

**IMPLEMENTASI DAN ANALISIS *CLOUD GAMING*  
DENGAN SERVER GAMINGANYWHERE  
YANG DI BENTUK MENGGUNAKAN DOCKER CONTAINERS  
PADA TEKNOLOGI FOG COMPUTING**

*Implementation and Analysis of Cloud Gaming with the Gaminganywhere Server that is  
formed using Docker Containers on Fog Computing Technology*

**PROPOSAL PROYEK AKHIR**

**Diajukan sebagai syarat untuk mengambil Mata Kuliah Proyek Akhir**

oleh :

**ANDRO ELNATAN HARIANJA**

**6705184033**



**D3 TEKNOLOGI TELEKOMUNIKASI**

**FAKULTAS ILMU TERAPAN**

**UNIVERSITAS TELKOM**

**2021**

## LEMBAR PENGESAHAN

Proposal Proyek Akhir dengan judul :

IMPLEMENTASI DAN ANALISIS CLOUD GAMING  
DENGAN SERVER GAMINGANYWHERE  
YANG DI BENTUK MENGGUNAKAN DOCKER CONTAINERS  
PADA TEKNOLOGI FOG COMPUTING

*Implementation and Analysis of Cloud Gaming with the Gaminganywhere Server that is  
formed using Docker Containers on Fog Computing Technology*

oleh :

ANDRO ELNATAN HARIANJA

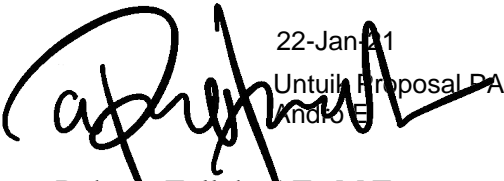
6705184033

Telah diperiksa dan disetujui untuk diajukan sebagai syarat mengambil  
Mata Kuliah Proyek Akhir  
pada Program Studi D3 Teknologi Telekomunikasi Universitas Telkom

Bandung, 20 Januari 2021

Menyetujui,

Pembimbing I

  
22-Jan-21  
Untuk Proposal PA  
Andro E

Rohmat Tulloh, S.T., M.T.

NIP. 06830002

Pembimbing II



Asep Mulyana, S.T., M.T.

NIP. 94570011

## ABSTRAK

Saat ini diketahui bahwa dalam beberapa tahun ini dunia game di dunia berkembang begitu pesat, mulai dari banyaknya *game* yang hadir dari *developer* yang tidak memiliki nama besar hingga game yang diciptakan oleh *developer* ternama. Hal ini juga diperkuat dengan hadirnya *device* dan *hardware* baru yang mendukung kehadiran game terbaru yang memiliki kualitas grafis yang sudah mendekati realistis yang tentunya membutuhkan *source* dari *device* dan *hardware* yang cukup besar dan tentunya berbanding lurus dengan biaya yang dikeluarkan untuk membeli *hardware* untuk menikmati *game* keluaran terbaru.

*Cloud gaming* merupakan proses menjalankan aplikasi gaming yang dilakukan secara remote pada sistem cloud dan memiliki keluaran dalam bentuk video ke perangkat pengguna melalui internet. Namun *cloud gaming* masih memiliki beberapa tantangan seperti harus menyewa *virtual machine* dengan spesifikasi yang cukup tinggi kepada vendor penyedia *cloud* untuk berkerja dengan baik dan memiliki tingkat *latency* yang cukup tinggi. Oleh karena pada proyek akhir kali ini digunakannya Docker Container untuk meningkatkan performa dari *cloud gaming* dan teknologi *fog computing* untuk menurunkan nilai *latency* agar *game* yang dimainkan lebih *play-able* oleh *gamer*.

Diharapkan dengan adanya *Cloud Gaming* dengan *Server Gaminganywhere* yang di bentuk menggunakan Docker Containers pada Teknologi *Fog Computing* ini *gamer* bisa menikmati *cloud gaming* dengan tingkat *latency* yang rendah, memiliki tingkat FPS yang tinggi dan memiliki kualitas video ada dengan resolusi 720p bahkan lebih.

Kata kunci : *Game, Hardware, Cloud Gaming, Docker Container, Fog Computing*

## DAFTAR ISI

<b>LEMBAR PENGESAHAN.....</b>	<b>i</b>
<b>ABSTRAK.....</b>	<b>ii</b>
<b>DAFTAR ISI .....</b>	<b>iii</b>
<b>DAFTAR GAMBAR .....</b>	<b>v</b>
<b>DAFTAR TABEL .....</b>	<b>vi</b>
<b>BAB I PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Tujuan dan Manfaat .....	3
1.3 Rumusan Masalah.....	3
1.4 Batasan Masalah .....	4
1.5 Metodologi.....	4
<b>BAB II DASAR TEORI .....</b>	<b>6</b>
2.1. <i>Cloud Computing</i> .....	6
2.2. <i>Cloud Gaming</i> .....	6
2.3. <i>Fog Computing</i> .....	6
2.4. Perbandingan Cloud Computing Dengan Fog Computing .....	7
2.5. Amazon Web Service .....	8
2.6. Gaminganywhere .....	9
2.7. Docker Containers .....	9
2.8. PuTTY .....	10
2.9. Wireshark.....	10
2.10. Amazon Cloudwatch.....	10
<b>BAB III MODEL SISTEM.....</b>	<b>12</b>
3.1 Blok Diagram Sistem.....	12
3.2 Tahapan Perancangan .....	13
3.3 Perancangan .....	15
<b>BAB IV BENTUK KELUARAN YANG DIHARAPKAN .....</b>	<b>16</b>

