VISEDU LIGHT: visualizador de ray tracing

Aluno(a): Daniel Rossato Martini

Orientador: Dalton Solano dos Reis

FURB
UNIVERSIDADE DE BLUMENAU

Roteiro

- Introdução
- Objetivos
- Fundamentação Teórica
- Trabalhos Correlatos
- Requisitos
- Especificação
- Implementação
- Resultados e Discussões
- Conclusões e Sugestões
- Demonstração

Introdução

- Apenas usado em renderizações *offline*.
- Poucos explorado para aplicações em tempo real;
- Nova ênfase em *ray tracing* em jogos;

Objetivos

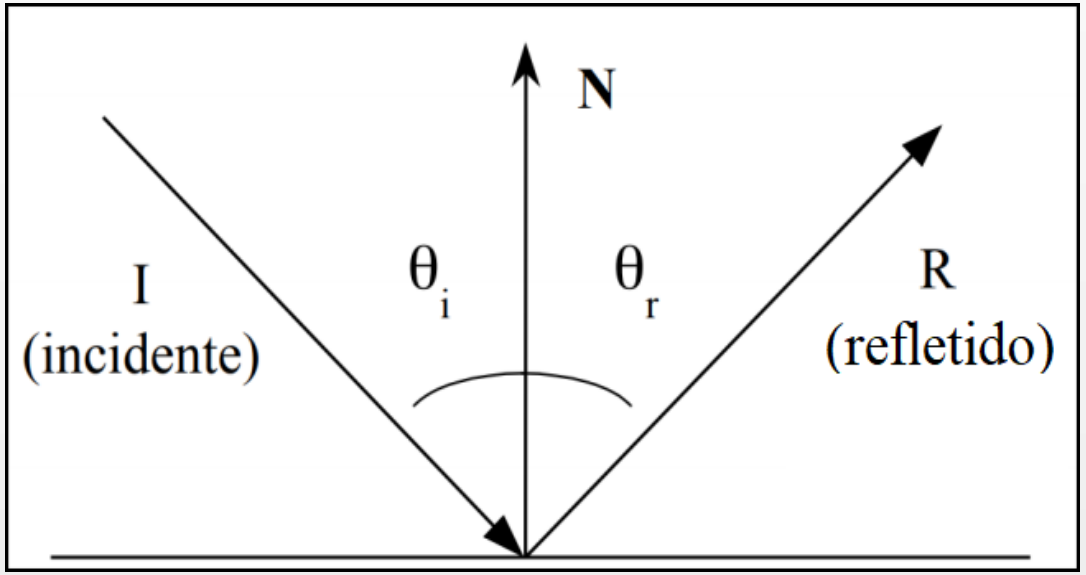
Desenvolver uma ferramenta para a visualização do *ray tracing*, com explicações das técnicas utilizadas.

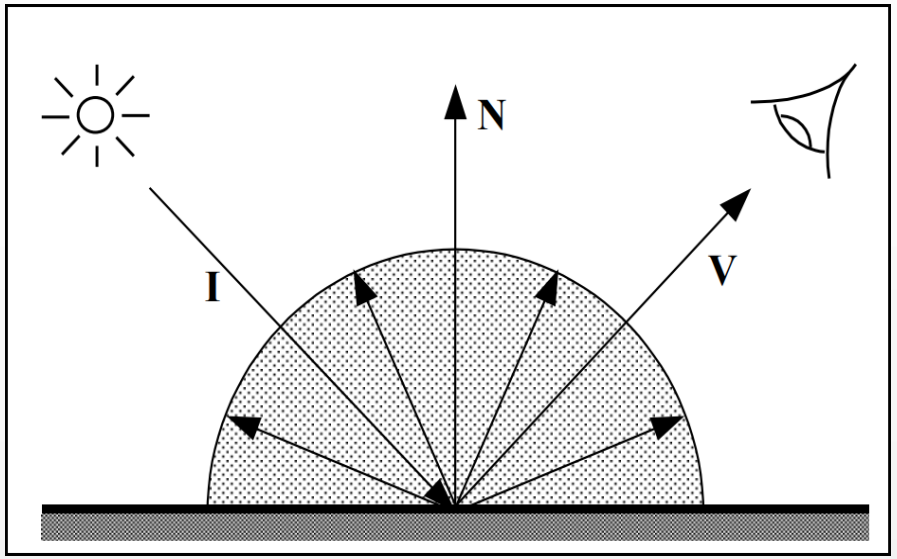
Objetivos Específicos

- Permitir o usuário alterar variáveis de ambiente;
- Criar três salas com diferentes técnicas em cada;
- Explicar cada etapa que o usuário passa;

