

Relatório Trabalho 4

PUC-Rio Sistemas Distribuídos 2018.1

Renan Almeida (1812805) e Matheus Ambrozio (1812794)

O jogo desenvolvido é um *top-down multiplayer shooter*. Cada jogador controla um avatar na forma de um quadrado. Cada avatar possui um círculo colorido indicando a direção para onde o jogador está mirado. Um jogador controla seu avatar com as teclas WASD do teclado, podendo atirar ao pressionar a barra de espaço. O objetivo do jogo é eliminar um outro jogador. O primeiro jogador a eliminar outro jogador ganha. Cada jogador possui 10 vidas e 100 munições. A qualquer momento um jogador pode pressionar ESC para sair do jogo.

O código contém duas classes, *Player* e *Projectile*, que sabem como se desenhar no canvas e atualizar seu estado dados os comandos do jogador. O arquivo *main.lua* gerencia todos os outros aspectos do jogo. Os arquivos *mqtt_library.lua* e *utility.lua* são parte da *lib* para Lua do Mosquitto. As mensagens trocadas nos canais do MQTT são dados serializados de jogadores. Os canais de comunicação definidos são:

- **trabalho4/sd/keypressed:** Para quando um jogador efetua uma ação. A alteração de estado é feita dentro de cada objeto da classe *Player*. O jogo possui uma tabela com todos os *players* instanciados.
- **trabalho4/sd/keyreleased:** para quando o jogador libera uma tecla. Similar ao canal *keypressed*.
- **trabalho4/sd/joined:** para quando novos jogadores entram no jogo. Os jogadores trocam informações entre si sobre seu estado atual.
- **trabalho4/sd/unjoined:** para quando um jogador deixa o jogo (seu avatar deixa de existir).

Para iniciar o jogo a partir da linha de comando o jogador deve executar “*love . username host port*”. *Username* é o nome do jogador e *host* e *port* indicam o endereço do servidor Mosquitto.

Para testar a escalabilidade do MQTT, criamos um script em que um programa (o *receiver*), recebia mensagens de 100 outros (*senders*) paralelamente. Cada *sender* enviava 100 mensagens antes de terminar. O teste executava numa média de tempo de 10 segundos (1000 mensagens por segundo). Buscamos simular um ambiente onde muitos jogadores conversam com uma espécie de servidor e percebemos que a latência pode se tornar um problema dependendo das exigências do jogo. Alterando um pouco os parâmetros e mandando mais mensagens, não conseguimos fazer com que uma mensagem se perdesse.

Os arquivos do jogo encontram-se na pasta *game*. O script de teste e o código do *sender* e do *receiver* encontram-se na pasta *tests*.