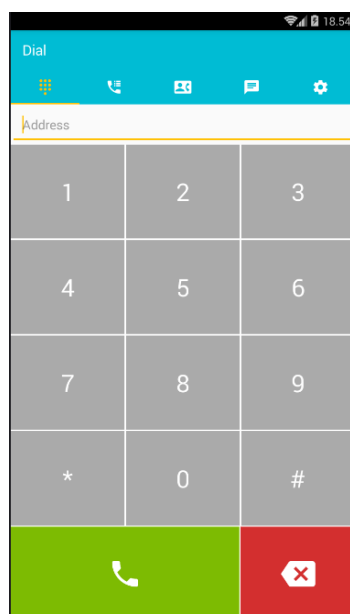
Hasil rancangan ini membahas tampilan antarmuka aplikasi *client* untuk jaringan VOIP. Terdapat beberapa hasil rancangan antarmuka dalam aplikasi untuk melakukan panggilan suara dan *chat*, berikut pembahasan tampilan antarmuka tersebut. *Splash screen* adalah tampilan pertama ketika aplikasi dibuka. Berikut tampilan *splash screen* pada Gambar 10





Gambar 10. Tampilan *Splash Screen*

Halaman menu utama tampil setelah tampilan *splash screen* selesai seperti Gambar 10. Pada menu utama menampilkan beberapa menu yaitu *Dial*, *Recent Dial*, *Contact*, *Chat* dan *Setting* yang akan dijelaskan pada poin berikutnya. Berikut tampilan menu utama pada aplikasi *client* untuk jaringan VOIP. Tampilan menu utama dijelaskan pada Gambar 11.

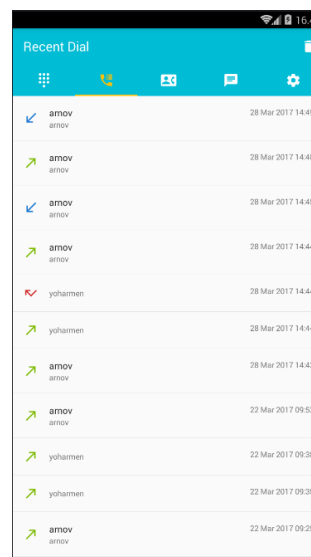


Gambar 11. Tampilan Menu Utama

Gambar 11 merupakan tampilan menu utama yang ada diaplikasi setelah *splashscreen*

berjalan. Menu utama berfungsi untuk memanggil nomor VOIP atau *id account user* yang akan dihubungi.

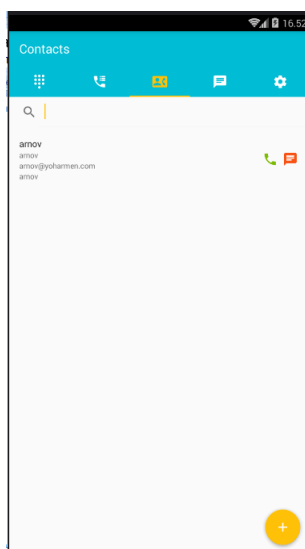
Halaman *Recent Dial* berisikan tentang informasi seluruh panggilan yang masuk, keluar dan tidak terjawab. Halaman menu *recent dial* dapat dilihat pada Gambar 12 berikut ini :



Gambar 12. Tampilan *Recent Dial*

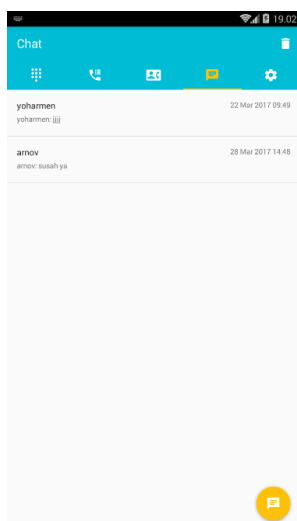
Halaman *Recent Dial* pada Gambar 12 untuk melihat informasi seluruh data panggilan VOIP.