4.1	Keluaran yang Diharapkan .....	16
4.2	Jadwal Pelaksanaan.....	16
<b>DAFTAR PUSTAKA .....</b>		<b>17</b>

## DAFTAR GAMBAR

<i>Gambar 2.1 Framework dari cloud gaming secara sederhana [9] .....</i>	<i>6</i>
<i>Gambar 2 2 Amazon Web Service .....</i>	<i>8</i>
<i>Gambar 2.3 Arsitektur Gaminganywhere [3] .....</i>	<i>9</i>
<i>Gambar 2 4 Docker Containers .....</i>	<i>9</i>
<i>Gambar 2 5 PuTTY.....</i>	<i>10</i>
<i>Gambar 2 6 Wireshark .....</i>	<i>10</i>
<i>Gambar 2 7 Amazon Cloudwatch.....</i>	<i>10</i>
<i>Gambar 3.1 Model Sistem Perancangan Cloud Gaming .....</i>	<i>12</i>
<i>Gambar 3.2 Flowchart Tahapan Perancangan .....</i>	<i>14</i>

## DAFTAR TABEL

<i>Tabel 1.1 Referensi Tugas Akhir .....</i>	<i>2</i>
<i>Tabel 2.1 Perbandingan Cloud Computing dan Fog Computing [12] .....</i>	<i>7</i>
<i>Tabel 4.2 Jadwal Pelaksanaan .....</i>	<i>16</i>

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Saat ini diketahui bahwa dalam beberapa tahun ini dunia game di dunia berkembang begitu pesat, mulai dari banyaknya game yang hadir dari developer yang tidak memiliki nama besar hingga *game* yang diciptakan oleh *developer* ternama seperti EA, Ubisoft, Rockstar Games dan sebagainya dengan kualitas AAA.

Hal ini juga diperkuat dengan hadirnya *device* dan *hardware* baru yang mendukung kehadiran *game* terbaru yang memiliki kualitas grafis yang sudah mendekati realistis yang tentunya membutuhkan *source* dari *device* dan *hardware* yang cukup besar. Namun *device* dan *hardware* yang baru dan memiliki performa yang baik memiliki harga yang cukup mahal agar kita dapat menikmati pengalaman bermain game dengan kualitas maksimal. Sebagai contoh yang saya kutip dari website resmi Ubisoft yaitu untuk memainkan game Assassin's Creed Valhalla dengan resolusi 1080p dengan FPS (*Frames Per Seconds*) ada di 60 FPS membutuhkan spesifikasi yang setara dengan laptop Lenovo ThinkPad T470 dengan rentang harga Rp 9 juta hingga Rp 10 juta, belum termasuk dalam membeli gamenya dimana jika menggunakan *cloud gaming* kita tidak memerlukan hardware yang tidak terlalu tinggi, namun kita hanya mengeluarkan biaya untuk menyewa layanan cloud dari Amazon Web Service pada rentang harga Rp 500 ribu hingga Rp 1 juta tergantung dari *play time* selama sebulan.

Berdasarkan uraian tersebut, maka *cloud gaming* sangat memungkinkan menjadi solusi agar *gamer* dapat menikmati pengalaman terbaik walaupun memiliki device dan hardware yang tidak terlalu power full. Cloud gaming merupakan bagian dari cloud computing yang memiliki arti 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.[5] Adapun beberapa penelitian yang berkaitan dengan Tugas Akhir saya pada Table 1.1



Tabel 1.1 Referensi Tugas Akhir

No	Judul Penelitian /Karya Ilmiah	Tahun	Keterangan
1.	Penggunaan Virtual Machine Untuk Mengoptimalkan Kualitas Bermain Dengan Cloud Gaming [6]	2020	Pada penelitian ini pemateri dari Telkom University membuat <i>cloud gaming</i> dengan <i>server</i> pada jaringan <i>local</i> tepatnya pada laptop yang menggunakan VMWare / VirtualBox. Perbedaan jurnal ini dengan yang akan dibuat yaitu saya menggunakan <i>virtual machine</i> pada Amazon Web Service, docker container dan teknologi fog computing untuk meningkatkan performa pada cloud computing
3.	Cloud Gaming System in Docker Container Image [7]	2018	Pada penelitian ini pemateri membuat <i>cloud gaming</i> dengan sisten docker container untuk meningkatkan performa FPS dalam bermain game dan seluruh performa dari <i>cloud gaming</i> . Perbedaan jurnal ini dengan yang akan dibuat yaitu saya menggunakan teknologi <i>fog computing</i> untuk meningkatkan performa dari <i>cloud gaming</i>
4.	Implementasi Dan Analisis Mobile Cloud Gaming Online Menggunakan Open-Source Cloud Gaming Server Gaminganywhere Pada Perangkat Android [11]	2017	Pada penelitian ini pemateri dari Telkom University membuat <i>cloud gaming</i> menggunakan server Gaminganywhere pada perangkat android. Perbedaan jurnal ini dengan yang akan dibuat yaitu saya menggunakan docker container dan teknologi <i>fog computing</i> untuk meningkatkan performa pada <i>cloud computing</i>
5.	CloudFog: Towards High Quality of	2015	Pada penelitian ini pemateri membuat <i>cloud gaming</i> dengan menggunakan sisten <i>Fog</i>