Fundamentação Teórica – *Ray Tracing*

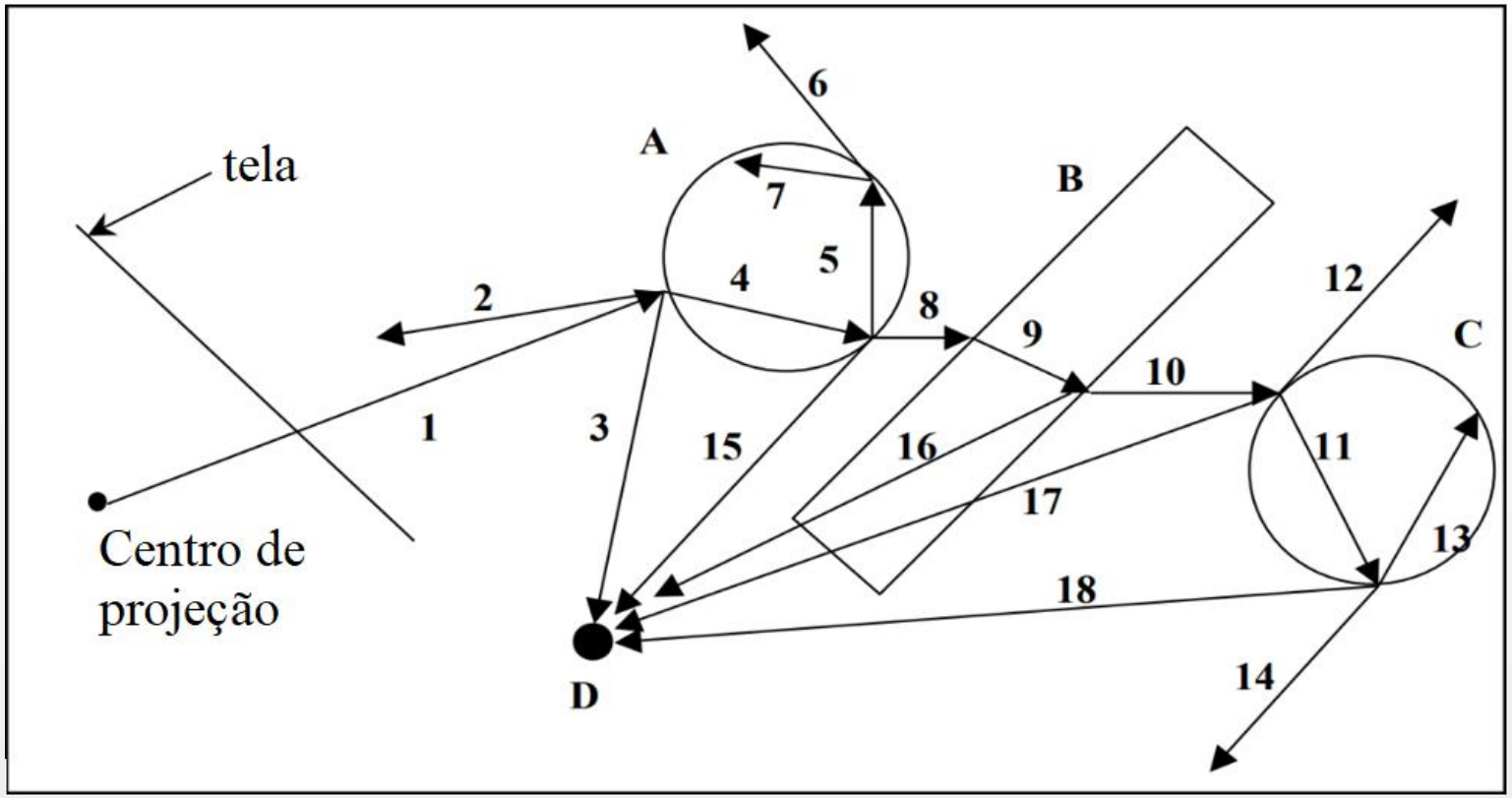
- Reflexão Perfeita;
- Reflexão Difusa;
- Refração;
- Sombras.