Halaman *Contact* berisikan informasi dari kontak yang telah disimpan pada aplikasi ini. Halaman menu *contact* dapat dilihat pada Gambar 13 berikut ini :



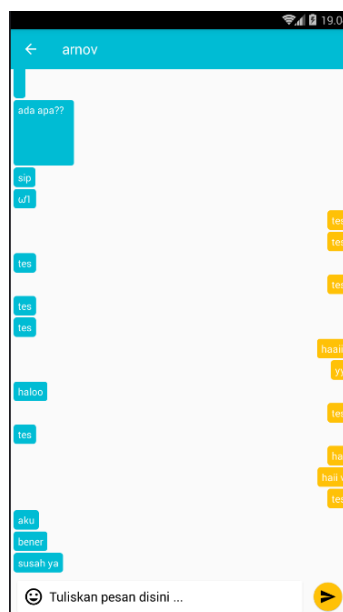
Gambar 13. Tampilan Contact

Halaman menu *chat* akan menampilkan informasi dari *chat* yang telah dilakukan oleh *user* pada aplikasi ini. Halaman *chat* dapat dilihat pada Gambar 14 dibawah ini :



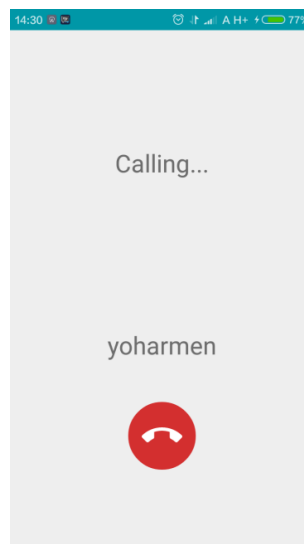
Gambar 14. Tampilan Chat

Antarmuka dari halaman *chat personal* merupakan halaman yang akan muncul ketika *user* telah memasukkan *username* kontak yang dituju untuk *chat* dapat dilihat pada Gambar 15.



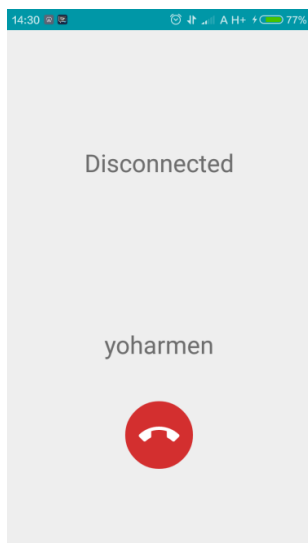
Gambar 15. Halaman Chat Personal

Antarmuka halaman untuk melakukan panggilan VOIP akan muncul setelah *user* memasukkan nomor VOIP atau username yang akan dihubungi pada *menu dial* dijelaskan pada Gambar 16.



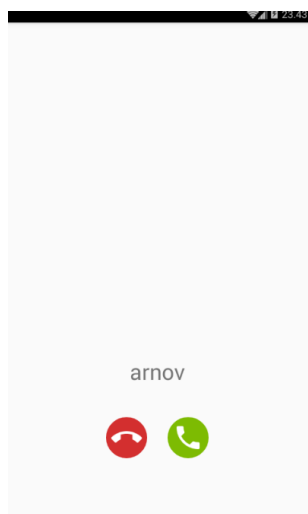
Gambar 16. Tampilan panggilan VOIP

Jika *user* akan mengakhiri panggilan VOIP, maka *user* cukup menekan tombol *icon* telepon yang berwarna merah pada Gambar 17. Maka akan muncul halaman *Call End* seperti dibawah ini.



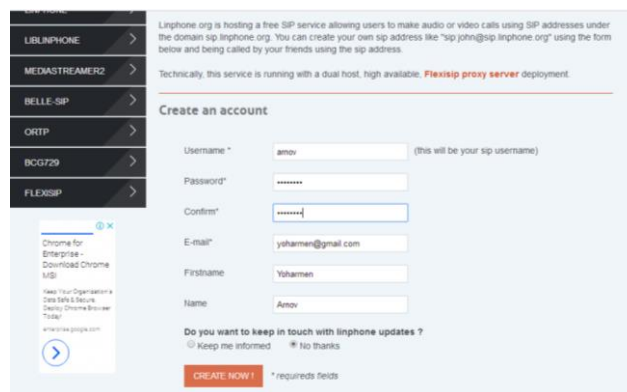
**Gambar 17. Tampilan Call End**

Halaman yang muncul pada penerima telepon yang dipanggil akan muncul halaman panggilan masuk seperti Gambar 18 dibawah. Untuk menerima telepon maka *user* harus menekan tombol hijau dan untuk menolak maka *user* harus menekan tombol merah.



