

IMPLEMENTASI DAN ANALISIS MOBILE CLOUD GAMING ONLINE MENGUNAKAN OPEN-SOURCE CLOUD GAMING SERVER GAMINGANYWHERE PADA PERANGKAT ANDROID

IMPLEMENTATION AND ANALYSIS ONLINE MOBILE CLOUD GAMING FOR ANDROID CLIENT USING OPEN-SOURCE CLOUD GAMING SERVER GAMINGANYWHERE

Edoardo Setiawan¹, Dr. Ir. Rendy Munadi, M.T², Sussi, S.Si, M.T,³

^{1,2,3}Prodi S1 Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom, Bandung

¹edhosetiawan@gmail.com, ²rendymunadi@telkomuniversity.ac.id, ³sussiss@telkomuniversity.ac.id

Abstrak

Mobile cloud gaming merupakan pengembangan dari infrastruktur sebagai model layanan *cloud computing*. Dengan adanya layanan *cloud gaming server* menggunakan *open-source GamingAnywhere*, perangkat *mobile phone* dengan *operation sytem android* yang bekerja sebagai client mampu mengakses server yang menjalankan suatu game melalui jaringan nirkabel secara *online*. Server melakukan capture terhadap *frame audio* dan *frame video* kemudian dikodekan dan dikirimkan kepada client. Setelah client mendapatkan *frame A/V*, client akan mendekodekan *frame-frame* yang diterima sehingga user bisa bermain dan melakukan *input control*. Untuk mendapatkan hasil performa dari sistem *cloud gaming* yang diimplementasikan ini, dilakukan pengujian dengan tiga parameter yaitu kebutuhan *resource*, *Quality of Service*, dan *Quality of Experience*.

Hasil kebutuhan resource client yang hanya memakai 6.2% cpu usage, 20.6 MB RAM untuk game Neverball dan 3.6% cpu usage, 11.9 MB RAM untuk game Deadpool. Nilai FPS client yaitu 29.1 untuk Neverball dan 14.5 untuk Deadpool, nilai ini terbilang kecil dikarenakan kemampuan rendering server yang kurang maksimal. Untuk delay total sistem cukup memuaskan yaitu bernilai 0.19 detik untuk Neverball pada jaringan lokal dan 0.35 detik pada jaringan online. Sedangkan 0.13 detik untuk Deadpool dengan jaringan lokal dan 0.19 detik untuk jaringan online pada bandwidth 5 Mbps.

Kata Kunci: *Cloud Gaming, Mobile Cloud Gaming, GamingAnywhere, Cloud Computing.*

Abstract

Mobile cloud gaming is a development of infrastructure as a model of cloud computing services. With a cloud gaming server service using open-source GamingAnywhere, mobile phone devices with android operating sytem working as a client are able to access servers that run a game over the wireless network online. The server captures the audio frame and video frame and then encoded and sent to the client. After the client gets the A / V frame, the client will decode the received frames so that the user can play and perform the input control. To get the performance results of the cloud gaming system implemented, tested with three parameters namely resource requirements, Quality of Service, and Quality of Experience.

Result of resource client needs uses only 6.2% cpu usage, 20.6 MB RAM for Neverball games and 3.6% cpu usage, 11.9 MB RAM for Deadpool games. FPS client value is 29.1 for Neverball and 14.5 for Deadpool, this value is small due to less than the maximum server rendering capability. For the total system delay is quite satisfactory that is worth 0.19 seconds for Neverball on the local network and 0.35 seconds on the online network. While 0.13 seconds for Deadpool with local network and 0.19 seconds for online network at 5 Mbps bandwidth.

Keywords: *Cloud Gaming, Mobile Cloud Gaming, GamingAnywhere, Cloud Computing.*

1. Pendahuluan

Perkembangan teknologi disebuah telepon seluler atau *mobile phone* yang pesat serta hadirnya *operating system android* pada telepon seluler, pengguna telepon seluler mampu mengakses serta mengatur layanan maupun fitur-fitur pada sistem operasi tersebut dengan bebas. Telepon seluler atau *mobile phone* tidak hanya berfungsi sebagai alat komunikasi saja, namun dengan adanya berbagai layanan *multimedia*, pengguna telepon seluler dapat bermain *game* dengan pengalaman yang menyenangkan, maka kebutuhan akan teknologi *game* tersebut juga akan besar terutama teknologi dibidang *mobile cloud gaming*.

