

Implementasi Live Audio Streaming Menggunakan Raspberry Pi

Hudaya¹Gita Indah Hapsari²Giva Andriana Mutiara³

¹mulkihudaya@gmail.com ²gitaindahhapsari@tass.telkomuniversity.ac.id ³giva.andriana@tass.telkomuniversity.ac.id

Abstrak

Penelitian ini adalah penelitian perancangan *server live audio streaming* dengan beberapa fitur tambahan lainnya menggunakan *Raspberry Pi*. Diawali dengan instalasi aplikasi-aplikasi terkait untuk menunjang *server live audio streaming*, *mp3 streaming*, dan *user feedback* berupa *shoutbox* beserta konfigurasinya. Kemudian mengimplementasikan ke dalam aplikasi web untuk diakses oleh user dan merubah tampilan default halaman web sesuai kebutuhan aplikasi dan kenyamanan user dalam mengaksesnya. Dari hasil perancangan hasil pengujian tersebut dapat diambil kesimpulan sebagai berikut : Sistem pada *Raspberry Pi* bisa memanfaatkan aplikasi aplikasi yang tersedia pada *debian linux* seperti *icecast2*, halaman default untuk akses aplikasi oleh user dapat diubah sesuai dengan kehendak berdasarkan kebutuhan menggunakan aplikasi desain web pada umumnya serta semua aplikasi web untuk Live Audio Streaming dapat berjalan dengan baik.

Kata kunci: Raspberry Pi, Aplikasi Web, Broadcast, Live Audio Streaming, Shoutbox

Abstract

This research is design of live audio streaming server with several other additional features using the Raspberry Pi. Starting with the installation of applications related to server support live audio streaming, mp3 streaming, and VoIP and configurations. Then implement into a web application for access by the user and change the default display of web pages according to the needs of applications and user convenience in accessing it. From the design of the design results of the test results can be concluded as follows: System on Raspberry Pi can utilize application applications available on debian linux like darkice and Icecast, the default page to access the application by the user can be changed in accordance with the will of the need to use applications based on web design generally and all of web applications for Live Audio Streaming can be used properly.

Keywords: Raspberry Pi, Web Application, Broadcast, Live Audio Streaming, Shoutbox

1. Latar Belakang

Teknologi yang terus berkembang dengan pesat membantu pekerjaan kita terselesaikan dengan mudah, namun dengan ukuran *hardware* yang terlalu besar membatasi ruangnya. *Raspberry Pi* yang saat ini sedang marak di Indonesia adalah salah satu komponen *hardware* utama yang hadir dengan ukuran yang sangat kecil dengan kualitas hampir sama dan fitur yang tidak kalah dengan komponen sejenisnya. *Raspberry Pi* adalah komponen *motherboard* mini berukuran sebesar kartu kredit dengan sistem operasi *Raspbian* yang berbasis *Debian GNU* atau *Linux* sehingga tidaklah sulit dalam mengoperasikannya.

Keterbatasan dari segi ketersediaan ruang untuk perancangan suatu sistem server menjadi salah satu kendala bagi ruang server saat ini. Keterbatasan

ini akhirnya memunculkan sebuah ide, yaitu bagaimana seseorang merancang sebuah server dengan komponen-komponen hardware yang berukuran jauh lebih kecil dibandingkan dengan komponen-komponen yang biasa diterapkan untuk membangun sebuah server. Live audio streaming menggunakan *Raspberry Pi* dibangun untuk mengatasi keterbatasan ini, dimana komponen utamanya yaitu *Raspberry Pi* sebagai *mini embedded server*.

Melihat dari pesatnya perkembangan teknologi *Raspberry Pi*, namun belum adanya aplikasi untuk *live audio streaming* berbasis *Raspberry Pi*, maka dari itu proyek akhir ini penulis mengambil judul "*Implementasi Live Audio Streaming Menggunakan Raspberry Pi*".