	Experience in Cloud Gaming [13]		<i>Cloud</i> dapat mengurangi latensi respons dan konsumsi <i>bandwidth</i> serta meningkatkan jangkauan pengguna. Perbedaan jurnal ini dengan yang akan dibuat yaitu saya menggunakan docker container untuk meningkatkan performa dari <i>cloud gaming</i>
--	---------------------------------	--	--

Pada pengerjaan akhir ini akan membahas tentang Implementasi dan Analisis *Cloud Gaming* dengan *Server Gaminganywhere* yang di bentuk menggunakan Docker Containers pada teknologi *Fog Computing*.

## 1.2 Tujuan dan Manfaat

Adapun tujuan dan manfaat dari Proyek akhir ini, sebagai berikut:

1. Dapat merancang sebuah *cloud gaming* yang dapat menggantikan sistem *gaming* konvensional.
2. Dapat menghubungkan *server cloud gaming* dengan docker containers.
3. Dapat melakukan konfigurasi agar mengalihkan *server* ke teknologi *fog computing* dengan algoritma pemilihan *node*.

## 1.3 Rumusan Masalah

Adapun rumusan masalah dari Proyek akhir ini, sebagai berikut:

1. Bagaimana merancang sebuah *cloud computing* yang dapat menggantikan sistem *gaming* konvensional?
2. Bagaimana cara melakukan konfigurasi *server cloud gaming* dapat berkerja dengan docker containers?
3. Bagaimana cara melakukan konfigurasi *server* agar dapat mengalihkan *server* ke teknologi *fog computing*?

## 1.4 Batasan Masalah

Dalam Proyek akhir ini, dilakukan pembatasan masalah sebagai berikut:

1. *Server cloud computing* yang digunakan adalah gaminganywhere yang bersifat *open sources*.
2. Konfigurasi *server* gaminganywhere menggunakan Ubuntu 16.04 pada Amazon Web Service.
3. Menggunakan containers dari Docker Containers.
4. Menggunakan teknologi *fog computing* untuk menurunkan tingkat *latency*.
5. Menggunakan PuTTY untuk mensimulasikan Ubuntu pada *host* dalam melakukan konfigurasi.

## 1.5 Metodologi

Berikut uraian metodologi yang digunakan pada pengerjaan proyek ini adalah :

### 1.5.1. Studi Literature

Pada tahap ini yang dilakukan adalah mengidentifikasi sebuah masalah yang akan dijadikan sebagai sebuah proyek akhir. Setelah ide terbentuk, maka dilakukan sebuah observasi dan pencarian data-data yang terkait dengan permasalahan. Pencarian data dilakukan dengan mencari data valid di internet maupun di perpustakaan berupa jurnal yang dapat dipertanggung-jawabkan, link resmi dari produk / jasa yang bersangkutan dalam pembuatan sistem *cloud computing* dan beberapa buku.

### 1.5.2. Analisis Kebutuhan Sistem

Pada tahap ini menganalisa kebutuhan sistem dan yang akan dibuat. Analisis dilakukan seperti penentuan spesifikasi yang dibutuhkan untuk menjalankan *cloud gaming*, merancang arsitektur sistem dan alur sistem dari *cloud gaming* yang dibuat.

### 1.5.3. Implementasi

Diprosesnya menjadi serangkaian kode yang nantinya akan membentuk sebuah sistem dan aplikasi yang dengan menggunakan komponen-komponen yang sudah disurvei. Untuk pembuatan sistem, dibutuhkan akun Amazon Web Service yang memiliki *credits* minimal 100 dollar, PuTTY, platform gaminganywhere, dan docker containers. Dalam melakukan pengukuran untuk menganalisis kinerja dari *cloud computing* tersebut dibutuhkan Amazon Cloudwatch dan Wireshark.

#### **1.5.4. Pengujian**

Proses pengujian akan dilakukan setelah tahapan perancangan sistem selesai dibuat. Aplikasi akan diuji kebeberapa narasumber. Kemudian akan dilakukan pengisian kuisioner pengguna untuk memberikan kritik dan saran terhadap aplikasi yang diuji cobakan.

#### **1.5.5. Dokumentasi**

Pada tahap ini, dilakukan pembuatan laporan yang berupa buku Proyek dan jurnal dari hasil analisis.