Mobile cloud gaming merupakan sebuah layanan *cloud computing* yang merupakan pengembangan dari infrastruktur sebagai model layanan *cloud computing*, *cloud gaming* pada umumnya, menghasilkan sebuah aplikasi *gaming* interaktif secara *remote* di dalam *cloud* dan meneruskan keluaran berupa video ke perangkat *android* melalui internet. Pemain akan berinteraksi dengan aplikasi tersebut melalui sebuah *thin client*, yang bertanggung jawab untuk menampilkan video dari *server cloud* dan juga mengumpulkan masukan perintah dari pemain dan mengirimkan interaksi tersebut kembali ke dalam *cloud*. *Cloud gaming* tercipta karena spesifikasi didalam *game* membutuhkan spesifikasi yang tinggi seperti CPU (*Central Processing Unit*), GPU (*Graphics Processing Unit*), RAM (*Random-Access Memory*) maupun *Harddisk* yang besar. Dengan spesifikasi seperti itu maka tidak efisien jika 1 perangkat hanya dapat bermain *game* dengan spesifikasi yang tinggi, dan perangkat dengan spesifikasi yang rendah tidak dapat memainkan *game* tersebut. *Cloud gaming* menjadi solusi atas spesifikasi yang tinggi tersebut. Hanya *server* yang akan menggunakan spesifikasi yang tinggi sedangkan *client* dengan spesifikasi rendah dapat menikmati *game* tersebut.

Pada Tugas Akhir ini, akan dilakukan implementasi sebuah infrastruktur *mobile cloud gaming* baik secara lokal maupun secara *online* yang melalui jaringan internet menggunakan *server* GamingAnywhere.

2. Teori

2.1 Cloud Computing

Cloud computing adalah model komputasi yang memungkinkan akses jaringan dimana saja, disesuaikan dengan permintaan pada sumber daya komputasi bersama yang dapat dikonfigurasi (seperti jaringan, server, penyimpanan, aplikasi dan layanan) yang dapat dengan cepat ditetapkan dan dirilis dengan upaya manajemen atau layanan interaksi penyedia jasa. Cloud computing ini terdiri dari 5 karakteristik penting, 3 model layanan dan 4 infrastruktur di cloud computing^[10], sedangkan Menurut Onno W. Purbo, cloud computing adalah suatu model komputasi, dimana sumber daya seperti processor/computing power, storage, network, dan software menjadi abstrak dan diberikan sebagai layanan di jaringan/internet menggunakan pola akses remote^[12].

2.2 Cloud Gaming

Cloud gaming dalam bentuk sederhananya yaitu proses menjalankan sebuah aplikasi *gaming* interaktif secara *remote* di dalam *cloud* dan mengeluarkan keluaran dalam bentuk *video* ke perangkat pengguna melalui internet^[11]. Pengguna melakukan interaksi dengan aplikasi melalui sebuah *thin client*, yang bertanggung jawab untuk menampilkan video dari *cloud rendering server* dan juga mengumpulkan perintah-perintah dari pengguna serta mengirimkannya kembali ke *cloud*^[11].

2.2.1 Mobile Cloud Gaming

Mobile cloud gaming didefinisikan sebagai suatu perangkat *mobile* yang mampu mengakses *cloud* sebagai eksternal *resource* untuk pengolahan skenario dan interaksi untuk menjalankan suatu *game*. Ada dua tipe *mobile cloud games*^[13], yaitu:

1. *Mobile cloud video gaming*

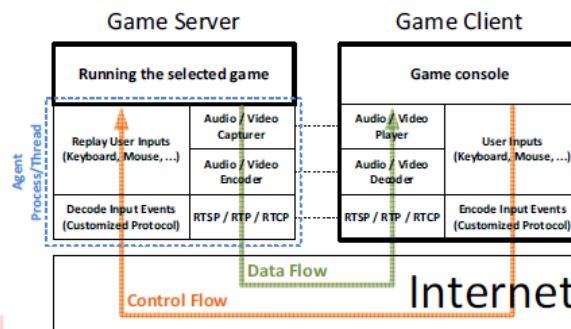
Pemain mengakses *server game* dari portal yang tersedia di *mobile phone*, kemudian client melakukan inisiasi pada *server machine* dan menjalankan game yang telah di pilih di *mobile phone*. Saat game sedang berjalan di *server machine* atau *virtual machine*, sisi *client* lalu melakukan *game streaming* dimana game berjalan bersamaan pada *client* dan *server machine* atau *virtual machine* di *cloud*.

