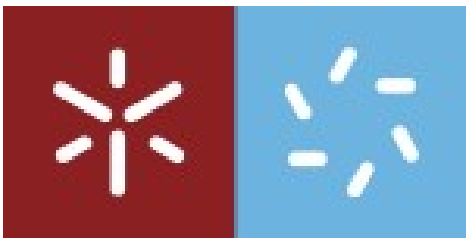

Universidade do Minho

Programação Concorrente - Relatório

Grupo 8

Ana Beatriz Silva (91678) Eduardo Soares (95917) Francisco Paulino (91666)
Pedro Viana(98979)

Maio 2025

Conteúdo

Introdução

No âmbito da unidade curricular Programação Concorrente, foi proposto um projeto cujo objetivo é a implementação de um mini-jogo. A interface gráfica deve ser escrita em Java, enquanto que o Servidor do mesmo deve ser feito em Earlang. O mini-jogo permite a interação de múltiplos jogadores, que se movimentam num espaço 2D; estes podem interagir entre eles próprios e com o ambiente criado.

Cliente

0.1 Player

A classe *Player* é onde são organizadas as informações relativas aos jogadores, como o nome, o número de vitórias e derrotas, e também o seu nível.

0.2 Shooter

A classe *Shooter* trata da parte gráfica do player.

0.3 Modifier

A classe *Modifier* aborda as informações sobre cada modifier, como cor e raio, e define o método para desenhar o mesmo. Cada tipo de modifier tem a sua cor e tem o seu respetivo impacto no player.

0.4 Bullet

Identicamente à classe anterior, *Bullet* possui os dados relativos às mesmas.

0.5 Connector

Esta é a classe onde ocorre a ligação com o servidor, de forma a que seja possível a leitura e escrita entre o cliente e servidor para se poder concretizar o desenho do jogo.

0.6 Game

A classe *Jogo* faz uma gestão dos Shooters, Modifiers e das Bullets. Esta classe contem um método *update*, que atualiza o jogo de uma forma segura, usando um *lock()*. Finalmente, temos também um método *draw* que chama as funções de desenho dos shooters, modifiers e das bullets.

0.7 Client

A classe Cliente é onde foram criadas todas as funcionalidades do menu. Esta classe é também responsável por trabalhar com o *Connector0* para que a interligação com o Servidor seja feita com sucesso, de forma a que, por exemplo, quando ocorre um login no Cliente, este seja autenticado pelo Servidor.

Servidor

0.8 Collision

Neste módulo, estão presentes todas as funções que lidam com os diferentes casos de colisões possíveis.

0.9 Conversor

Como o nome sugere, no *Conversor* temos as funções que são usadas para, ao longo do programa, ajudar na conversão de tipos de dados.

0.10 Login Manager

No *login_manager* temos as diferentes funções usadas quando uma conta é criada ou removida por um jogador, e também quando este ganha/perde um jogo, em particular, o algoritmo criado para obter os valores corretos de vitórias e derrotas.

0.11 Modifier

Aqui vamos ter as diferentes funções que "lidam"com os modifiers, ou seja, funções para criar um novo modifier, escolher o tipo deste, remover um modifier baseado na sua posição, a conta de quantos modifiers temos de cada tipo, e uma função para atualizar a lista de modifiers existentes.

0.12 Player

Neste módulo, estão presentes as funções para a criação de um novo jogador, atualiza os valores alterados pelos modificadores, limita a velocidade dos jogadores, e tambem uma função para atualizar os buffers dos jogadores. Finalmente, o módulo *Player* é ainda responsavel por atualizar a pontuação do jogo/partida.

0.13 Projectile

No *Projectile* possuímos as função que criam os projetores, e ainda as funções que determinam se estes estão fora dos limites, e as que realizam a atualização das suas posições.

0.14 State

Este módulo garante que um jogo esta a correr com sucesso, fazendo as atualizações que acontecem nos players, projectiles, shooters e também colisões à medida que estas acontecem, e garantindo que estas atualizações são refletidas no jogo.

0.15 Server

No *Server*, é utilizado o *Login_Manager* para gerir aquilo que acontece no Login Temos também a função *cicloJogo()* que se encontra sempre à espera de menasgens eviadas pelo cliente.

Conclusão

Durante a realização do projeto, após uma inicial desorganização, fomos capazes de encontrar uma boa dinâmica de trabalho e unir os nossos esforços para realizar o trabalho presente. Não fomos capazes de implementar todas as funcionalidades em mente, e que eram pedidas pelos docentes, mas como grupo, estamos satisfeitos com o resultado final que apresentamos.

O projeto foi incrivelmente vantajoso para ajudar a consolidação do material lecionado ao longo do semestre e os desafios presentes neste ajudaram a desenvolver o nosso conhecimento sobre o material, em particular Erlang.