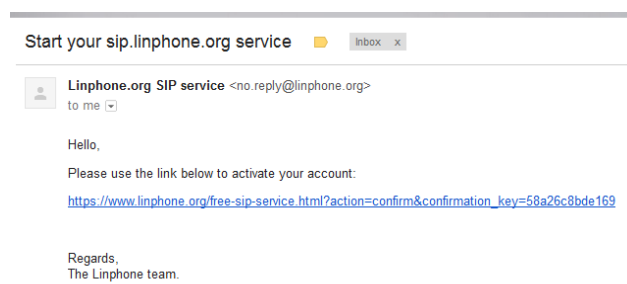
**Gambar 18. Tampilan panggilan masuk**

Halaman *webserver* dari Linphone yang berfungsi untuk mengelola *user* yang akan menggunakan komunikasi VOIP. Halaman *Registrasi User* dijelaskan pada Gambar 19 berikut:



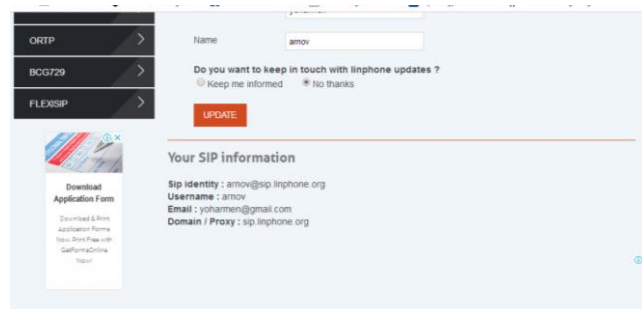
**Gambar 19. Halaman Registrasi User**

Antarmuka halaman *activation account* adalah halaman antarmuka yang muncul pada *email user* yang telah melakukan registrasi pada Linphone *webserver* digambarkan seperti Gambar 20 berikut :



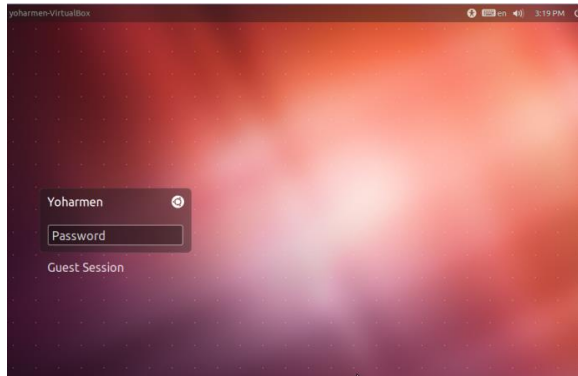
**Gambar 20. Tampilan Halaman Activation Account**

Setelah *user* menekan *link* aktivasi pada *email*-nya maka *browser* akan mengarahkan ke halaman informasi *detail* aktivasi pada Linphone *Webserver*. Tampilan *account* yang sudah aktif dapat dilihat pada Gambar 21.



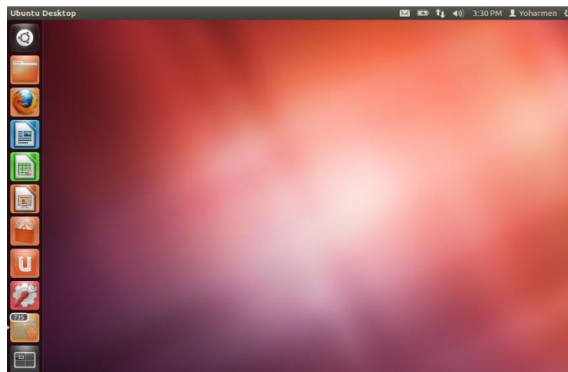
**Gambar 21. Tampilan account yang sudah aktif**

Tampilan *Server* Ubuntu 12.04 yang akan mengelola konfigurasi VOIP pada aplikasi. Tampilan *login* muncul setelah proses *booting server* selesai seperti Gambar 22.



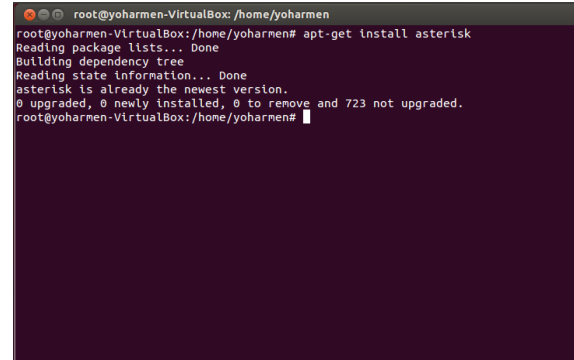
Gambar 22. Tampilan *login* Ubuntu 12.04