2. *Mobile browser gaming*

Browser games merupakan jenis dari *games* komputer, banyak dari *game* tersebut berjalan di situs jejaring sosial dengan pengguna yang cukup banyak. *Server mobile browser games* berjalan di *cloud*. Sementara *web browser* pada ponsel berfungsi sebagai *interface* untuk pemain. Komponen penting dalam *mobile browser games* meliputi:

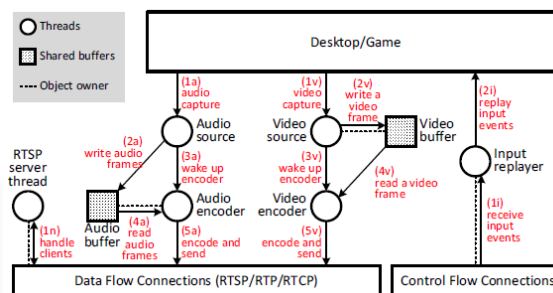
- a) *Game server* : aplikasi pada sisi *server*, yang terdiri dari *server-side scripting* dan *user-interface scripting*.
- b) *Communication protocol* : untuk mengaktifkan prosedur sebuah game, komunikasi antara *server* dengan *mobile browser*.
- c) Aplikasi *runtime* : pada ponsel, aplikasi *runtime* harus ter-install pada *web browser* sebagai *plugin*.

2.3 GamingAnywhere



Gambar 2.1 Hubungan Server dan Client^[2]

Terdapat dua jenis *network flow* pada arsitektur *GamingAnywhere*, yakni *data flow* dan *control flow*. *Data flow* digunakan untuk men-stream-kan *frames* audio dan video dari *server* ke *client*, sedangkan *control flow* berjalan dalam arah yang berlawanan, yakni digunakan untuk mengirimkan gerakan pengguna dari sisi *client* ke *server*. Pada sistem *GamingAnywhere*, terdapat beberapa *external libraries* yang digunakan untuk menjalankan sistem, yaitu *libavcodec/libavformat*, *live555* dan *SDL library*^[2].



Gambar 2.2 Hubungan antar modul server, shared buffer dan koneksi jaringan^[2]

Pada *GamingAnywhere*, *RTSP server thread* merupakan hal pertama yang dijalankan pada *server*. *RTSP server thread* menerima *RTSP commands* dari sebuah *client*, menjalankan *encoder* dan mengatur *data flow* untuk mengirimkan *frames* yang telah diencode^[2]. *Data flows* dapat dibawa/disampaikan melalui koneksi jaringan tunggal atau banyak (*multiple*) berdasarkan 12rotocol layer *transport* yang diinginkan, seperti TCP atau UDP. Pada TCP, *frames* yang terencode dikirimkan sebagai data biner dalam *RTSP*. Kedua *RTSP command* dan *RTP/RTCP packets* dikirimkan melalui koneksi *RTSP over TCP* yang terbangun dengan *client*. Pada UDP, *frames* yang terencode dikirimkan melalui *RTP over UDP*.

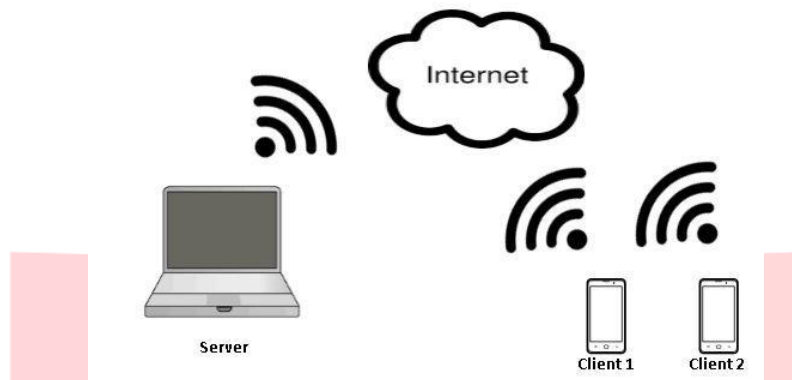
3. Perancangan Sistem

3.1 Model Sistem

Sistem ini menghasilkan sebuah interaksi *game* di sisi *client* dan *server*. Dengan memanfaatkan teknologi *cloud computing* dalam *cloud gaming*, pengguna akan dapat memainkan *game* tertentu yang sebelumnya telah dipasangkan dalam *server* mengendalikannya melalui perangkat *android* baik secara lokal maupun secara *online* melalui jaringan internet. Di dalam sistem ini, satu laptop dijadikan sebagai *server* tunggal beserta judul *game* tertentu untuk melayani 2 *client*, *client 1* yang bertindak sebagai pemain utama dan *client 2* yang bertindak sebagai *observer*, tetapi masih dapat melakukan input kendali. *Server* akan mengirimkan *stream video* dan *audio* kepada pengguna melalui jaringan lokal maupun melalui jaringan internet. Di sisi pengguna, pengguna melakukan interaksi dengan *game* melalui aplikasi yang sebelumnya sudah di *install* terlebih dahulu untuk memainkan *game* yang *server* jalankan melalui perangkat *android*.