#### **1.5.6. Promosi**

Pada tahap ini, dilakukan pembuatan poster produk agar dapat menjangkau khalayak banyak.

## BAB II

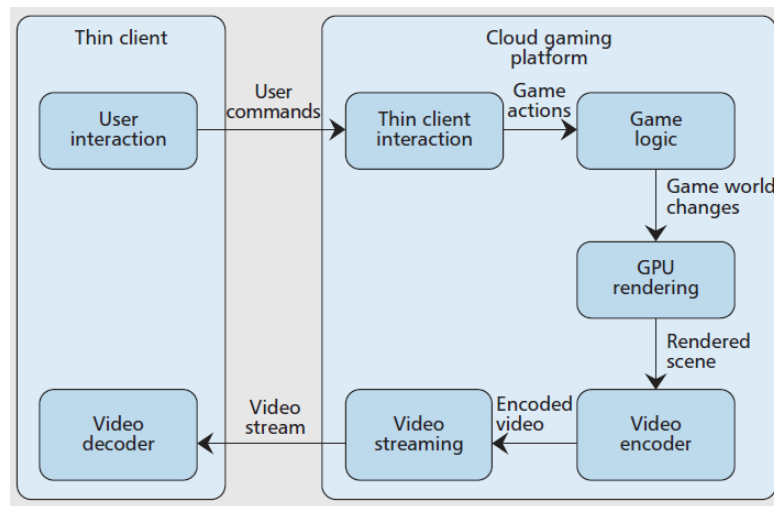
### DASAR TEORI

#### 2.1. Cloud Computing

*Cloud computing* adalah komputasi yang dapat melakukan akses jaringan dimana saja sesuai dengan permintaan pada sumber daya komputasi bersama yang dapat dikonfigurasi (seperti jaringan, *server*, penyimpanan, aplikasi dan layanan) yang dapat dengan cepat dalam menentukan sesuai keinginan dan dirilis dengan upaya manajemen atau layanan interaksi penyedia jasa.[5]

#### 2.2. Cloud Gaming

*Cloud gaming* merupakan proses menjalankan aplikasi *gaming* yang dilakukan secara *remote* pada sistem *cloud* dan memiliki keluaran dalam bentuk video ke perangkat pengguna melalui internet. 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*. [9]



Gambar 2.1 Framework dari cloud gaming secara sederhana [9]

#### 2.3. Fog Computing

*Fog Computing* adalah pengembangan dari *cloud computing* dimana komputasi didistribusikan ke *edge* dari sebuah jaringan yang memiliki konsep hampir sama dengan *mobile collaborative devices* atau *fixed nodes* yang didalamnya terdapat *storage*, komputasi dan *communication devices*. [12]

## 2.4. Perbandingan Cloud Computing Dengan Fog Computing

Dibawah ini merupakan tabel perbandingan antara cloud computing dengan fog computing

Tabel 2.2 Perbandingan Cloud Computing dan Fog Computing [12]

<b>Parameters</b>	<b>Fog Computing</b>	<b>Cloud Computing</b>
<b>Goal</b>	<i>Enhance proficiency and execution of process that should be transported to the cloud for handling, investigation and storage</i>	<i>Give a request of greatness change in the practical, powerful provisioning of IT administrations</i>
<b>Computational focuses</b>	<i>Fog operates on network edge</i>	<i>Data and applications are processed in the Cloud</i>
<b>Abstraction Level</b>	<i>High</i>	<i>High</i>
<b>scalability Degree</b>	<i>High</i>	<i>High</i>
<b>Support of Multitask</b>	<i>Yes</i>	<i>Yes</i>
<b>Level Transparency</b>	<i>High</i>	<i>High</i>
<b>Run time</b>	<i>Real time services</i>	<i>Real time services</i>
<b>Type of Requests</b>	<i>Lots of High allocation</i>	<i>Lots of small allocation</i>
<b>Allocation unit</b>	<i>All shapes and sizes (wide &amp; narrow)</i>	<i>All shapes and sizes (wide &amp; narrow)</i>
<b>Level of Virtualization</b>	<i>Vital</i>	<i>Vital</i>
<b>Accessible type</b>	<i>IP</i>	<i>IP</i>
<b>Transmission</b>	<i>Device to device</i>	<i>Device to Cloud</i>
<b>Security</b>	<i>Possible, determined</i>	<i>Undefined</i>
<b>Infrastructure</b>	<i>Flexible</i>	<i>3 models (PaaS, IaaS, SaaS)</i>
<b>Support of Operating System</b>	<i>hypervisor virtualization</i>	<i>A hypervisor (VM) on which multiple OSs run</i>
<b>Ownership</b>	<i>Multiple</i>	<i>Single</i>
<b>Service negotiation</b>	<i>SLA based</i>	<i>SLA based</i>
<b>Support of User management</b>	<i>Centralized</i>	<i>Centralized or can be delegated to third party</i>
<b>Resource management</b>	<i>Centralized</i>	<i>Centralized / Distributed</i>
<b>Allocation/Scheduling</b>	<i>Centralized</i>	<i>decentralized/centralized</i>
<b>Interoperability</b>	<i>Interoperability between heterogeneous resources.</i>	<i>Web Services (SOAP and REST)</i>
<b>Failure management</b>	<i>rescheduling of failed tasks</i>	<i>Strong (VMs can be easily migrated from one node to other)</i>
<b>Service price</b>	<i>Utility pricing and Pay per use as go</i>	<i>Utility pricing, discounted for larger customers</i>
<b>Type of service</b>	<i>CPU, network, memory, bandwidth, device, storage</i>	<i>IaaS, PaaS, SaaS, Everything as a service</i>