Fundamentação Teórica – *Path Tracing*

- Forma única de renderização;

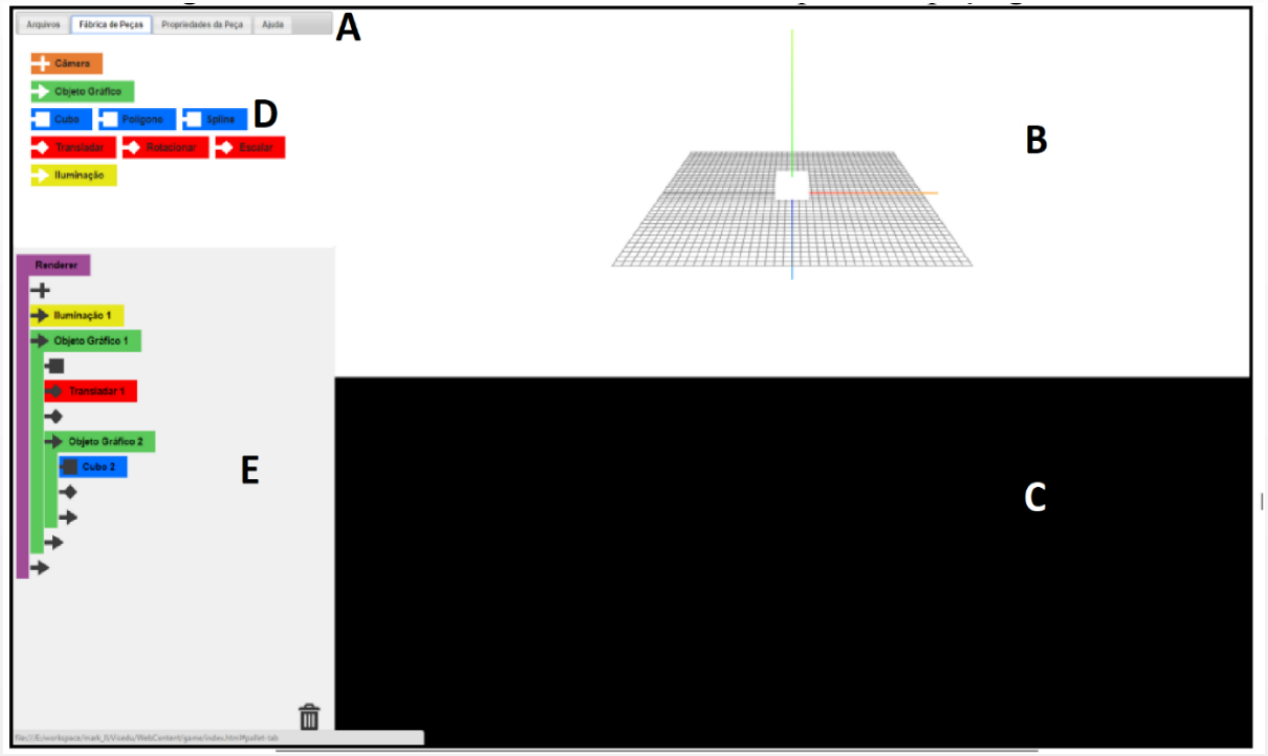


Fundamentação Teórica – *Denoiser*

- Integral ao *ray tracing*;
- Limpa a imagem gerada;
- Diversos tipos.