3.2 Desain Model Sistem

3.2.1 Blok Sistem



Gambar 3.1 Blok sistem mobile cloud gaming secara online

Gambar 3.1 menunjukkan bahwa sistem terdiri dari dua sisi, yaitu sisi *server* dan sisi pengguna (*client*). Di dalam sistem ini, satu laptop dijadikan sebagai *server* tunggal beserta judul *game* tertentu untuk melayani 2 *client*, *client 1* yang bertindak sebagai pemain utama dan *client 2* yang bertindak sebagai *observer*, tetapi masih dapat melakukan input kendali. Pada sisi *server*, *server* menjalankan *game* dengan mengirimkan *video stream* maupun *audio stream* melalui jaringan internet, kemudian pada sisi *client*, *client* memasukan ip publik dari *server* dari aplikasi yang sebelumnya telah ter *install* pada sisi *client* untuk memainkan *game* yang dijalankan oleh *server* melalui jaringan internet.

3.3 Skenario Pengujian

Dalam pengujian *mobile cloud gaming* pada tugas akhir ini, ada tiga aspek dalam pengujian diantaranya yaitu kebutuhan *resource* yang meliputi *CPU Usage*, *RAM Usage*, *Frame Rate (FPS)* untuk sisi *server* dan sisi *client*, *Quality of Service (QOS)* yang meliputi *delay*, *packet loss video*, dan *throughput* pada sistem *cloud gaming* dan *Quality of experience (QOE)* yang meliputi *responsiveness*, *control*, dan *graphic quality*.

4. Analisis

4.1 Pengujian Resource

Berikut ini hasil pengujian resource server dengan masing-masing rata-rata 10 kali percobaan untuk setiap resolusi yang digunakan

Tabel 4.1 Hasil pengukuran *resource usage* pada game Neverball - GA Tidak Aktif

Game	Resolusi	Avg. CPU Usage (%)	RAM (MB)	FPS
Neverball	1024 x 768	26.4	89.5	90.5
	1280 x 720	27.4	95.6	80.4
	1360 x 768	24.4	93.4	67.6

Tabel 4.2 Hasil pengukuran *resource usage* pada game Neverball - GA Aktif

Game	Resolusi	Avg. CPU Usage (%)	RAM (MB)	FPS
Neverball	1024 x 768	26.6	98.4	84.2
	1280 x 720	27.4	151.5	73.4
	1360 x 768	24.6	211.5	62.8

Tabel 4.3 Hasil pengukuran *resource usage* pada game Deadpool - GA Tidak Aktif

Game	Resolusi	Avg. CPU Usage (%)	RAM (MB)	FPS
Deadpool	1024 x 768	55.1	348.1	20.8
	1280 x 720	63.6	365.2	17.1
	1360 x 768	67.1	369.7	15.5

Tabel 4.4 Hasil pengukuran *resource usage* pada game Deadpool - GA Aktif

Game	Resolusi	Avg. CPU Usage (%)	RAM (MB)	FPS
Deadpool	1024 x 768	59.6	616.1	16.1
	1280 x 720	70.4	650.4	14.7
	1360 x 768	72.6	654.4	12.6

Tabel 4.5 Hasil pengukuran *resource usage* pada game Neverball - GA Aktif 1 client

Game	Resolusi	Avg. CPU Usage (%)	RAM (MB)	FPS
Neverball	1024 x 768	35.4	341.0	80.3
	1280 x 720	44.2	365.0	69.1
	1360 x 768	49.4	389.4	61.8

Tabel 4.6 Hasil pengukuran *resource usage* pada game Deadpool - GA Aktif 1 client

Game	Resolusi	Avg. CPU Usage (%)	RAM (MB)	FPS
Deadpool	1024 x 768	68.8	650.1	16
	1280 x 720	74.1	694.7	14.4
	1360 x 768	79.2	664.8	12.4

Tabel 4.7 Hasil pengukuran *resource usage* pada game Neverball - GA Aktif 2 client

Game	Resolusi	Avg. CPU Usage (%)	RAM (MB)	FPS
Neverball	1024 x 768	40.1	357.7	80.1
	1280 x 720	47.2	392.5	68.6
	1360 x 768	56.6	403.1	61.1

Tabel 4.8 Hasil pengukuran *resource usage* pada game Deadpool - GA Aktif 2 client