2. Dasar Teori

2.1. Streaming

Sarana komunikasi tidak hanya dalam bentuk berbicara lisan saja, tetapi dapat juga dalam bentuk pertukaran *file* atau data juga termasuk bentuk komunikasi. Sebagai contoh sarana komunikasi tersebut adalah *Streaming*. *Streaming* merupakan istilah dimana sebuah *file video* maupun *audio* dapat dimainkan tanpa terlebih dahulu di-*download*. *Streaming* juga dapat diartikan teknik yang digunakan untuk melakukan transfer data sehingga dapat diproses secara tetap dan berlanjut. Teknologi streaming berkembang sesuai dengan perkembangan internet, dimana kebanyakan user internet masih belum memiliki koneksi *broadband* untuk mengunduh file multimedia berukuran besar dengan cepat.

Konsep *streaming* saat ini banyak sekali digunakan oleh perusahaan-perusahaan pertelevisian atau perusahaan-perusahaan yang berhubungan dengan penyiaran. Institusi-institusi pemerintahan pun tidak kalah untuk memanfaatkan metode tersebut untuk keperluan pekerjaan mereka. *Streaming* identik dengan *realtime*. Namun tidak dapat dipungkiri setiap media *streaming* memiliki kendala *delay*. *Delay* adalah jumlah waktu yang tertinggal dengan waktu *real* atau waktu nyata. Mengapa *delay* terjadi pada proses *streaming* tersebut ? karena *video streaming* merupakan metode *transfer data audio* maupun *video* sehingga ada proses-proses tertentu seperti proses *encoder transfer file* yang akhirnya menghasilkan *delay*. Walaupun demikian, *delay* pada *video streaming* tidak memakan waktu yang lama, sehingga pemanfaatannya lebih banyak digunakan dibandingkan dengan metode *transfer audio* ataupun *video via on demand* dan *via download*

2.2. Radio Streaming

Radio Streaming adalah media *audio* yang dapat didengarkan melalui jaringan *internet*. Radio

streaming ini juga disebut oleh banyak orang sebagai radio *online*. *Radio streaming* pada prinsipnya adalah *file audio* yang dikirim dari komputer sumber streaming ke server secara terus menerus dimana pendengar *radio streaming* dapat mengakses atau mendengarkan *file streaming* yang berada di *server* tersebut melalui jaringan *internet*.

Hal ini dimaksudkan agar pendengar dapat mengakses secara *real time* tanpa menunggu *file* terkirim secara utuh dari komputer sumber *streaming* ke *server* dan dari *server* ke komputer pendengar, meskipun terdapat jeda waktu karena proses *streaming* itu.

2.3. Konsep Streaming

Adapun konsep utama dari proses menerima *stream* data yaitu :

1. Download

Pada penerimaan dengan cara *download*, akses file dilakukan dengan cara melakukan *download* terlebih dahulu suatu *file* dari *server*. Penggunaan ini mengharuskan keseluruhan file diterima secara lengkap oleh *client*.

2. Streaming

Pada penerimaan *video* secara *streaming*, pengguna dapat melihat atau mengakses suatu *file* multimedia hampir bersamaan ketika file tersebut mulai diterima. Penggunaan cara ini mengharuskan pengiriman suatu *file* multimedia ke pengguna secara konstan.

3. Progressive download

Adalah suatu metode *hybrid* yang merupakan hasil penggabungan antara metode *download* dan *streaming*, dimana *file* yang sedang diakses dapat diterima dengan cara *download* sehingga *player* yang ada pada pengguna sudah

menampilkan *file* tersebut sejak sebagian dari *file* tersebut diterima walaupun *file* tersebut belum diterima secara lengkap.