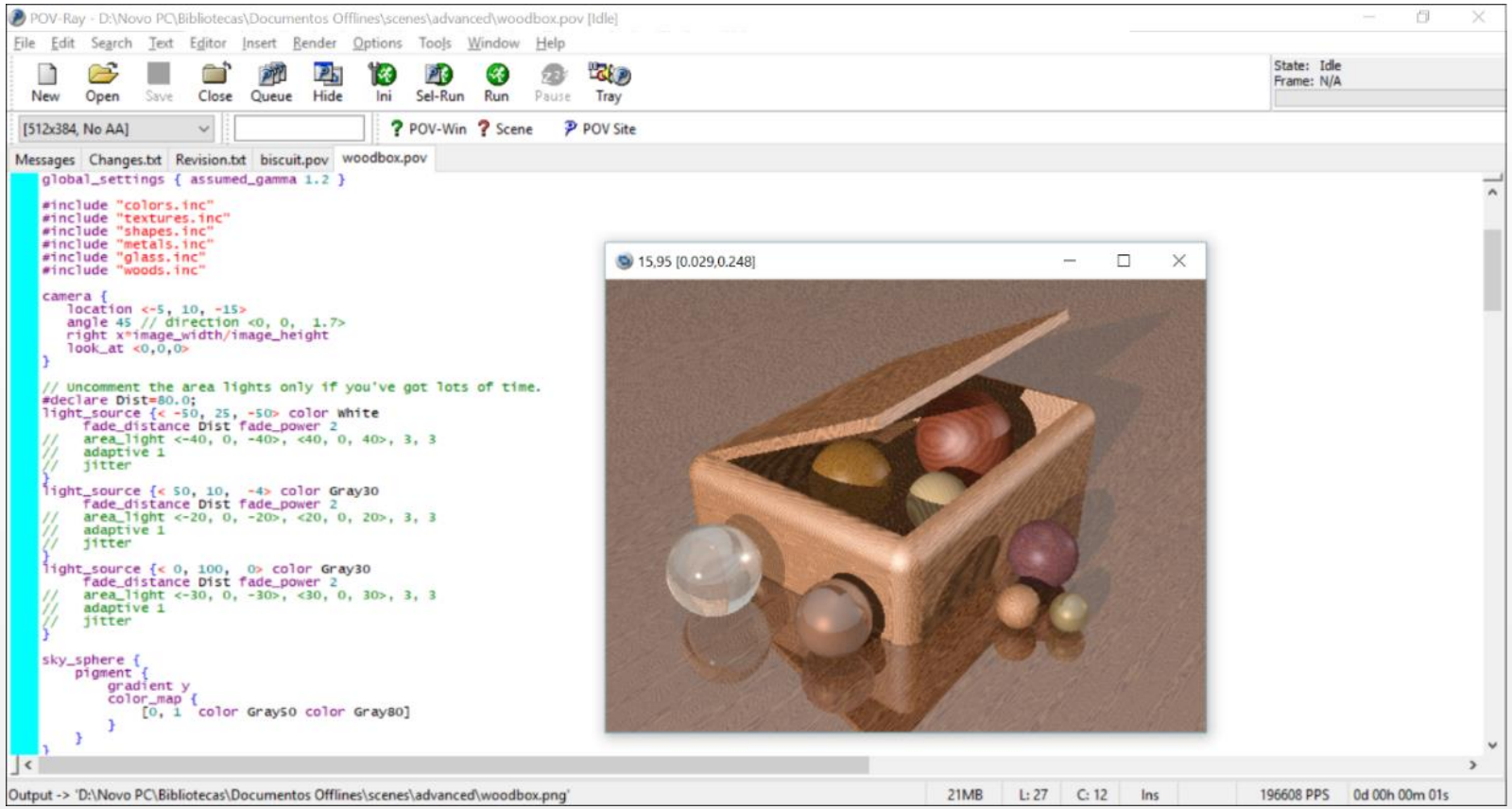
Trabalhos Correlatos

VISEDU-CG 4.0: visualizador de material educacional



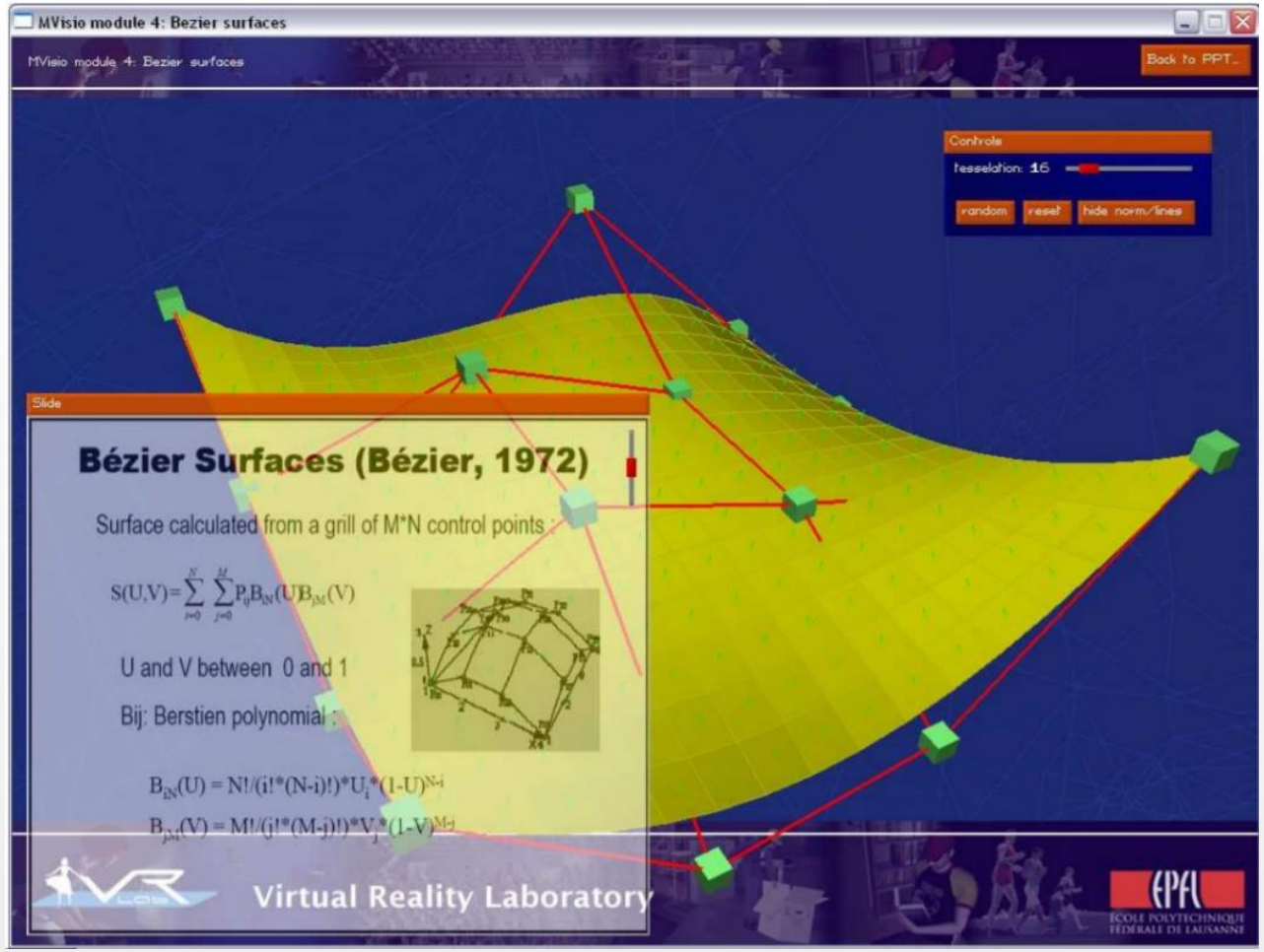
Trabalhos Correlatos

POV-Ray



Trabalhos Correlatos

Mental Vision



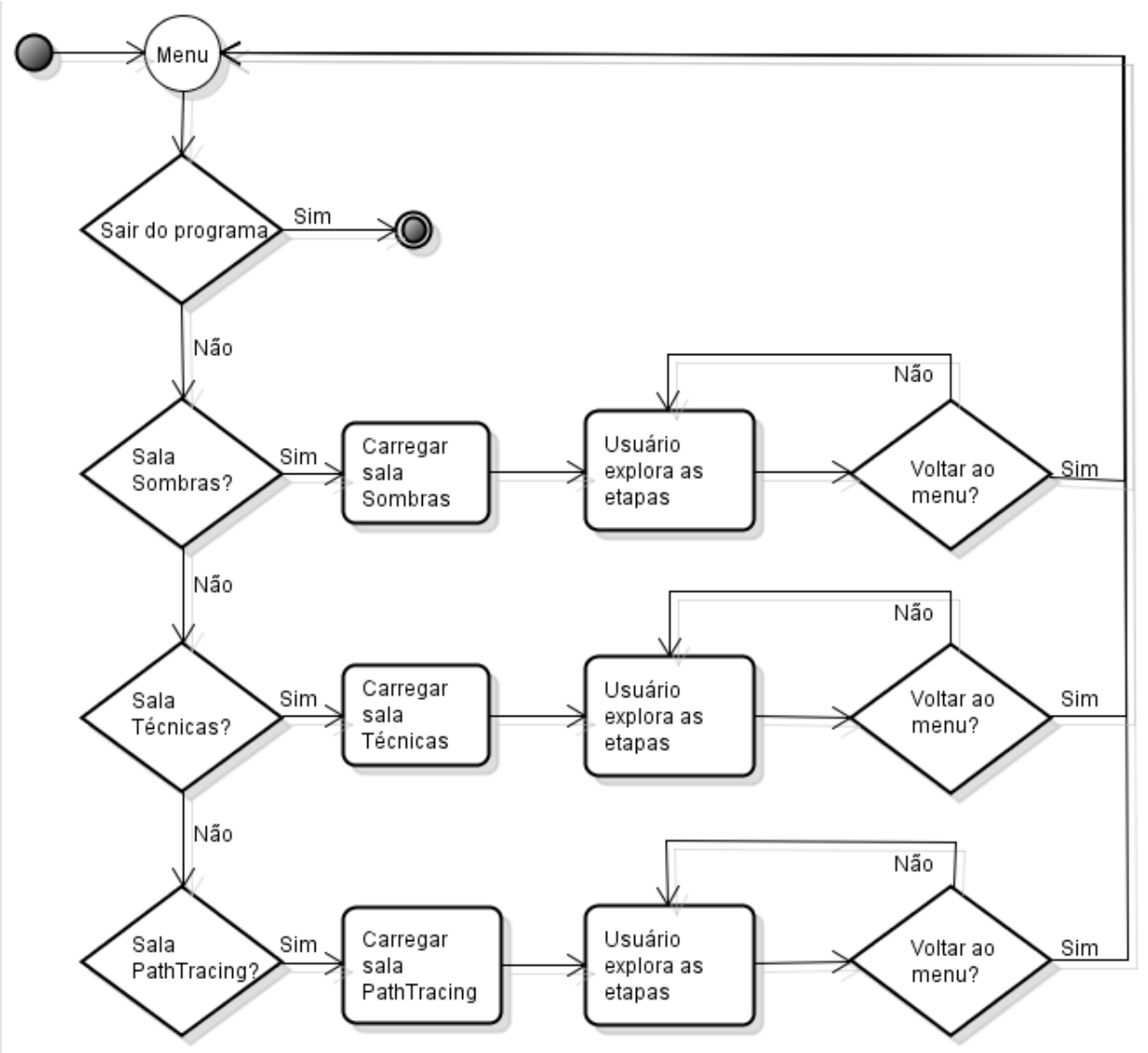
Requisitos

- Requisitos Funcionais:
  - possuir uma cena com um quarto branco, com dois cubos dentro e um holofote como fonte de luz apontando para um dos cubos;
  - possuir uma cena com um chão branco e uma parede branca ao fundo, um objeto cúbico e outro objeto esférico, um holofote apontando para a esfera e uma luz dispersa logo acima dos objetos;
  - possuir uma cena com dois quartos ligados por uma porta, com uma fonte de luz dispersa num dos quartos, com dois objetos esféricos no quarto com a luz, e um objeto cúbico no quarto sem a luz;
  - permitir o usuário alterar a textura e a cor dos objetos;
  - permitir o usuário ligar e desligar o *ray tracing;*
  - possuir duas telas de visualização, uma com a visão da câmera e outra em terceira pessoa mostrando a cena como um todo;
  - criar traços na visualização em terceira pessoa mostrando o caminho dos raios de luz;

