**LAPORAN TUGAS BESAR**

**SISTEM PARALEL DAN TERDISTRIBUSI**

**“*Game Multiplayer Shooter*”**



Disusun Oleh:

IF-40-06

Ahmad Arsyel Abdul H. 1301164193

Ayumi Rahmadani H. 1301164606

Fakry Adi Permana 1301164034

Galih Yudhasena T. 1301164451

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY**

**2019**

1. Peran Anggota

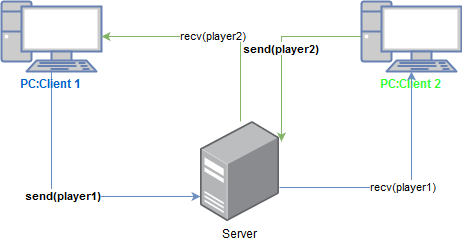
|  |  |  |
| --- | --- | --- |
| Nama | Peran | Tanda Tangan |
| Ahmad Arsyel Abdul H. | *Code*, Laporan |  |
| Ayumi Rahmadani H. | *Testing, Debugging* |  |
| Fakry Adi Permana | *Testing, Code* |  |
| Galih Yudhasena T. | Laporan, *Testing* |  |

1. Gambaran Umum Sistem

Sistem yang kami buat adalah *game multiplayer*, di mana dua *player* atau pemain dapat bermain *game* secara *realtime*. Pada *game shooter* ini akan ada dua kotak, yang setiap kotaknya digerakkan oleh masing-masing *player*. Untuk kotak di sebelah atas (ungu) adalah *player* 1 dan kotak di sebelah bawah (biru muda) adalah *player* 2. Untuk memenangkan permainan, salah satu *player* harus mengalahkan *player* lainnya dengan cara menembak peluru dengan tombol ‘spasi’ di keyboard. Selain itu, *player* juga bisa menggerakan kotak ke kanan atau ke kiri dengan tombol ‘panah kiri’ dan ‘panah kanan’ di keyboard untuk menghindari serangan. Setiap *player* masing-masing diberikan 3 nyawa.

Sistem ini menggunakan metode *Interprocess Communication* (IPC) *client*-*server* dengan *object message passing*, di mana *client* merupakan *player* yang mengatur pergerakan kotak dan *server* sebagai jembatan terjadinya pertukaran data permainan, juga terdapat *thread* untuk masing-masing *client* pada *server*.

1. Rancangan Sistem
   1. Arsitektur Sistem



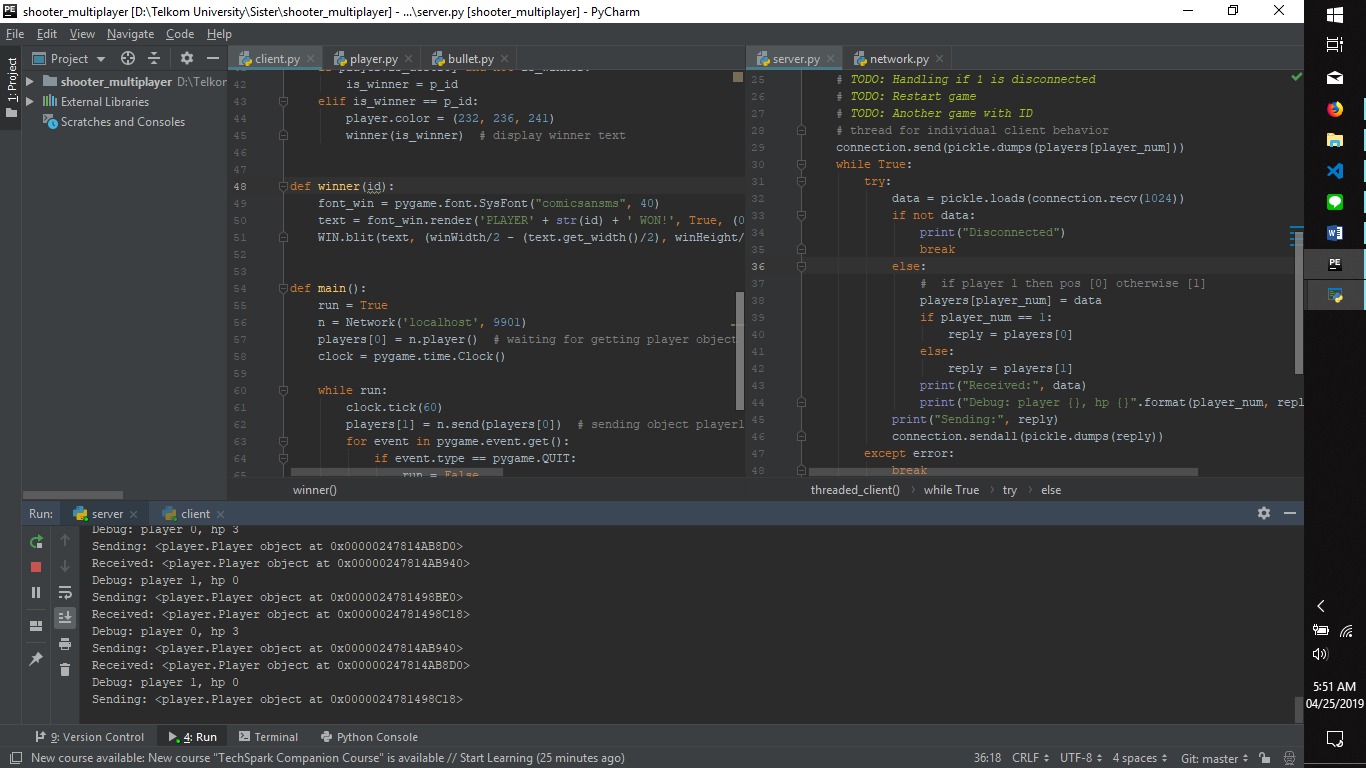
Gambar 1. Arsitektur sistem untuk *game multiplayer*.

