1. **Latar Belakang**

*Game* merupakan suatu sistem yang memiliki aturan-aturan dimana pemain akan terlibat di dalam suatu permasalahan sehingga dapat menghasilkan suatu hasil yang dapat diukur yaitu menang atau kalah (Salen & Zimmerman, 2003).

*Gang Garrison 2* merupakan *game*  yang ber­-*genre shooting game* dimana pemain melawan pemain lainnya ataupun melawan *bot*, pada *game* tersebut tersedia 9 (sembilan) karakter. (http://www.ganggarrison.com/). Kemudian ada 7 model permainan yang tersedia namun yang akan digunakan dalam tugas akhir ini hanya 2 (dua) model permainan yaitu *generator dan control point.* Pada model *Generator*, pemain harus menghancurkan generator lawan sebelum generator pemain dihancurkan oleh lawan dalam waktu tertentu. Pada model *Control Point*, pemain diharuskan untuk mengambil semua *base* yang disediakan untuk memenangkan *game*. Total ada 5 (lima) *base* yang harus direbut dalam kurun waktu tertentu untuk memenangkan *game*, bila waktu habis maka akan dihitung tim yang lebih banyak merebut *base* yang menang. Namun dalam pengembangannya saat ini *bot* yang tersedia pada permainan tidak terlalu cerdas, *bot* masih kaku sehingga membuat pemain menjadi jenuh. *Bot* diharapkan dapat belajar dari pemain dan menyimpan langkah-langkah yang baik sehingga permainan menjadi tidak membosankan.

Dari permasalahan diatas, ada berbagai metode *artificial inteligence* dapat digunakan untuk menyelesaikan permasalahan *bot* yang kaku tersebut, salah satunya adalah *learning*. Salah satu metode dalam *learning* adalah *reinforcement learning.* *Reinforcement* *Learning* dianggap dapat menyelesaikan masalah bot yang kaku dan dapat menjadi solusi, karena *reinforcement* *learning* dapat belajar dari pengetahuan yang didapatkan dari interaksi antara sistem dengan lingkungan (Marco A. Wierinh and Jan Peter Patist, 2007). Salah satu algoritma yang akan menjadi solusi dalam metode *reinforcement learning* adalah SARSA (*State-Action-Reward-State-Action*), dimana algoritma ini lebih diklaim lebih baik dalam mengambil keputusan dari algoritma *reinforcement learning* yang lain.

Dalam tugas akhir ini, maka ingin di kembangkan sebuah program yang mengimplementasikan metode *reinforcement learning* untuk membuat *bot* dalam permainan bisa belajar dari pemain dengan berbagai percobaan sehingga *bot* dapat menentukan mana langkah terbaik yang dapat diambil sehingga dapat memenangkan permainan. Oleh karena itu, diambil tugas akhir yang berjudul "**Perkembangan Kecerdasan Bot pada Permainan *Gang Garrison 2* dengan *Reinforcement Learning***".

1. **Rumusan Masalah**

Adapun permasalahan dalam tugas akhir ini dapat dirumuskan sebagai berikut:

1. Tidak dinamisnya kecerdasan *bot* dalam permainan ini membuat tantangan menjadi rendah, sehingga pemain mudah jenuh.
2. Sistem yang dikembangkan belum menggunakan sistem cerdas yang mampu mempelajari strategi bermain dari pemain.
3. **Tujuan**

Adapun tujuan dari tugas akhir ini adalah menerapkan perbaikan terhadap cara kerja kecerdasan buatan *bot* dalam *gang garrison 2*.