Game	Resolusi	Avg. CPU Usage (%)	RAM (MB)	FPS
Deadpool	1024 x 768	73.5	641.2	16
	1280 x 720	75.4	800.3	13.9
	1360 x 768	79.6	800.8	12.1

Berikut ini hasil pengujian resource *client* dengan masing-masing 10 kali percobaan :

Tabel 4.9 Hasil pengukuran resource usage di sisi client pada game Neverball

Game	Resolusi	CPU Usage (%)		RAM (MB)		FPS	
		Client 1	Client 2	Client 1	Client 2	Client 1	Client 2
Neverball	1024x768	6.6	5.6	20.2	21.1	29.7	25.6
	1280x720	6.5	6.1	20.5	21.1	30.2	29.5
	1360x768	6.3	6.2	20.2	20.8	30.2	29.4

Tabel 4.10 Hasil pengukuran resource usage di sisi client pada game Deadpool

Game	Resolusi	CPU Usage (%)		RAM (MB)		FPS	
		Client 1	Client 2	Client 1	Client 2	Client 1	Client 2
Deadpool	1024x768	3.3	4.3	11.2	12.3	14.7	15.3
	1280x720	3.2	4.5	11.5	12.2	13.9	14.5
	1360x768	2.6	4.2	12.2	12.3	14.4	14.4

4.2 Pengujian Quality of Service

Berikut ini hasil *delay*, *packet loss* dan *throughput* pada game Neverball dan Deadpool. Total *delay* adalah jumlah dari keseluruhan *delay*. Hasil pengukuran merupakan rata-rata dari 10 kali percobaan.

Tabel 4.11 Hasil pengukuran delay pada game Neverball jaringan lokal

Bandwidth	Delay (s)				
	Video	Audio	Control	Downlink Delay	Uplink Delay
1 Mbps	0.0166206	0.016414	0.0179873	0.1157228	0.1143696
3 Mbps	0.0106601	0.0210341	0.0126602	0.0865143	0.0941126
5 Mbps	0.0098998	0.027157	0.0125569	0.0462754	0.1019164

Tabel 4.14 Hasil pengukuran delay pada game Deadpool jaringan lokal

Bandwidth	Delay (s)				
	Video	Audio	Control	Downlink Delay	Uplink Delay
1 Mbps	0.0147855	0.0158496	0.0159574	0.0539087	0.044104
3 Mbps	0.0104348	0.0224977	0.0130493	0.0486959	0.0404753
5 Mbps	0.0100717	0.0262771	0.0129326	0.0447844	0.0395795

Tabel 4.12 Hasil pengukuran delay pada game Neverball jaringan online

Bandwidth	Delay (s)				
	Video	Audio	Control	Downlink Delay	Uplink Delay
1 Mbps	0.037872	0.0383324	0.0376191	0.16635783	0.1898438
3 Mbps	0.0351402	0.0373459	0.0363268	0.1221853	0.1942574
5 Mbps	0.0079461	0.0089587	0.0083202	0.1532133	0.17529

Tabel 4.15 Hasil pengukuran delay pada game Deadpool jaringan online

Bandwidth	Delay (s)				
	Video	Audio	Control	Downlink Delay	Uplink Delay
1 Mbps	0.041712	0.0353285	0.041834	0.1473213	0.1778631
3 Mbps	0.0338692	0.0348678	0.0339504	0.0830628	0.0831158
5 Mbps	0.0153988	0.0156424	0.0150861	0.0790402	0.0700595

Tabel 4.13 Total delay game Neverball

Bandwidth	Total Delay (s)	
	Lokal	Online
1 Mbps	0.2811143	0.47002513
3 Mbps	0.2249813	0.4252556
5 Mbps	0.1978055	0.3537283

Tabel 4.16 Total delay game Deadpool

Bandwidth	Total Delay (s)	
	Lokal	Online
1 Mbps	0.1446052	0.4440589
3 Mbps	0.135153	0.268866
5 Mbps	0.1336453	0.195227

Berikut ini hasil pengukuran *packet loss*:

Tabel 4.17 Hasil pengukuran packet loss pada game Neverball

Bandwidth (Mbps)	Packet Loss (%)			
	Lokal		Online	
	Video	Audio	Video	Audio
1	54.7	53.55	56.58	54.96
3	3.92	4.27	40.77	39.42
5	0.35	0.36	0.11	0.11

Tabel 4.18 Hasil pengukuran packet loss pada game Deadpool

Bandwidth (Mbps)	Packet Loss (%)			
	Lokal		Online	
	Video	Audio	Video	Audio
1	37.97	36.9	65.04	64.67
3	1.33	1.47	40.76	38.26
5	0.09	0.07	12.45	7.39

Berikut ini hasil pengukuran *throughput* :