Pada sistem ini, *client* 1 akan membangun koneksi terhadap *server*, sehingga *server* menerima data *object* yang dikirimkan oleh *client* 1. Kemudian *server* menunggu koneksi dari *client* 2. Ketika dua *client* telah dikenali, jalur pengiriman data tiap *client* akan dipisah. *Client* 1 akan terus menerima data terbaru dari object *client* 2, dan sebalikanya. Lalu, pygame akan melakukan *rendering* ulang terhadap *object client* yang memuat posisi dari gambar, sehingga data tiap *client* pada masing-masing tampilan layar akan *update*.

* 1. *Environment System Development*

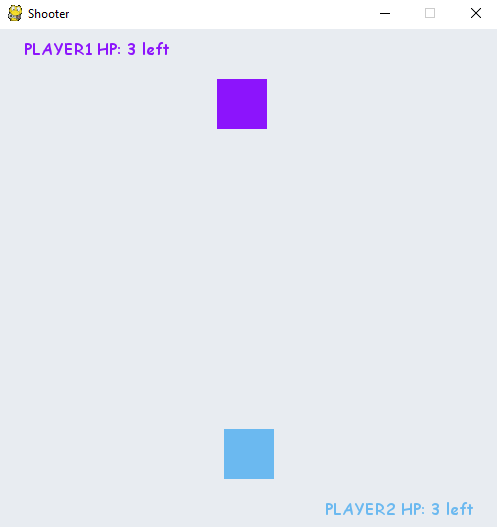
Sistem diimplementasikan menggunakan bahasa Python dengan *Integrated Development Environment* (IDE)PyCharm, library tambahan dari Python untuk pengelolaan gambar, yaitu Pygame, lalu menggunakan *library* Pickle untuk membaca kode *object* yang dijadikan pesan komunikasi.

1. Hasil



Gambar 2. *Server* running.

Gambar 2 merupakan hasil *running* sebagai *server*. Antar-*client* akan saling berkomunikasi melemparkan pesan *object*. Setelah 2 *client* telah terhubung dan diidentifikasi oleh *server*, maka *client* dapat langsung mengakses logika sistem yang telah dibangun.



Gambar 3. Rancangan *game*.

Gambar 3 merupakan gambaran rancangan *game*. *Player* 1 sebagai *client* pertama dapat langsung berinteaksi secara *realtime*, dan *client* lain atau *player* 2 dapat mengetahui kondisi terbarunya. Masing-masing *client/player* memiliki batasan 3 nyawa untuk bertahan di *game* dalam menerima serangan lawan.