1. **Manfaat**

Adapun manfaat dari tugas akhir ini adalah:

1. Meningkatkan kecerdasan *bot* sehingga *bot* dapat setara dengan pemain.
2. Menjadi bahan pembelajaran lebih lanjut mengenai *reinforment learning* serta menjadi bahan referensi untuk pengembangan selanjutnya.
3. **Batasan Masalah**

Adapun batasan masalah dari tugas akhir ini adalah:

1. Menggunakan 2 (dua) model permainan dari 7 (tujuh) model permainan yang tersedia pada ssistem.
2. Penggunaan Bot maksimal 3 bot.
3. Menggunakan SARSA dari metode *reinforcement learning*.
4. **Metodologi Pengembangan Sistem**

Dalam pelaksanaan tugas akhir ini, langkah-langkah yang dilakukan adalah sebagai berikut:

1. Pengumpulan referensi yang berhubungan dengan algoritma *reinforment learning* dan referensi-referensi lain yang berhubungan dengan tugas akhir.
2. Pengembangan sistem yang menggunakan metode *waterfall*. Adapun tahapan-tahapan dalam metode *waterfall* adalah :
   1. Analisis

Pada tahap ini mencakup analisis proses, pemodelan sistem identifikasi persyaratan fungsional dari sistem yang akan dikembangkan. Analisis fungsional sistem akan digambarkan dalam bentuk *class diagram*.

* 1. Desain

Pada tahap ini akan menterjemahkan syarat kebutuhan pengembangan perangkat lunak. Tahapan proses ini berfokus pada struktur data.

* 1. Pengkodean

Pada tahap ini akan dilakukan implementasi dari analisis dan desain dengan menggunakan *coding* GML (Game Maker Language).

* 1. Pengujian

Proses pengujian dilakukan dengan cara menjalankan algoritma RL (*reinforcement learning)* sebanyak 1.000 kali permainan tanpa menghapus memori yang sudah tersimpan. Setelah 1.000 kali permainan, memori itu dihapus dan *bot* mulai belajar dari awal. Kemudian

1. Menarik kesimpulan berdasarkan hasil pengujian terhadap program dan menyusun laporan tugas akhir.
2. **Tinjauan Pustaka**

**7.1. Game**

*Video game* merupakan salah satu media untuk menghibur diri yang paling popular untuk semua usia. Sejak pertama kali ditemukan sampai sekarang, teknologi *game* telah mengalami kemajuan. Hal ini ditandai dengan berkembangnya jenis, produk, alat, dan jenis interaksi *game*.

*Game* merupakan sebuah bentuk interaksi dimana pengguna disebut dengan pemain (*player*), diharuskan membuat keputusan-keputusan dengan tujuan untuk mengelola sumber daya yang diperoleh dari kesempatan-kesempatan bermain miliknya, untuk mencapai suatu tujuan tertentu.

Berdasarkan representasi visualnya, *game* dapat dibedakan menjadi 2 dimensi (2D) dan 3 dimensi (3D). *Game* 2D adalah *game* yang hanya melibatkan 2 elemen koordinat kartesius, yaitu x dan y, sehingga konsep kamera pada *game* 2D hanya menentukan gambar mana pada game yang dapat dilihat oleh pemain. Sedangkan *game* 3D melibatkan elemen x, y, dan z pada perhitungannya, sehingga konsep kamera pada *game* 3D benar-benar menyerupai konsep kamera pada kehidupan nyata.

*Game* secara umum dapat diklasifikasikan berdasarkan *platform* dan *genre*. *Platform* merupakan tempat (wadah) dimana *game* dijalankan, berikut *game* yang diklasifikasikan berdasarkan *platform*:

1. *Arcade games*

Memiliki box atau mesin yang memang khusus di design untuk jenis video games tertentu dan tidak jarang bahkan memiliki fitur yang dapat membuat pemainnya lebih merasa “masuk” dan “menikmati”, seperti pistol, kursi khusus, sensor gerakan, sensor injakkan dan stir mobil (beserta transmisinya tentunya).

1. PC *games*

Permainan yang dimainkan di komputer pribadi, bukan pada video game konsol atau mesin arcade.

1. *Console games*

Sebuah sistem komputer hasil kostumasi yang dapat menghasilkan gambar berupa sinyal, selanjutnya dapat diterjemahkan perangkat display (TV, monitor) menjadi video game, jadi dapat disimpulkan bahwa Game Console adalah komputer yang dikostumasi khusus untuk memainkan video game di TV atau monitor.