2.4. Metode Transmisi Data Streaming

Adapun metode transmisi *streaming* yaitu :

1. Transmisi *unicast*

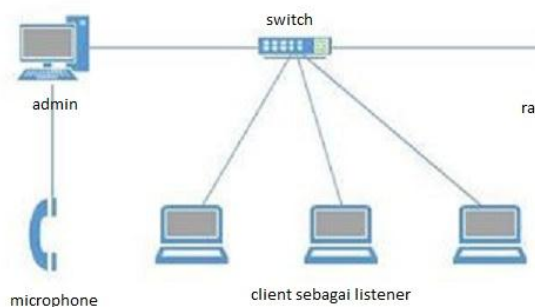
Transmisi *unicast* adalah metode koneksi secara langsung antara *user* dan *server*. Transmisi *unicast* bersifat *end to end*, dimana setiap *client* mendapatkan stream data yang berbeda. Meskipun data atau file yang distream sama, namun data yang dialirkan melalui jaringan akan disalin dan diterima oleh setiap client berbeda, sehingga menguras bandwidth.

2. Transmisi *multicast*

Transmisi *multicast* adalah metode distribusi dari satu sumber untuk banyak pengguna, atau dari satu grup untuk banyak pengguna. Tidak ada koneksi langsung antara *user* dan *server*. Metode transmisi *multicast* sesuai diterapkan pada proses *streaming* karena *bandwidth* yang diproses sama, dimana setiap user yang mengakses suatu *file* secara bersamaan dari *server*.

2.5. Skema Permodelan Live Audio Streaming

Adapun skema permodelan untuk *Live Audio Streaming* adalah sebagai berikut :



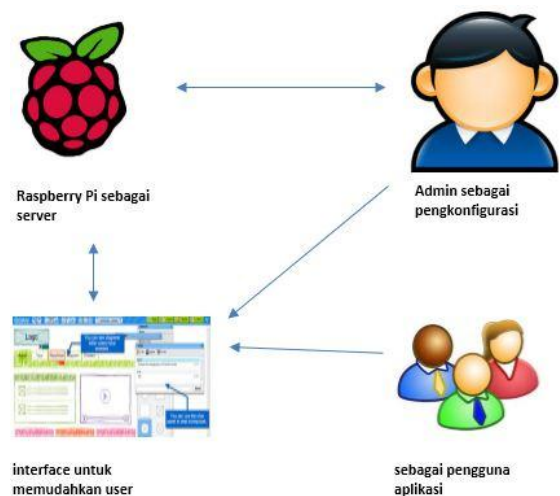
Gambar 1 Konsep Sistem Live Audio Streaming

Dari gambar skema diatas, dapat dilihat proses cara kerja *streaming*, yaitu pertama kita memasukkan sumber *audio* ke *microphone*, lalu diinputkan kedalam *pc* yang telah disetting didalamnya aplikasi *sam broadcast*, kemudian *server broadcast* akan melakukan *broadcasting data audio* tersebut secara bertahap dan terus menerus ke *client* yang membuka halaman *web live streamingnya*.

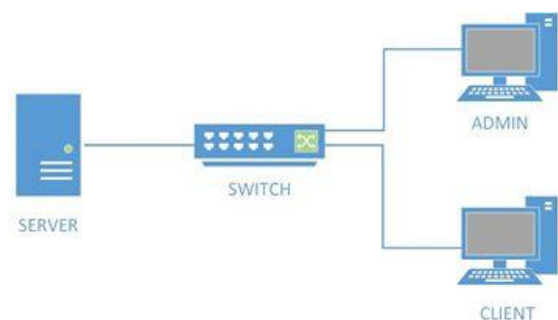
3. Analisis dan Perancangan

3.1. Gambaran Sistem Live Audio Streaming

Aplikasi *web live audio streaming* menggunakan *raspberrypi* ini merupakan sebuah aplikasi *broadcast audio* secara *live* melalui *local area network (LAN)*. Aplikasi ini memiliki tiga fitur utama, yaitu *live audio streaming*, *user feed back* berupa *shoutbox*, dan *mp3 streaming*.