<b>Example of real world</b>	<i>Significant Fog applications involve real-time interactions rather than 228 batch processing.</i>	<i>Amazon Web Service (AWS), Google apps</i>
<b>Response Time</b>	<i>Low</i>	<i>High</i>
<b>Critical object</b>	<i>Service</i>	<i>Service</i>
<b>Number of users</b>	<i>Unlimited</i>	<i>Unlimited</i>
<b>Resource</b>	<i>Unlimited</i>	<i>Unlimited</i>

Dari tabel perbandingan diatas, ada beberapa parameter yang sama antara *cloud computing* dan *fog computing* seperti *portable access, virtualization, multitask, transparency, service negotiation, critical object, number of users support and resources*. Parameter lainnya dimana *fog computing* memberikan performa yang lebih baik seperti memberikan respon dalam waktu singkat, *request type, transmission, security, user management and resource management, scheduling, interoperability, failure management, pricing of services and type of services*. [12]

## 2.5. Amazon Web Service

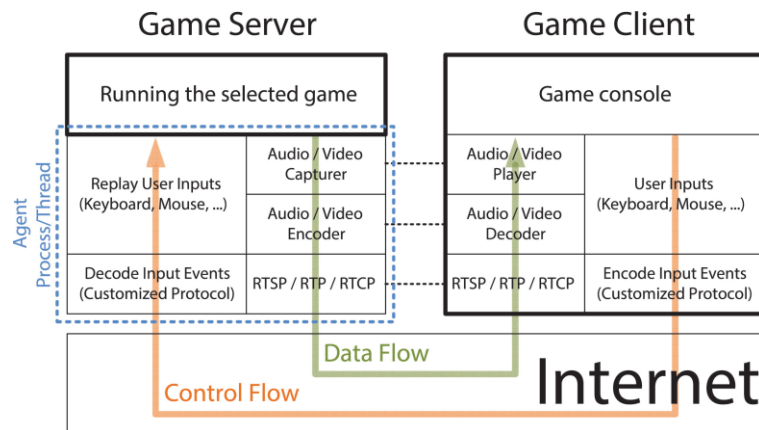


Gambar 2 2 Amazon Web Service

Amazon Web Services (AWS) adalah platform cloud paling komprehensif dan digunakan secara luas di dunia, menawarkan lebih dari 175 layanan unggulan yang lengkap dari pusat data secara global. Jutaan pelanggan termasuk beberapa startup dengan pertumbuhan tercepat, perusahaan terbesar, dan lembaga pemerintah terkemuka menggunakan AWS untuk memangkas biaya, menjadi lebih sigap, dan inovasi lebih cepat. [2]

## 2.6. Gaminganywhere

Gaminganywhere merupakan sistem yang bersifat *open source* yang berarti bahwa komponen dari *video streaming pipeline* dapat dengan mudah diganti komponennya dengan algoritma, standar atau protokol yang berbeda. Gaminganywhere juga bersifat *cross-platform* yang berarti dapat dijalankan pada Windows, Linux, OS X dan Android.[3]



Gambar 2.3 Arsitektur Gaminganywhere [3]

Pada Gambar 2.2, terdapat 2 tipe *network flows* pada arsitektur gaminganywhere yaitu *data flow* dan *control flow*. Fungsi dari kedua *network flows* memiliki perbedaan yaitu dimana *data flow* digunakan untuk melakukan *stream audio* dan *video* dari *game server* menuju *game client* sedangkan *control flow* untuk mengirimkan input dari *game client* menuju *game server*. [3]

## 2.7. Docker Containers



Gambar 2 4 Docker Containers

Docker merupakan platform virtualisasi container yang berkembang pesat dan sangat populer untuk Linux. Docker menggunakan teknik *copy-on-write*: sistem file *container* Docker didasarkan pada *read-only images*; perubahan yang dibuat dalam *container* yang sedang berjalan dapat disimpan ke *images* baru. Docker menyediakan sejumlah *tools* untuk membuat, memantau, dan mengelola kontainer, serta membuat *images* baru, membagikannya melalui repositori dan mencari gambar yang dibuat oleh pengguna lain.[10]



## 2.8. PuTTY



*Gambar 2 5 PuTTY*

PuTTY adalah klien SSH dan telnet, awalnya dikembangkan oleh Simon Tatham untuk platform Windows. PuTTY adalah perangkat lunak sumber terbuka yang tersedia dengan kode sumber dan dikembangkan serta didukung oleh sekelompok sukarelawan.[8]