1. *Handheld games*

Mesin untuk memainkan game, bentuknya tidak lebih besar dari genggaman kedua tangan, mudah dibawa dalam genggaman, dan ringan.

1. *Mobile games*

Permainan yang ada pada telepon genggam.

(Aji dan Irfan Satya,2014)

Klasifikasi berdasarkan *genre* yaitu:

1. **Aksi-*Shooting***

*Video game* jenis ini sangat memerlukan kecepatan refleks, koordinasi mata-tangan, juga timing, inti dari game jenis ini adalah tembak, tembak dan tembak.

1. ***Fighting* (pertarungan)**

Jenis ini memang memerlukan kecepatan refleks dan koordinasi mata-tangan, tetapi inti dari game ini adalah penguasaan jurus, pengenalan karakter dan timing sangatlah penting, combo-pun menjadi esensial untuk mengalahkan lawan secepat mungkin. Dan berbeda seperti game Aksi pada umumnya yang umumnya hanya melawan Artificial Intellegence atau istilah umumnya melawan komputer saja, pemain jenis fighting game ini baru teruji kemampuan sesungguhnya dengan melawan pemain lainnya.

1. **Aksi-*Adventure***

Memasuki gua bawah tanah, melompati bebatuan di antara lahar, bergelayutan dari pohon satu ke pohon lain, bergulat dengan ular sambil mencari kunci untuk membuka pintu kuil legendaris, atau sekedar mencari telepon umum untuk mendapatkan misi berikutnya, itulah beberapa dari banyak hal yang karakter pemain harus lakukan dan lalui dalam video game jenis ini.

1. **Simulasi**

Video Game jenis ini seringkali menggambarkan dunia di dalamnya sedekat mungkin dengan dunia nyata dan memperhatikan dengan detil berbagai faktor.

1. ***Role Playing***

Video game jenis ini sesuai dengan terjemahannya, bermain peran, memiliki penekanan pada tokoh/peran perwakilan pemain di dalam permainan, yang biasanya adalah tokoh utamanya, dimana seiring kita memainkannya, karakter tersebut dapat berubah dan berkembang ke arah yang diinginkan pemain ( biasanya menjadi semakin hebat, semakin kuat, semakin berpengaruh, dan lain-lain) dalam berbagai parameter yang biasanya ditentukan dengan naiknya **level**, baik dari status kepintaran, kecepatan dan kekuatan karakter, senjata yang semakin sakti, ataupun jumlah teman maupun mahluk peliharaan.

1. Strategi

Kebalikan dari *video game* jenis *action* yang berjalan cepat dan perlu refleks secepat kilat, *video game* jenis strategi, layaknya bermain catur, justru lebih memerlukan keahlian berpikir dan memutuskan setiap gerakan secara hati-hati dan terencana.

1. *Puzzle*

*Video game* jenis ini sesuai namanya berintikan mengenai pemecahan teka-teki, baik itu menyusun balok, menyamakan warna bola, memecahkan perhitungan matematika, melewati labirin, sampai mendorong-dorong kota masuk ke tempat yang seharusnya, itu semua termasuk dalam jenis ini.

(Nurul Iman ,2013)

*Genre game* yang ada pada tugas akhir ini adalah Aksi-*Shooting* dan juga dalam 2D (dua dimensi). Nama *game* yang digunakan pada tugas akhir ini adalah *Gang Garrison 2*.

**7.2. *Artificial Intelligence***

Kecerdasan buatan atau *Artificial Intelligence* merupakan salah satu ilmu pengetahuan berhubungan dengan pemanfaatan mesin untuk memecahkan masalah atau mempermudah manusia dalam mengerjakan tugas atau menyelesaikan suatu masalah yang sulit untuk diselesaikan oleh manusia.