Gambar 2 Konsep Sistem Live Audio Streaming



Gambar 3 Topologi Jaringan

3.2. Kebutuhan Perangkat Keras

Dalam pengerjaan proyek akhir ini, digunakan perangkat keras dengan spesifikasi sebagai berikut :

Tabel 1 Perangkat Keras

Jenis	Jumlah	Keterangan
Raspberry Pi	1	Quad Core
SD Card	1	16 GB
PC Admin	1	Intel Corei3; 4 GB DDR3; 500GB HDD;
PC Client	1	Intel Corei3; 2 GB DDR3; 500GB HDD;
Switch	1	8 port
Kabel Utp	3	

3.3. Kebutuhan Perangkat Lunak

Dalam pengerjaan proyek akhir ini, digunakan perangkat lunak dengan spesifikasi sebagai berikut :

Tabel 2 Perangkat Lunak

Jenis	Versi	Keterangan
Icecast	2.3.2	Aplikasi server streaming
SAM Broadcast	2013.6	Aplikasi broadcast Audio
Mysql server	5.6.24	Aplikasi database
GNUMP3d	3.0	Aplikasi streaming mp3
Apache	2	Aplikasi web server
Bind9	-	Aplikasi dns server

3.4. Langkah Pengerjaan

Adapun tahap pengerjaan proyek akhir ini adalah :

1. Melakukan konfigurasi jaringan.
 - a. *Server* : 192.168.1.1
 - b. *Admin* : 192.168.1.2
 - c. *Client 1* : 192.168.1.3
 - d. *Client 2* : 192.168.1.4
 - e. *Client 3* : 192.168.1.5
2. Melakukan konfigurasi *web server*.
3. Melakukan konfigurasi *dns* untuk *domain web*.
4. Melakukan konfigurasi tampilan *web*.
5. Melakukan konfigurasi *streaming server*.
6. Melakukan konfigurasi aplikasi *broadcast*.
7. Melakukan konfigurasi *shoutbox*.
8. Melakukan konfigurasi *mp3 streaming*.
9. Melakukan dokumentasi terhadap konfigurasi-konfigurasi tersebut.

3.5. Rencana Pengujian

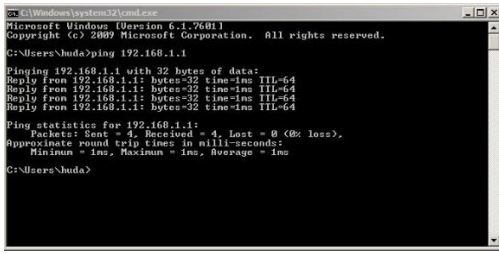
Pengujian yang dilakukan meliputi :

1. Dilakukan proses ping antar jaringan internal sehingga terhubung satu dengan yang lain.
2. *Client* melakukan akses *live audio streaming*.
3. *Client* melakukan akses *shoutbox*.
4. *Client* melakukan akses *mp3 streaming*.

4. Pengujian

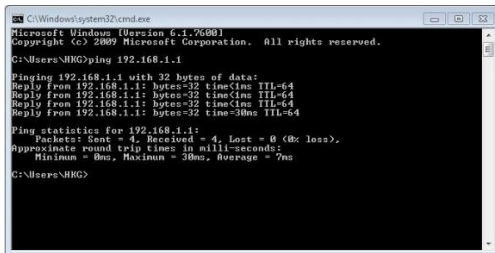
Adapun pengujian dari proyek akhir ini sebagai berikut.

1. Ping dari Admin ke server



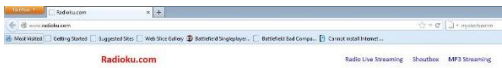
Gambar 4 Ping Admin - Server Raspberry pi

2. Ping dari Client ke Server



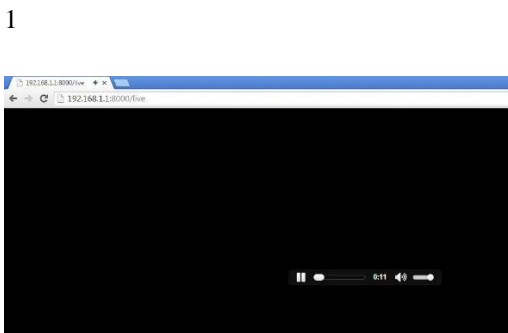
Gambar 5 Ping Client - Server Raspberry pi