## 2.9. Wireshark



*Gambar 2 6 Wireshark*

Wireshark disebut juga Network packet analyzer yang berfungsi menangkap paket-paket jaringan dan berusaha untuk menampilkan semua informasi dipaket tersebut sedetail mungkin. [4]

## 2.10. Amazon Cloudwatch



*Gambar 2 7 Amazon Cloudwatch*

Amazon CloudWatch adalah layanan pemantauan dan pengamatan yang dibangun untuk teknisi DevOps, *developer*, teknisi keandalan di lokasi dan manajer TI. CloudWatch memberi Anda wawasan mengenai data dan tindakan untuk memantau aplikasi Anda, merespon perubahan kinerja di seluruh sistem, mengoptimalkan pemanfaatan sumber daya, dan mendapatkan gabungan tampilan kesehatan operasional.

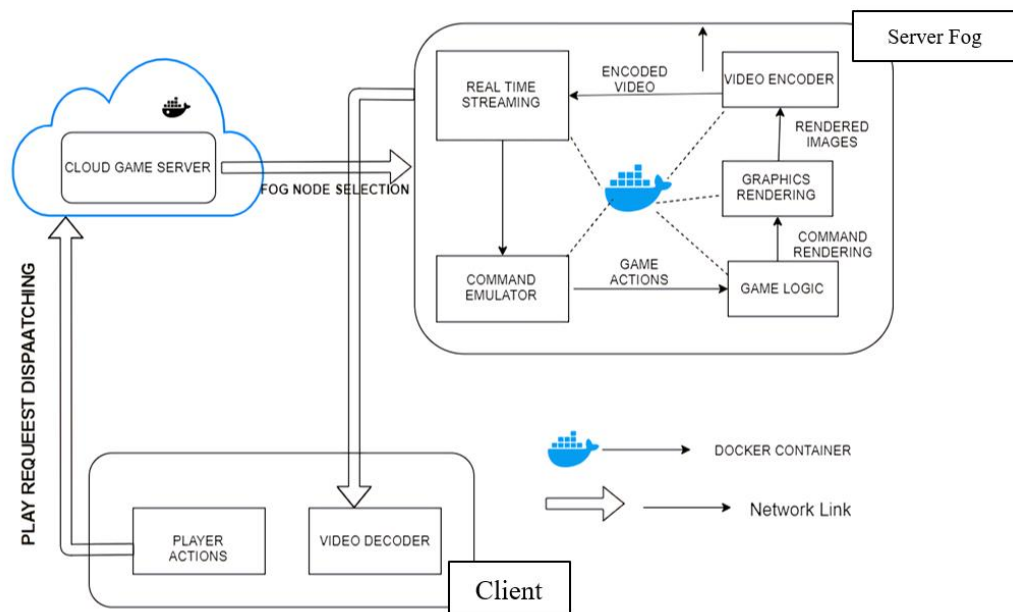
CloudWatch mengumpulkan data pemantauan dan operasional dalam bentuk log, metrik, dan peristiwa, memberi anda gabungan tampilan dari sumber daya, aplikasi, dan layanan AWS yang berjalan di AWS, dan server di lokasi.[1]

## BAB III

### MODEL SISTEM

#### 3.1 Blok Diagram Sistem

Pada bab ini akan dijelaskan mengenai perancangan sistem dari *cloud computing* yang dibentuk menggunakan *container* pada teknologi *fog computing*. Pertama client memberi input kepada *server gaminganywhere*. Setelah itu server mengidentifikasi alamat *IP public client* dan mengirim masukkan kepada *server fog* yang memiliki tingkat latensi yang terendah. Selanjutnya *server gaminganywhere* memilih *edge network client* dan menyebarkan file *container image* menuju *server fog*. Setelah *container image* berhasil di sebarakan maka *server gaminganywhere* siap melakukan *hosting game*. Selanjutnya *server fog* menerima *IP public client* maka *server fog* mengirimkan permintaan berkomunikasi dengan *client*. Setelah saling terhubung, *client* dapat bermain dan *video game* akan di *encoder* melalui *video encoder* dan di transfer ke *video streaming service*. Terakhir, *video streaming service* mengirim video permainan yang di *encoded game* menuju client dan akan di *decode* serta *video game* akan di *streaming* kan ke media player di sisi *client*. Adapun model sistem Gambar 3.1 dibawah ini.



Gambar 3.1 Model Sistem Perancangan Cloud Gaming

### 3.2 Tahapan Perancangan

Pada tahapan perancangan sistem ini dijelaskan langkah-langkah melakukan pembuatan Cloud Gaming dengan Server Gaminganywhere yang di bentuk menggunakan Docker Containers pada teknologi Fog Computing, tahapan pembuatanya adalah sebagai berikut:

1. Tahap Analisis

Langkah awal dalam proses perancangan suatu produk perlu adanya suatu kajian awal yang berguna untuk menentukan tujuan suatu produk dibangun. Analisis penting dalam perancangan suatu produk karena hal ini dapat menjamin keefektifan sumber daya serta tepat sasaran tujuan suatu produk yang dibangun. Tahap analisis dilakukan dengan mencari referensi dari beberapa forum yang membahas tentang *game* dan *cloud gaming*

2. Tahap Pembuatan *Cloud* pada Amazon Web Service

Dalam proses pembuatan *cloud* pada Amazon Web Service ini dimulai dengan cara Membuat *instance* berupa Ubuntu 16.04 AMI pada AWS EC2 *instance*. Dalam pembuatan *instance*, pastikan memilih GPU yang mendukung *graphics acceleration* karena digunakan untuk me-render *game*.