Tampilan *desktop* dari Ubuntu 12.04 yang berisikan aplikasi dan *tool* dari sistem operasi Linux tersebut pada Gambar 23 berikut:



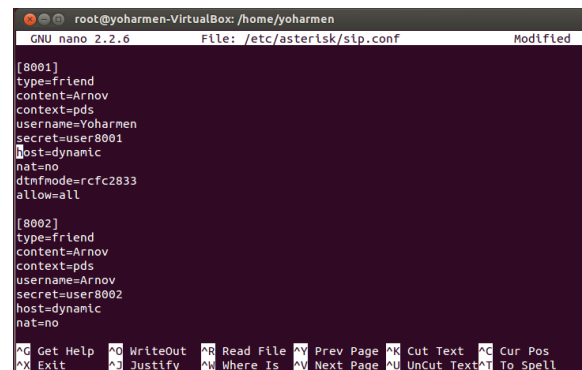
Gambar 23. Tampilan *Desktop* Ubuntu 12.04

Tampilan instalasi asterisk dengan mengetikkan perintah `apk-get install asterisk` pada terminal linux seperti Gambar 24 dibawah ini.



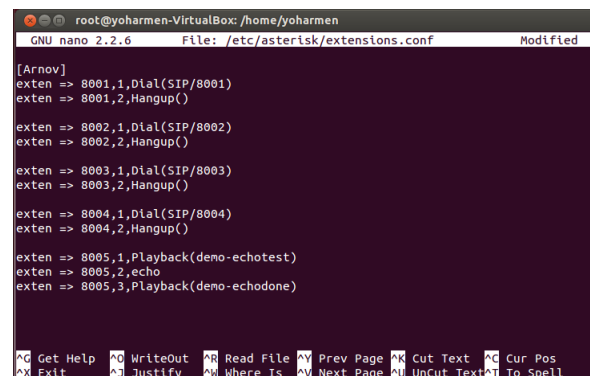
Gambar 24. Tampilan instalasi asterisk

Tampilan konfigurasi `sip.conf` untuk *detail* dari *user* VOIP dapat dijelaskan pada Gambar 25.



Gambar 25. Tampilan konfigurasi `sip.conf`

Tampilan konfigurasi `extensions.conf` untuk *detail* dari *user* VOIP dijelaskan pada Gambar 26.



Gambar 26. Tampilan konfigurasi `extensions.conf`

## KESIMPULAN

Adapun kesimpulan yang didapat dari perancangan Aplikasi

Komunikasi VOIP (arnoVoip) pada *Smartphone* berbasis Android adalah sebagai berikut:

1. Dengan pemanfaatan *smartphone* berbasis android dihasilkan aplikasi untuk komunikasi suara dan pesan melalui IP (Internet Protocol) serta DNS (*Domain Name Server*) dengan teknologi VOIP (*Voice Over Internet Protocol*).
2. Dengan *Server* Internal dapat meminimalisir anggaran biaya berkomunikasi bila dibandingkan dengan komunikasi telepon konvensional (telepon rumah/*dial-up*).
3. Aplikasi arnoVoip dapat dipakai oleh banyak pengguna (*multiuser*).

Adapun saran dari penulis setelah merancang Aplikasi komunikasi VOIP (arnoVoip), antara lain:

1. Diharapkan aplikasi arnoVoip ini dapat berkontribusi dan dikembangkan untuk memudahkan komunikasi suara dan pesan menjadi lebih baik.
2. Instansi dan perusahaan dapat meminimalisir pengeluaran biaya komunikasi dengan menggunakan aplikasi ini.

#### DAFTAR PUSTAKA

- [1] Alfa Satyaputra dan Eva Maulina Aritonang. 2016. *Let's Build Your Android Apps with Android Studio*. Jakarta: Jasakom.
- [2] Purbo, Onno W. 2007. *VoIP Cikal Bakal "Telkom Rakyat"*. Yogyakarta: Infokomputer.
- [3] Safaat H, Nazruddin. 2015. *Pemograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Bandung: Informatika.
- [4] Sugeng, Winarno. 2007. *Membangun Telepon Berbasis Voip*. Bandung: Informatika.

- [5] Universitas Negeri Padang. 2010. *Buku Panduan Penulisan Tugas Akhir/Skripsi Universitas Negeri Padang*. Padang: Universitas Negeri Padang.