3. Akses web halaman utama oleh client



Gambar 6 Tampilan Halaman Utama

4. Akses web live audio streaming oleh client



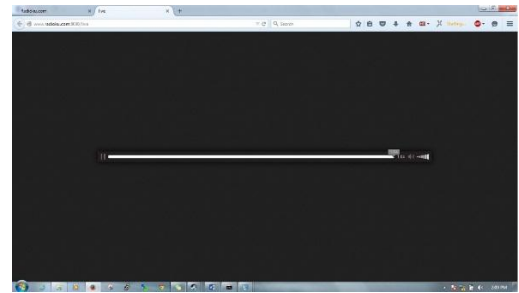
Gambar 7 Tampilan Live Audio Streaming client 1

5. Akses web live audio streaming oleh client



Gambar 8 Tampilan Live Audio Streaming client 2

6. Akses web live audio streaming oleh client



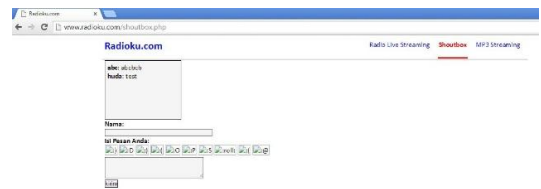
Gambar 9 Tampilan Live Audio Streaming client 3

Berdasarkan pengujian akses oleh ketiga client diatas, dapat dilihat bahwa setelah admin mem-broadcast audio terdapat delay 2 detik untuk client ke-1 serta bersautan untuk client ke-2 dan client ke-3.

Tabel 3 Delay Akses Client Pada Menu Live Audio Streaming

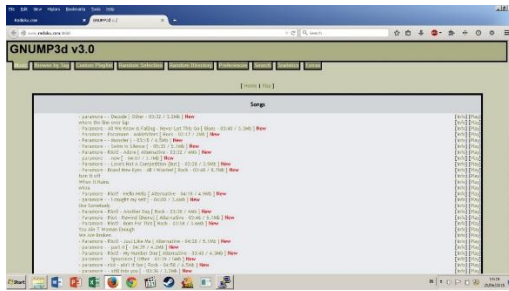
	Delay
Client ke-1	2 detik
Client ke-2	2,25 detik
Client ke-3	2,5 detik

7. Akses web shoutbox oleh client



Gambar 10 Tampilan Shoutbox

8. Akses mp3 streaming oleh client



Gambar 11 Tampilan Mp3 Streaming

5. Kesimpulan dan Saran

5.1. Kesimpulan

Dari pengujian pada bab 4, dapat ditarik kesimpulan yaitu :

1. Raspberry pi dapat diimplementasikan sebagai server untuk live audio streaming, user feedback dan mp3 streaming.
2. Berdasarkan pengujian yang telah dilakukan, aplikasi web untuk live audio streaming dapat berjalan dan diimplementasikan, namun masih terdapat delay 2 detik serta bersautan dimulai dari client 1 sampai client 3 untuk audio setelah dibroadcast oleh admin kepada user.
3. Setelah melakukan pengujian terhadap tab menu shoutbox dan mp3 streaming, dapat disimpulkan bahwa fitur user feedback dan fitur mp3 streaming berhasil diimplementasikan.

5.2. Saran

saran yang dapat penulis berikan pada proyek akhir ini, yaitu :

1. Untuk jaringan server live audio streaming ini kedepannya, diharapkan dapat diintegrasikan dengan ip publik agar dapat digunakan untuk umum.

Daftar Pustaka

- [1] A. C. A. S, Cara Cepat Membuat Live TV di Blog dan Website, Yogyakarta: C.V ANDI OFFSET, 2010.
- [2] D. Austerberry, The Technology of Video & Audio Streaming, San Francisco: Focal Press, 2004.
- [3] S. Nazarko, Raspberry Pi Media Center, Birmingham: Packt Publishing Ltd., 2013.
- [4] K. Yaghmour, Building Embedded Linux Systems, New York: O'Reilly Media, Inc., 2008.
- [5] J. William E. Shotts, The Linux Command Line: A Complete Introduction, San Francisco: William Pollock, 2012.