3. Tahap Pembuatan *Docker Containers*

Dalam proses pembuatan Docker Containers, lakukan installasi pada *instance* yang sudah dibuat lalu lakukan konfigurasi pada docker containers untuk *cloud gaming* dengan cara *install Nvidia graphics driver* untuk mendukung *GPU acceleration* pada *instance*.

4. Tahap Konfigurasi *Server Gaminganywhere* Pada Docker Containers

Dalam proses konfigurasi *server gaminganywhere* pada Docker Containers yaitu membuat sebuah docker *file* yang berisi semua pengaturan instalasi yang dibutuhkan dalam membangun sebuah docker *image* dari *gaminganywhere open source frame work*, setelah docker *file* dan docker *image* terbentuk. Setelah itu lakukan konfigurasi pada docker *file*.

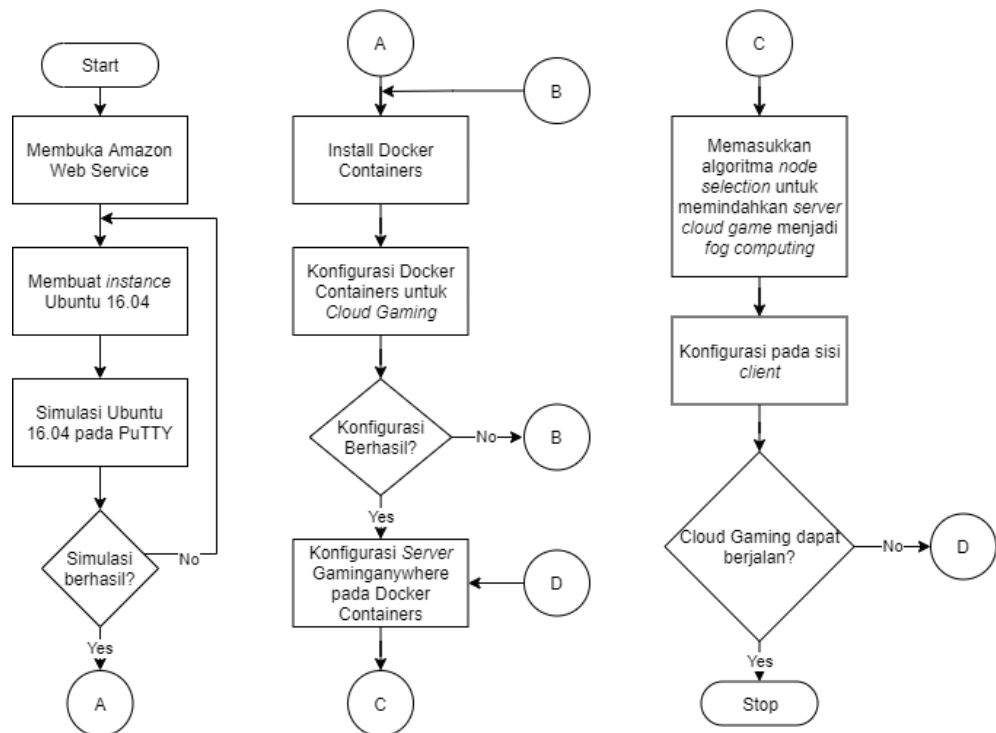
##### 5. Tahap Memindahkan *Server Cloud Game* Menjadi *Fog Computing*

Dalam proses memindahkan *server cloud game* menjadi *fog computing*, lakukan konfigurasi pada Docker Containers dengan memasukan algoritma *node selection* dimana algoritma ini akan mengidentifikasi *fog nodes* yang ada di *edge server*.

##### 6. Tahap Konfigurasi Pada Sisi *Client*

Dalam proses melakukan konfigurasi pada sisi *client* yaitu *download pre-build* dari *gaminganywhere file* dan ubah konfigurasi *client* berdasarkan *game setup*. Pastikan RTSP *Client* pada kondisi *open* untuk menerima *video streaming* dari *fog server*.

Adapun *Flowchart* tahapan perancangan dari pembentukan *Cloud Gaming* ini pada Gambar 3.2



Gambar 3.2 Flowchart Tahapan Perancangan

### 3.3 Perancangan

Pada Proyek akhir ini akan dirancang sebuah *cloud gaming* dengan *server gaminganywhere* pada *docker containers* dengan teknologi *fog computing* untuk menggantikan *gaming* konvensional dengan judul “Implementasi dan Analisis *Cloud Gaming* dengan *Server Gaminganywhere* yang di bentuk menggunakan Docker Containers pada Teknologi *Fog Computing*”.