*Artificial Intelligence* adalah agen-agen yang ada untuk belajar melalui lingkungan dan persepsi dan tindakan (Stuart J. Russell and Peter Norvig, 1995).

Ada beberapa tipe agen *Artificial Intelligence*:

* + - 1. *Reasoning*

*Reasoning* adalah serangkaian proses yang memungkinkan untuk menyediakan dasar untuk penilaian, membuat keputusan, dan prediksi.

* + - 1. *Learning*

*Learning* adalah kegiatan memperoleh pengetahuan atau keterampilan dengan mempelajari, berlatih, diajarkan, atau mengalami sesuatu. *Learning* meningkatkan pengetahuan ke agen yang diteliti.

*Learning* dapat dikategorikan sebagai berikut:

* *Auditory Learning*

*Auditory Learning* belajar dengan cara mendengarkan dan pendengaran.

* *Episodic Learning*

*Episodic Learning* belajar dengan cara mengingat urutan peristiwa yang satu telah menyaksikan atau mengalami. Hal ini sejalan dan teratur.

* *Motor Learning*

*Motor Learning* belajar dari keakuratan gerakan dari suatu motor.

* *Observational Learning*

*Observational Learning* belajar dengan mengamati dan meniru orang lain.

* *Perceptual Learning*

*Perceptual Learning* belajar untuk mengenali sesuatu yang telah terlihat/terjadi sebelumnya.

* *Relational Learning*

*Relational Learning* belajar untuk membedakan antara berbagai input berdasar sifat relasional, bukan sifat mutlak.

* *Spatial Learning*

*Spatial Learning* belajar melalui input visual seperti gambar, warna, peta, dll untuk diolah.

* *Stimulus-Response Learning*

*Stimulus-Response Learning* belajar untuk melakukan perilaku tertentu ketika pemicu tertentu hadir

* + - 1. *Problem Solving*

*Problem Solving* adalah proses melakukan percobaan untuk mencapai solusi yang diinginkan dari situasi sekarang dengan cara mengambil beberapa jalur, baik rintangan yang diketahui atau tidak diketahui.

* + - 1. *Perception*

*Perception* adalah proses memperoleh, menafsirkan, memilih, dan mengatur informasi sensorik.

* + - 1. *Linguistic Intelligence*

*Linguistic Intelligence* merupakan kemampuan menyerupai seseorang dalam menggunakan, memahami, berbicara, dan menulis bahasa verbal dan tertulis. Hal ini penting dalam komunikasi interpersonal.

(<http://www.tutorialspoint.com/artificial_intelligence/artificial_intelligent_systems.htm>, tanggal akses 07 April 2016)

*Machine learning* merupakan bagian dari disiplin ilmu matematika lanjutan yang mencakup pemodelan yang canggih, optimasi, dan pembelajar untuk penelitian. Algoritma *machine learning* dibagi menjadi beberapa subkategori, yakni *supervised learning, unsupervised learning*, *semi-supervised learning,* *transduction*, *reinforcement learning*, dan *developmental learning*. (Patrick Hall dan Jared Dean,2014)

**7.3. Algoritma yang diajukan**

Kita tahu agen dapat belajar untuk bermain catur dengan diawasi belajar, dengan diberikan contoh-contoh situasi permainan bersama dengan langkah terbaik untuk situasi itu. Tapi jika tidak ada guru yang memberikan contoh, apa yang bisa agen lakukan? Dengan mencoba langkah acak, agen akhirnya dapat membangun model prediksi lingkungannya: seperti apa nantinya papan catur setelah langkah itu diberikan, dan bahkan bagaimana kemungkinan lawan untuk membalas dalam situasi yang diberikan. Tapi tanpa adanya beberapa umpan balik seperti apa yang baik dan apa yang buruk, agen tidak memiliki alasan untuk memutuskan langkah yang akan diambil. Untungnya, agen bermain catur menerima beberapa umpan balik, bahkan tanpa guru di akhir pertandingan, agen menerima apakah telah menang atau hilang. jenis umpan balik disebut *reward*, atau *reinforcement*.