Tabel 4.19 Hasil pengukuran throughput pada game Neverball

Bandwidth	Throughput(Mbps)	
	Lokal	Online
1 Mbps	0.8092	0.246
3 Mbps	0.9538	0.2482
5 Mbps	0.9954	0.9879

Tabel 4.20 Hasil pengukuran throughput pada game Deadpool

Bandwidth	Throughput(Mbps)	
	Lokal	Online
1 Mbps	0.8344	0.2157
3 Mbps	0.9392	0.3226
5 Mbps	0.9637	0.6749

Berikut ini hasil pengukuran *latency* :

Tabel 4.21 *Latency game Neverball*

Bandwidth	Latency (ms)	
	Lokal	Online
1 Mbps	76	349
3 Mbps	41	201
5 Mbps	17	122

Tabel 4.22 *Latency game Deadpool*

Bandwidth	Latency (ms)	
	Lokal	Online
1 Mbps	26	321
3 Mbps	4	156
5 Mbps	3	103

4.3 Pengujian Quality of Experience

Hasil pengukuran yaitu rata-rata dari 10 orang koresponden untuk masing-masing tiap *game*.

Tabel 4.23 Hasil pengukuran QoE

Game	Responsiveness	Control	Graphic Quality
Neverball	3.4	3.5	4.4
Deadpool	3.5	3.2	4.1

4.4 Analisis Kebutuhan Resource

Pada hasil kebutuhan *resource*, terlihat bahwa kebutuhan *resource* meningkat disaat GamingAnyWhere aktif. Peningkatan yang cukup signifikan yaitu pada CPU *usage*. Untuk RAM mengalami peningkatan sekitar 268 MB untuk *game* Deadpool. Untuk perubahan resolusi, hal yang paling berpengaruh yaitu pada FPS, dikarenakan resolusi yang semakin berat, maka proses *rendering* semakin berat juga dan membenani kinerja CPU dan RAM, dikarenakan pengujian dilakukan dengan *server* yang tidak memiliki spesifikasi tinggi. Maka hasil *frame* per detik yang dihasilkanpun rendah.

Hasil *resource client* terlihat sangat kecil dikarenakan *client* mengakses *game* dengan cara *streaming* sehingga proses *rendering game* dibebankan pada *server*.

4.5 Analisis Quality of Service

Tingkat total *delay* pada *game* Neverball jaringan lokal maupun *online* bernilai lebih tinggi dibandingkan *game* Deadpool jaringan lokal maupun *online* disebabkan paket *video* pada *game* Neverball lebih besar dan hal ini dibuktikan dengan lebih interaktifnya *game* Neverball dibandingkan *game* Deadpool. Oleh karena itu, kualitas *game* Neverball sangat buruk bila dijalankan dengan batas *bandwidth* 1 Mbps. Lain halnya dengan *game* Deadpool walaupun dijalankan dengan *bandwidth* 1 Mbps, namun *delay* total cenderung kecil dan kualitasnya cukup baik. Nilai tingkat *delay* total jaringan *online* cenderung lebih tinggi dibandingkan jaringan lokal hal ini dikarenakan jaringan *online* dilalui oleh jaringan *internet* yang membutuhkan *transfer rate* yang cepat dan stabil untuk mendapatkan nilai tingkat *delay* total yang kecil.

Untuk *bandwidth* 3 dan 5 Mbps, *game* dapat berjalan dengan kualitas yang cukup baik untuk jaringan lokal, sedangkan untuk jaringan *online*, *delay* total cenderung menurun dibandingkan dengan *bandwidth* 1 Mbps dan untuk *bandwidth* 5 Mbps adalah batas minimal untuk memainkan *game cloud gaming* secara *online* untuk meminimalkan nilai *delay*.

packet loss pada *game* Neverball lokal maupun Neverball *online* dalam kondisi *bandwidth* 1 Mbps, sangat lah tinggi. *Packet loss video* pada Neverball lokal mencapai 54.7% dan pada *packet loss audio* sebesar 53.55%, sedangkan *packet loss* pada *game* Neverball *online* mencapai 56.58% dan *packet loss audio* sebesar 54.96% yang menyebabkan *video* di sisi *client* sangat tidak beraturan dan membuat *game* tidak dapat dimainkan. Dalam kondisi *bandwidth* 3 Mbps, jumlah *packet loss* terbilang rendah untuk *game* Neverball lokal karena sudah mencapai nilai 3.92% *packet loss video* dan 4.27% untuk *packet loss audio* dan sudah mendekati dengan apa yang diharapkan, sedangkan untuk *game* Neverball *online* masih terbilang tinggi namun berangsur turun dengan nilai 40.77% untuk *packet loss video* dan 39.42% untuk *packet loss audio*. Sedangkan untuk kondisi *bandwidth* 5 Mbps, *game* Neverball lokal maupun Neverball *online* sudah mendekati 0% dan *audio/video frames* yang diterima di sisi *client* sudah sesuai dengan yang diharapkan.