## BAB IV

### BENTUK KELUARAN YANG DIHARAPKAN

#### 4.1 Keluaran yang Diharapkan

Perancangan pada Proyek akhir akan dibuat memiliki bentuk keluaran yang diharapkan adalah dapat membuat sistem *cloud gaming* yang memiliki tingkat latensi yang rendah, memiliki tingkat FPS yang tinggi dan memiliki kualitas *video* ada dengan resolusi 720p bahkan lebih.

#### 4.2 Jadwal Pelaksanaan

Adapun jadwal pengerjaan Proyek akhir bisa dilihat pada tabel 4.2 sebagai berikut :

*Tabel 4.3 Jadwal Pelaksanaan*

Judul Kegiatan	Waktu						
	Nov	Des	Jan	Feb	Mar	Apr	Mei
Studi Literatur							
Perancangan dan Simulasi							
Pengujian							
Analisa							
Pembuatan Laporan							

## DAFTAR PUSTAKA

- [1] Amazon. (2021, January 05). *Amazon CloudWatch*. Diambil kembali dari <https://aws.amazon.com/id/cloudwatch/>
- [2] Amazon. (2021, January 05). *Komputasi cloud dengan AWS*. Diambil kembali dari <https://aws.amazon.com/id/what-is-aws/>
- [3] Chun-Ying Huang, K.-T. C.-Y.-J.-H. (2014). *GamingAnywhere: The First Open Source Cloud Gaming System*.
- [4] Kurniawan, A. (2012). *Network Forensics Panduan Analisis & Investigasi Paket Data Jaringan Menggunakan Wireshark*. Yogyakarta.
- [5] Peter Mell, T. G. (2011). *The NIST Definition of Cloud Computing* .
- [6] Prabowo, T. H. (2020). *Penggunaan Virtual Machine Untuk Mengoptimalkan Kualitas Bermain Dengan Cloud Gaming*.
- [7] Pugalendhi, A. (2018). *Cloud Gaming System in Docker Container*.
- [8] PuTTY. (2021, January 05). *Download PuTTY*. Diambil kembali dari <https://www.putty.org/>
- [9] Ryan Shea, J. L.-H. (2013). *Cloud Gaming: Architecture and Performance*. 16-21.
- [10] S P Polyakov, A. P. (2017). *Docker Container Manager: A Simple Toolkit for Isolated Work with Shared Computational, Storage, and Network Resources*.
- [11] Setiawan, E. (2017). *Implementasi Dan Analisis Mobile Cloud Gaming Online Menggunakan Open-Source Cloud Gaming Server Gaminganywhere Pada Perangkat Android*.
- [12] Vishal Kumar, A. A. (2018). *Comparison of Fog Computing & Cloud Computing*. 31-41.
- [13] Yuhua Lin, H. S. (2015). *CloudFog: Towards High Quality of Experience in Cloud Gaming*.





**UNIVERSITAS TELKOM**  
**FAKULTAS ILMU TERAPAN**  
**KARTU KONSULTASI**  
**SEMINAR PROPOSAL PROYEK AKHIR**

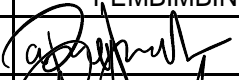
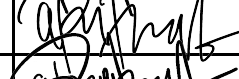

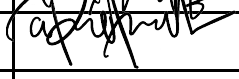
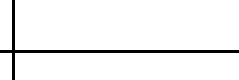



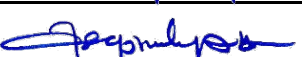

NAMA / PRODI : ANDRO ELNATAN HARIANJA / D3 TEKNOLOGI TELEKOMUNIKASI NIM : 6705184033

JUDUL PROYEK AKHIR :

IMPLEMENTASI DAN ANALISIS CLOUD GAMING DENGAN SERVER GAMINGANYWHERE YANG DI BENTUK MENGGUNAKAN DOCKER CONTAINERS PADA TEKNOLOGI FOG COMPUTING

CALON PEMBIMBING : I. Rohmat Tulloh, S.T., M.T.

II. Asep Mulyana, S.T., M.T.

NO	TANGGAL	CATATAN HASIL KONSULTASI	TANDA TANGAN CALON PEMBIMBING I
1	22 Januari 2021	BAB 1 (SELESAI)	
2	22 Januari 2021	BAB 2 (SELESAI)	
3	22 Januari 2021	BAB 3 (SELESAI)	
4	22 Januari 2021	BAB 4 (SELESAI)	
5	22 Januari 2021	FINALISASI PROPOSAL	
6			
7			
8			
9			
10			
NO	TANGGAL	CATATAN HASIL KONSULTASI	TANDA TANGAN CALON PEMBIMBING II
1	20 Januari 2021	BAB 1 (SELESAI)	
2	20 Januari 2021	BAB 2 (SELESAI)	
3	20 Januari 2021	BAB 3 (SELESAI)	
4	20 Januari 2021	BAB 4 (SELESAI)	
5	20 Januari 2021	FINALISASI PROPOSAL	
6			
7			
8			
9			
10			