*Reinforcement* *Learning* dapat belajar dari pengetahuan yang didapatkan dari interaksi antara sistem dengan lingkungan (Marco A. Wierinh and Jan Peter Patist, 2007).

Secara umum algoritma *Reinforcement learning* menggunakan *Markov Decision Process(MDP)*.(Nathan Sprague dan Dana Ballard,2003)

Ada beberapa aturan algoritma *Markov Decision Process* yang dapat digunakan dalam *Reinforcement learning* yakni *Q-Learning* dan *State-Action-Reward-State-Action(SARSA)*

*Q-Learing* merupakan algoritma *off-policy* dimana agen belajar nilai pasangan *State-Action* yang tidak perlu berdasarkan dari tindakan yang dilakukan , karena *update* mereka dilakukan terlepas dari tindakan saat ini, melaikan berkaitan dengan tindakan yang dapat memaksimalkan nilai dari pasangan *State-Action* berikutnya. Aturan pada *Q-Learning:*

𝑄𝑘+1(𝑠𝑡 , 𝑎𝑡 ) = 𝑄𝑘 (𝑠𝑡 , 𝑎𝑡 ) + 𝛼 [𝑟𝑡+1 + 𝛾 max 𝑄𝑘 (𝑠𝑡+1, 𝑎𝑡+1) − 𝑄𝑘 (𝑠𝑡 , 𝑎𝑡 )]

𝑎

Dimana α ∈ (0,1] adalah yang disebut *learning rate*. Hal ini menentukan tingkat *update*.

*State-Action-Reward-State-Action* (SARSA)merupakan algoritma *on-policy* dimana agen belajar nilai *State-Action* berdasarkan tindakan yang dilakukan. Dan akan dilakukan evaluasi pada kebijakan sekarang. Tidak seperti *Q-Learning* yang melakukan satu kebijakan dan mengevaluasi satu sama lain.

SARSA memperbarui aturan menjadi sebagai berikut:

𝑄𝑘+1(𝑠𝑡 , 𝑎𝑡 ) = 𝑄𝑘 (𝑠𝑡 , 𝑎𝑡 ) + 𝛼[𝑟𝑡+1 + 𝛾𝑄𝑘 (𝑠𝑡+1, 𝑎𝑡+1) − 𝑄𝑘 (𝑠𝑡 , 𝑎𝑡 )]

Dimana aksi 𝑎𝑡 + 1 adalah *action* yang dilakukan pada *state* 𝑠𝑡 + 1 berdasar pada kebijakan saat ini.( Marco Corazza dan Andrea Sangalli, 2015)

Aturan algoritma *Reinforcement Learning* yang akan diterapkan pada Bot adalah SARSA.

1. **Daftar Pustaka**

Stefan Wender, Ian Watson, 2012, **Applying Reinforcement Learning to Small Scale Combat in the Real-Time Strategy Game StarCraft:Broodwar.** http://geneura.ugr.es/cig2012/papers/paper44.pdf

Magnus Lindberg, 2014, **An Imitation-Learning based Agent playing Super Mario.** https://www.diva-portal.org/smash/get/diva2:831873/FULLTEXT01.pdf

Michael L. Littman, 2000, **Value-function reinforcement learning in Markov games.** http://www.cogsci.rpi.edu/~rsun/si-mal/article3.pdf

Xiaosong Lu, M.A.Sc., 2012, **Multi-Agent Reinforcement Learning in Games.** http://www.sce.carleton.ca/faculty/schwartz/RCTI/Seminar%20Day/Autonomous%20vehicles/theses/Lu.PhDECE.Spring12.pdf