Pada *game* Deadpool, dalam kondisi *bandwidth* 1 Mbps baik *game* Deadpool lokal maupun *online* *packet loss* yang dihasilkan sangatlah tinggi, *packet loss video* mencapai 39.97% dan *audio* mencapai 36.9% untuk *game* Deadpool lokal. Sedangkan untuk *game* Deadpool *online* *packet loss video* mencapai 65.04% dan *packet loss audio* mencapai 64.67%. Hal ini menyebabkan banyak *frame* yang hilang dan membuat *game* tidak dapat dimainkan sebagaimana semestinya. Dalam kondisi *bandwidth* 3 Mbps, *packet loss* yang dihasilkan cukup

rendah pada sisi *game* Deadpool lokal, hal ini bisa dilihat dari nilai *packet loss video* hanya sebesar 1.33% dan *packet loss audio* sebesar 1.47%, nilai ini sudah cukup untuk mendapatkan pengalaman bermain yang memuaskan. Sedangkan *packet loss game* Deadpool online masih terbilang sangat buruk hal ini dibuktikan dengan nilai *packet loss video* 40.76% dan *packet loss audio* 38.26% hal ini dipengaruhi oleh kecepatan internet yang digunakan oleh *client*. Pada *bandwidth* 5 Mbps untuk *game* Deadpool lokal sangat memuaskan karena hampir tidak ada paket yang hilang untuk setiap *frame game* yang dijalankan, hal ini dibuktikan dengan *packet loss video* yang hanya sebesar 0.09% dan 0.07% untuk *packet loss audio*, sedangkan untuk *game* Deadpool online nilai yang dihasilkan 12.45% *packet loss video* dan 7.39% *packet loss audio* walaupun ada sedikit *packet loss*, namun nilai ini masih cukup untuk mendapatkan pengalaman bermain yang cukup memuaskan terutama dimainkan secara *online*.

Untuk *throughput* dapat dilihat bahwa semakin kecil *throughput* pada jaringan, maka semakin besar nilai *packet loss* yang diterima baik jaringan lokal maupun jaringan *online*.

Untuk *latency* pada jaringan lokal *latency* dengan *limit bandwidth* 1 Mbps, membutuhkan waktu sekitar 76 ms untuk *game* Neverball lokal, dan 26 ms untuk *game* Deadpool lokal. Nilai *latency* pada jaringan lokal bernilai jauh lebih kecil dibandingkan dengan jaringan *online*, hal ini dikarenakan jarak antara *client* dan *server* pada jaringan lokal mempengaruhi besar kecilnya *latency*. Sedangkan jumlah waktu yang dibutuhkan paket data untuk berpindah pada jaringan *online* jauh lebih kompleks dan panjang dibandingkan dengan jaringan lokal.

5. Kesimpulan

Setelah melakukan pengukuran serta analisis hasil pengukuran, dapat ditarik beberapa kesimpulan, yaitu :

