

**UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL**  
**CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**ANDREI POCHMANN KOENICH**  
**GUSTAVO SPELLMEIER NEVES**  
**TURMA U**

**RELATÓRIO – TRABALHO FINAL**

**Disciplina: Fundamentos de Computação Gráfica**  
**Professor: Eduardo Simões Lopes Gastal**

**Porto Alegre, março de 2023.**

## 1 Contribuições de cada membro da dupla para o trabalho

Andrei Pochmann Koenich: construção do cenário com piso, *skybox* e objetos com malhas poligonais complexas (alvos e esferas); mapeamento de texturas para aplicação nos objetos e na *skybox*; implementação de câmera livre no cenário do campo de tiro; implementação da câmera *look-at* no cenário de observação do troféu; criação de crosshair, com implementação da mira e o posicionamento da arma em coordenadas de tela; implementação da tela inicial, com apresentação do jogo, utilizando textura específica e posicionamento em coordenadas de tela; transformações geométricas nos projéteis de bala disparados, com translação e rotação adequados ao vetor *view* relacionado à câmera livre; criação das animações e da movimentação do jogador baseadas no tempo; implementação de modelos de Blinn-Phong (presente nas esferas) e difusa (presente nos demais objetos com modelo de iluminação); teste de intersecção ponto-cubo para destruição dos alvos; teste de intersecção ponto-esfera para destruição das esferas.

Gustavo Spellmeier Neves: construção do cenário com objetos com malhas poligonais complexas (alvos, caixas e esferas); mapeamento de texturas para aplicação nos objetos do cenário; implementação do modelo de iluminação Blinn-Phong com modelo de interpolação de Gouraud nas esferas e modelo de iluminação difusa nos demais objetos; implementação de movimentação com curva Bézier Cúbica para um dos alvos do cenário; testes de intersecção linha-plano para limitar a movimentação do jogador no cenário envolvendo a câmera livre; testes de intersecção ponto-cubo envolvendo os projéteis de bala e as caixas e barras de concreto do cenário; testes de intersecção ponto-cubo envolvendo os projéteis de bala e os alvos.

## 2 Descrição do processo de desenvolvimento e do uso em sua aplicação dos conceitos de Computação Gráfica estudados

Os dois cenários do jogo consistem em um plano, os quais são cobertos por uma esfera que, por meio da aplicação das texturas, atua como *skybox* do cenário. No cenário inicial, existem quatro objetos diferentes: alvos, esferas, caixas de madeira e barras de concreto. Durante a movimentação nesse mesmo cenário. No cenário final do jogo, há a presença de um objeto representando um troféu. A criação dos cenários representa uma aplicação das aulas relacionadas com modelagens geométricas e transformações geométricas, uma vez que foi necessária a aplicação de operações de translação e escalamento em modelos geométricos 3D, para utilização no cenário (aulas 4, 6 e 7; laboratórios 1 e 2).

Para movimentação do jogador no cenário principal (contendo os alvos, para treinamento de mira), foi utilizada a implementação de uma *free camera*. No cenário final do jogo, utilizou-se uma *look-at camera*. Durante a movimentação com *free camera* no cenário principal do jogo, foi implementada uma mira (*crosshair*), e uma arma “segurada pelo jogador”, as quais são impressas diretamente em coordenadas de tela. Assim, foram utilizados os conceitos relacionados com a

implementação de câmeras e sistemas de coordenadas em Computação Gráfica (aula 8 e laboratório 2).

Foram usados mapeamentos de texturas em todos os objetos. Especificamente para as esferas relacionadas à implementação da *skybox*, foi utilizado o mapeamento de textura com projeção esférica. Para os demais objetos, utilizou-se projeção planar. Foram usados os conhecimentos obtidos nas aulas 20, 21 e no laboratório 5 da disciplina.

Em relação aos modelos de iluminação, foi implementado o modelo de iluminação Blinn-Phong nas esferas flutuantes do cenário (as quais também são alvos e devem ser destruídas pelos disparos do jogador), com modelo de interpolação de Gouraud. Para os demais objetos, foi utilizado o modelo de iluminação difusa, com referência à aula 18 e ao laboratório 4 da disciplina.

O jogo possui animações com movimentação linear (para movimentação dos alvos, do jogador e das esferas), além de uma animação baseada em uma curva de Bézier Cúbica. Todas as animações do jogo são baseadas em tempo, de acordo com o conteúdo visto nas aulas 16 e 24 da disciplina.

Por fim, o jogo também possui três tipos de intersecção: ponto-cubo, para os casos em que o projétil disparado pela arma do jogador atinge um alvo, ou para os casos em que esse projétil atinge um dos obstáculos (caixa de madeira ou barra de concreto); ponto-esfera, para os casos em que o projétil atinge uma das esferas-alvo do cenário principal; linha-plano, para limitar a movimentação do jogador no cenário principal. Para essas implementações, foi possível utilizar os fundamentos matemáticos da Computação Gráfica, abordados na aula 2 da disciplina.