**Gang Garrison 2** http://www.ganggarrison.com

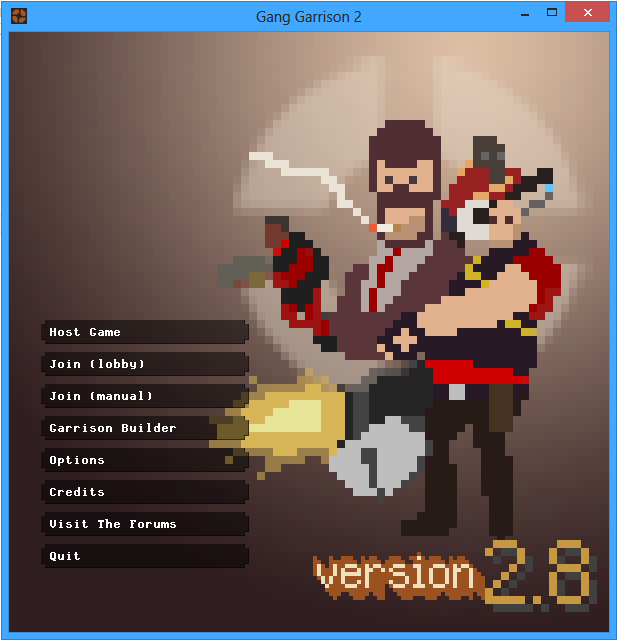
Yngvi Björnsson, 2007, **EFFICIENT USE OF REINFORCEMENT LEARNING IN A COMPUTER GAME.** http://www.ru.is/faculty/yngvi/pdf/bjornssonhjj04.pdf

Peng Ding and Tao Mao, 2009, **Reinforcement Learning in Tic-Tac-Toe Game and Its Similar Variations** http://www.cs.dartmouth.edu/~lorenzo/teaching/cs134/Archive/Spring2009/final/PengTao/final\_report.pdf

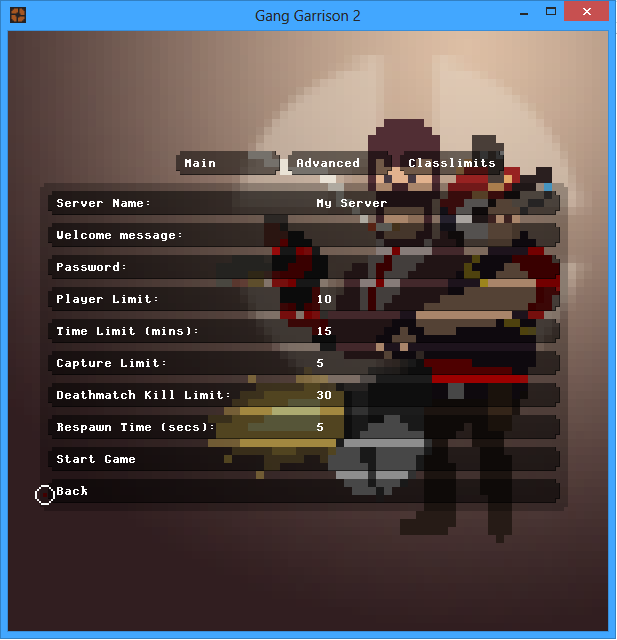
Russell S, Norvig P, Intelligence A. **A modern approach. Artificial Intelligence**. Prentice-Hall, Egnlewood Cliffs. 1995;25:27. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.259.8854&rep=rep1&type=pdf

1. **Lampiran**

Tampilan Menu Utama

****

Tampilan Host Game

**** Keterangan:

* Untuk memulai permainan dan menjadi host, pilih *Host Game*.
* Untuk mengikuti permainan dari host lain, pilih *Join*.
* *Garrison Builder* untuk membuat peta permainan baru.
* *Option* untuk mengatur suara dan visual dari game.
* *Credits* untuk memunculkan nama-nama orang yang berkontribusi dalam pengembangan game.
* *Visit the Forums* untuk mengunjungi forum dari game.
* *Quit* untuk keluar dari permainan.

Tampilan Pemilihan Tim

****

Keterangan : Tampilan untuk memilih bermain sebagai Tim Merah atau Tim Biru.