1. Sistem *mobile cloud gaming server* mampu mengimplementasikan suatu *game* PC yang dapat dijalankan pada *mobile phone* Android serta menekan *resource CPU usage* dan *RAM usage* pada *client* dengan dibuktikannya hasil *CPU usage* kurang dari 10% dan *RAM usage* kurang dari 25 MB untuk kedua *game*. *Frame* yang dihasilkan tiap detik yaitu sekitar 29.1 *frame* untuk Neverball dan 14.5 *frame* untuk Deadpool. FPS pada Deadpool terbilang rendah dikarenakan kemampuan *server* yang tidak mampu *render game* dengan maksimal.
2. *Trade off* pada implementasi pada sistem ini dibebankan pada sisi *server* dikarenakan *server* yang menanggung semua proses instalasi maupun *load ga-hook.dll* sehingga *resource CPU usage* dan *RAM usage* pada sisi *server* naik, sedangkan pada sisi *client* akan dihasilkan *resource* yang efisien pada *CPU usage* dan *RAM usage*.
3. Untuk meraih QoS yang optimal untuk jaringan lokal setidaknya dibutuhkan *bandwidth* sebesar 3 Mbps. Bila *bandwidth* yang diberikan kurang dari 3 Mbps, sistem akan mengalami *delay* yang massif bernilai ± 0.28 detik pada *game* Neverball dan bernilai ± 0.14 detik pada *game* Deadpool. *Packet loss* yang dihasilkan pun akan sangat tinggi, bernilai 54.7% untuk video dan 53.55% untuk audio pada *game* Neverball dan bernilai 39.37% untuk video dan 36.9% untuk audio pada *game* Deadpool. Sedangkan untuk meraih QoS yang optimal untuk jaringan *online* setidaknya dibutuhkan *bandwidth* sebesar 5 Mbps, bila *bandwidth* yang diberikan kurang dari 5 Mbps, sistem akan mengalami *delay* yang tinggi ± 0.42 detik sampai ± 0.47 detik pada *game* Neverball dan bernilai ± 0.26 detik sampai ± 0.44 detik pada *game* Deadpool. *Packet loss* yang dihasilkan pun akan sangat tinggi, bernilai 40.77% sampai 56.58% untuk video dan 39.42% sampai 54.96% untuk audio pada *game* Neverball dan bernilai 40.76% sampai 65.04% untuk video dan 38.26% sampai 64.67% untuk audio pada *game* Deadpool.
4. Alokasi *storage client* hanya membutuhkan 7,8 MB untuk file *GAClient.apk* karena *file game* sebesar 200 MB untuk Neverball dan 20 GB untuk Deadpool disimpan pada *server*.
5. Hasil *user experience* dibuktikan dengan tingkat kepuasan *user* bermain *game* dengan *cloud gaming* pada *game* Neverball yaitu bernilai 3.4 atau sebesar 68% untuk *responsiveness*, 3.5 atau sebesar 70% untuk *control*, dan 4.4 atau sebesar 88% untuk *graphic quality*. Sedangkan pada *game* Deadpool yaitu 3.5 atau sebesar 70% untuk *responsiveness*, 3.2 atau sebesar 64% untuk *control*, dan 4.1 atau sebesar 82% untuk *graphic quality*. Tingkat kepuasan *user* untuk *game* Neverball dengan hasil rata-rata puas, sedangkan untuk *game* Deadpool dengan hasil cukup puas.
6. Untuk menjalankan *game* melalui *cloud gaming service* setidaknya *game* yang dijalankan harus mempunyai *library* seperti *SDL*, *d9* dan *d10*. Jika *game* yang dijalankan tidak mempunyai *library* tersebut, maka *GamingAnywhere* tidak bisa menjalankan *cloud gaming service* dikarenakan *GamingAnywhere* tidak mendukung *library* lain.

Daftar Pustaka

- [1] Ahmad Zainudin, "PENGENALAN ANDROID" Politeknik Elektronika Negeri Surabaya.
- [2] Chun-Ying Huang, Cheng-Hsin Hsu, Yu-Chun Chang and Kuan-Ta Chen, "GamingAnywhere : An Open Cloud Gaming System", Department of Computer Science, National Taiwan Ocean University, 2013.
- [3] "Game Neverball" [Online]. Available : <http://www.mobygames.com/game/neverball>
- [4] "Game Deadpool" [Online]. Available : http://www.game-debate.com/games/index.php?g_id=4846&game=Deadpool
- [5] "GamingAnyWhere". [Online]. Available : <https://www.gaminganywhere.org>
- [6] Juha-Matti Vanhatupa, "BROWSER GAMES FOR ONLINE COMMUNITIES" International Journal of Wireless & Mobile Networks (IJWMN), Vol.2, No.3, August 2010
- [7] Kamaljit I. Lakhtaria, "Enhancing QOS and QOE in IMS enabled next generation networks" Networking and Internet Architecture, Aug, 2010.
- [8] Lia Kuswayatno.2006. Mahir Berkomputer, Jakarta : Penerbit Grafindo Media Pratama
- [9] Maulana Mujahidin, "Network Traffic Management , Quality of Service (Qos), Congestion Control dan Frame Relay" Universitas Gunadarma.
- [10] Peter Mell and Timothy Grance, "The NIST Definition of Cloud Computing," Recommendation of the National Institute of Standards and Technology, Sept, 2011.
- [11] R. Shea, J. Liu, E. Ngai, and Y. Cui, "Cloud gaming: Architecture and performance," *IEEE Netw.*, vol. 27, no. 4, pp. 16–21, 2013.
- [12] W. Purbo, Onno. 2012. Membuat Sendiri Cloud Computing Server Menggunakan Open Source, Yogyakarta : Penerbit Andi.
- [13] Wei Cai, Victor C.M Leung, and Min Chen, "Next generation Mobile Cloud Gaming," IEEE Seventh International Symposium on Service-Oriented System Engineering, 2013.