### 3 Imagens mostrando o funcionamento da aplicação

Na figura 1, temos uma imagem da tela inicial da aplicação, com os nomes do jogo, dos integrantes do grupo e da disciplina. O jogo inicia a partir do momento em que o jogador pressiona a tecla ENTER, conforme a orientação da tela.



Figura 1 - Tela de início do jogo

Nas figuras 2 e 3, é possível ver o cenário principal do jogo, no qual é apresentada a mira (crosshair) a ser utilizada pelo jogador para acertar os alvos, a arma utilizada pelo jogador para efetuar os disparos e as esferas flutuando no cenário. É possível visualizar os dois tipos de alvo (o alvo presente no chão e os alvos flutuantes, representados por esferas), e os objetos que consistem em caixas de madeira e barras de concreto. Também é possível perceber a presença da *skybox*.



Figura 2 – Primeira visualização do cenário principal

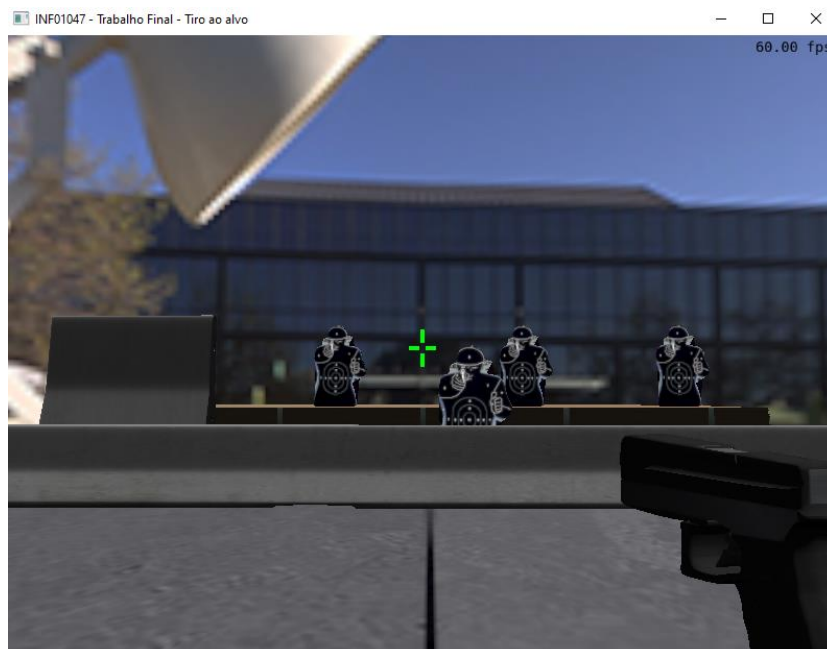


Figura 3 - Segunda visualização do cenário principal

Na figura 4, vemos o cenário final do jogo, contendo um troféu e uma *skybox* diferente, para onde o jogador é transferido ao destruir todos os alvos do cenário principal



Figura 4 - Visualização do cenário final do jogo

#### 4 Manual descrevendo a utilização da aplicação

Ao iniciar a aplicação, é necessário pressionar ENTER para iniciar o jogo. Após isso, o jogador é transferido para o cenário principal. Para realizar a movimentação no cenário principal (que é restrita pelas quatro barras de concreto presentes ao redor do jogador), é necessário pressionar as teclas W, A, S e D. Para efetuar um disparo, é necessário pressionar o botão esquerdo do mouse. O objetivo do jogo é destruir todos os alvos presentes no chão, e as esferas presentes mais acima, no ar. Caso um projétil atinja um obstáculo (uma caixa de madeira ou uma barra de concreto), esse projétil é destruído e não poderá atingir alvos localizados atrás do obstáculo atingido. Ao atingir todos os alvos, o jogador é transferido para o cenário da figura 4, indicando o término do jogo. É possível visualizar o troféu do cenário final do jogo sob diferentes ângulos da câmera *look-at* implementada, pressionando o botão esquerdo do mouse e realizando movimentações. Também é possível aproximar e afastar a câmera do troféu, utilizando o *scroll* do mouse. Durante qualquer ponto do jogo, é possível encerrá-lo pressionando a tecla ESC.

#### 5 Explicação de todos os passos necessários para compilação e execução da aplicação

A compilação e execução da aplicação é simples, e pode ser realizada da mesma forma utilizada nos laboratórios da disciplina, bastando utilizar a função "*build and run*" da IDE utilizada. Dependendo da IDE utilizada, pode ser necessário alterar a *build target* para compilação e execução. Os autores da aplicação fizeram uso da IDE CodeBlocks, com utilização da *build target* "Debug (CBlocks 17.12)".