* + - 1. *Auto-Select* : Memilih tim yang akan dimasuki secara acak.
      2. *Spectate* : Mengamati permainan yang sedang berlangsung atau berperan sebagai penonton.
      3. *RED* : Memilih untuk bermain sebagai anggota Tim Merah.
      4. *BLU* : Memilih untuk bermain sebagai anggota Tim Biru.

Tampilan Pemilihan Karakter

****

Keterangan :

Menu untuk memilih satu dari antara sembilan karakter yang tersedia :

1. *The Runner*:

Pelari yang cepat, untuk merebut area, mencuri data dan memata-matai musuh.

1. *The Firebug*:

Petarung jarak dekat, memakai *Flamethrower* untuk membakar dan menghalau musuh

1. *The Rocketman*:

Penembak roket, melancarkan serangan dari jarak jauh dengan tembakan roket.

1. *The Overweight*:

Petarung barisan depan, lambat tetapi kuat dan sulit dibunuh, cocok untuk mempertahankan area.

1. *The Detonator*:

Pemasang jebakan, memasang ranjau dan meledakkannya ketika musuh mendekat.

1. *The Healer*:

Dokter yang menyembuhkan teman dan membantu mereka dari belakang.

1. *The Constructor*:

Teknisi yang dapat membuat *Autogun* atau *turret* untuk membantu mempertahankan area.

1. *The Infiltrator*:

Penyusup yang dapat menghilang dan efektif untuk membunuh lawan dari belakang.

1. *The Rifleman*:

Penembak jitu yang dapat menembak jatuh musuh dari jauh dengan senapannya.

Mode Permainanan *Generator*

****

Keterangan :

Peta dari Mode Permainan *Generator,* Tim Merah di bagian kiri dan Tim Biru di bagian kanan. Untuk penjelasan selanjutnya diambil dari pihak Tim Merah.

****

Keterangan :

*Starting Point* / Markas Tim Merah, tempat dimana anggota Tim Merah mulai bermain dan sebagai untuk *respawn* / hidup kembali jika terbunuh dalam permainan ini.

****

Keterangan:

*Generator* Tim Merah yang harus dilindungi. Terletak di depan Markas Tim Merah, Tim Merah harus melindungi *Generator* mereka dari serangan Tim Biru.

****

****

****

Keterangan:

Perjalanan dari *Generator* Tim Merah ke arah *Generator* Tim Biru yang melewati beberapa gedung.

****

Keterangan:

*Generator* Tim Biru yang harus dihancurkan oleh Tim Merah untuk mendapatkan kemenangan.

Mode Permainan *Control Point*



Keterangan:

Peta dari Mode Permainan *Control Point,* Tim Merah di bagian kiri dan Tim Biru di bagian kanan. Dalam Mode Permainan ini, ada lima *Point* atau area yang harus dimiliki oleh sebuah tim untuk memenangkan permainan. Pada awal permainan, setiap tim mendapatkan dua area yang sudah dikuasai masing-masing tim, sebuah area di tengah peta bersifat netral dan harus direbut untuk dapat merebut area berikutnya. Untuk penjelasan selanjutnya diambil dari pihak Tim Merah.



Keterangan :

*Starting Point* / Markas Tim Merah, tempat dimana anggota Tim Merah mulai bermain dan sebagai untuk *respawn* / hidup kembali jika terbunuh dalam permainan ini.



Keterangan:

*Point* / Area Pertama milik Tim Merah dan paling dekat dengan Markas Tim Merah.



Keterangan:

*Point* / Area Kedua milik Tim Merah dan dekat dengan *Point* / Area netral di tengah peta.







Keterangan:

*Point* / Area Ketiga yang bersifat netral dan direbut oleh Tim Merah.



Keterangan:

*Point* / Area Keempat milik Tim Biru dan direbut oleh Tim Merah.



Keterangan:

Tim Merah berhasil meraih kemenangan dengan menduduki semua *Point* / Area yang ada.