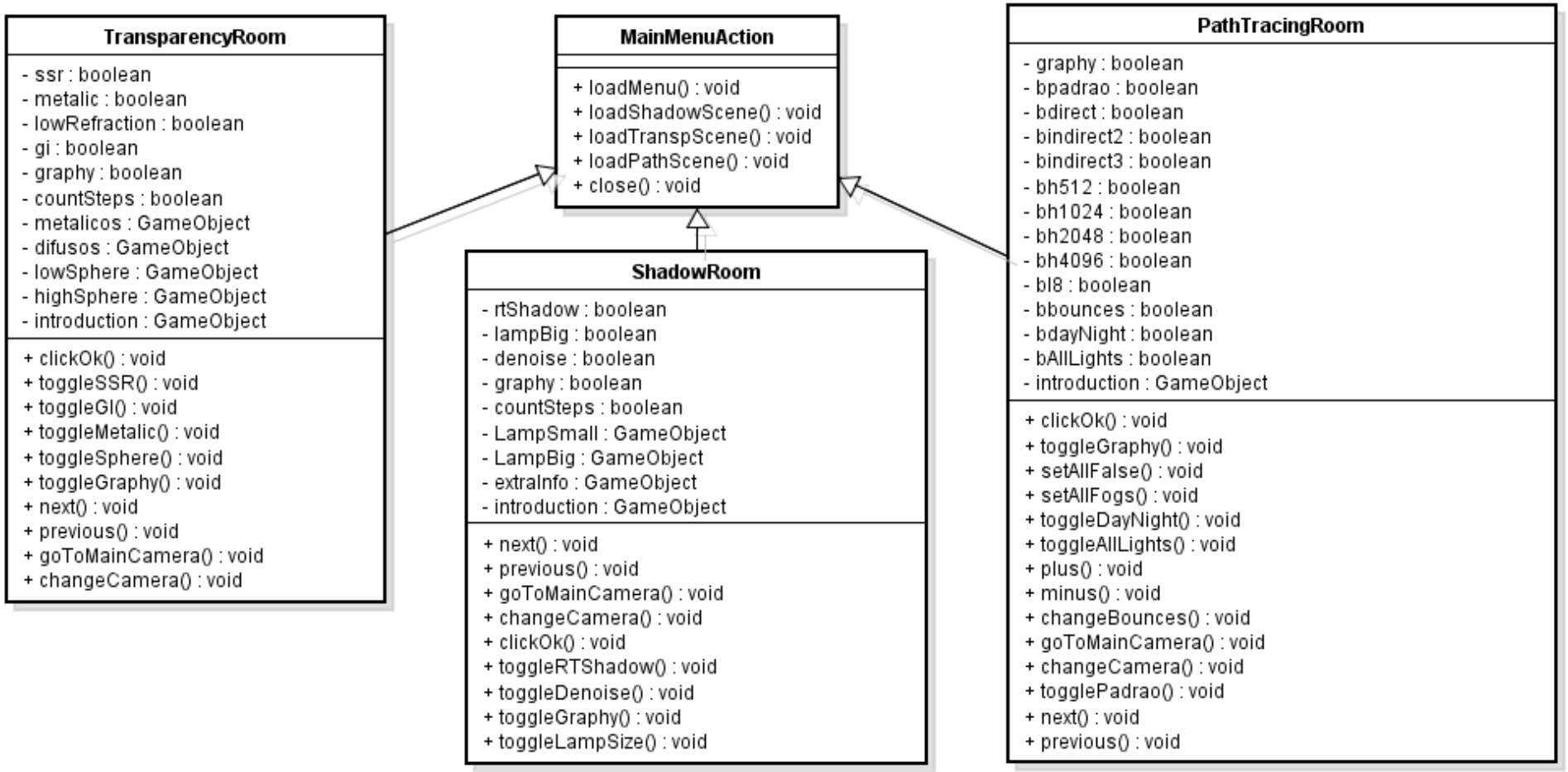
Requisitos

- Requisitos Não Funcionais:
  - criar três tipos de texturas, reflexiva, opaca e transparente;
  - possuir um menu que direcione para cada uma das cenas;
  - adicionar caixas de texto com explicações sobre como o *ray tracing* está sendo utilizado na cena;
  - permitir escolher três cores, vermelho, verde e azul;
  - utilizar aceleração de GPU quando disponível.

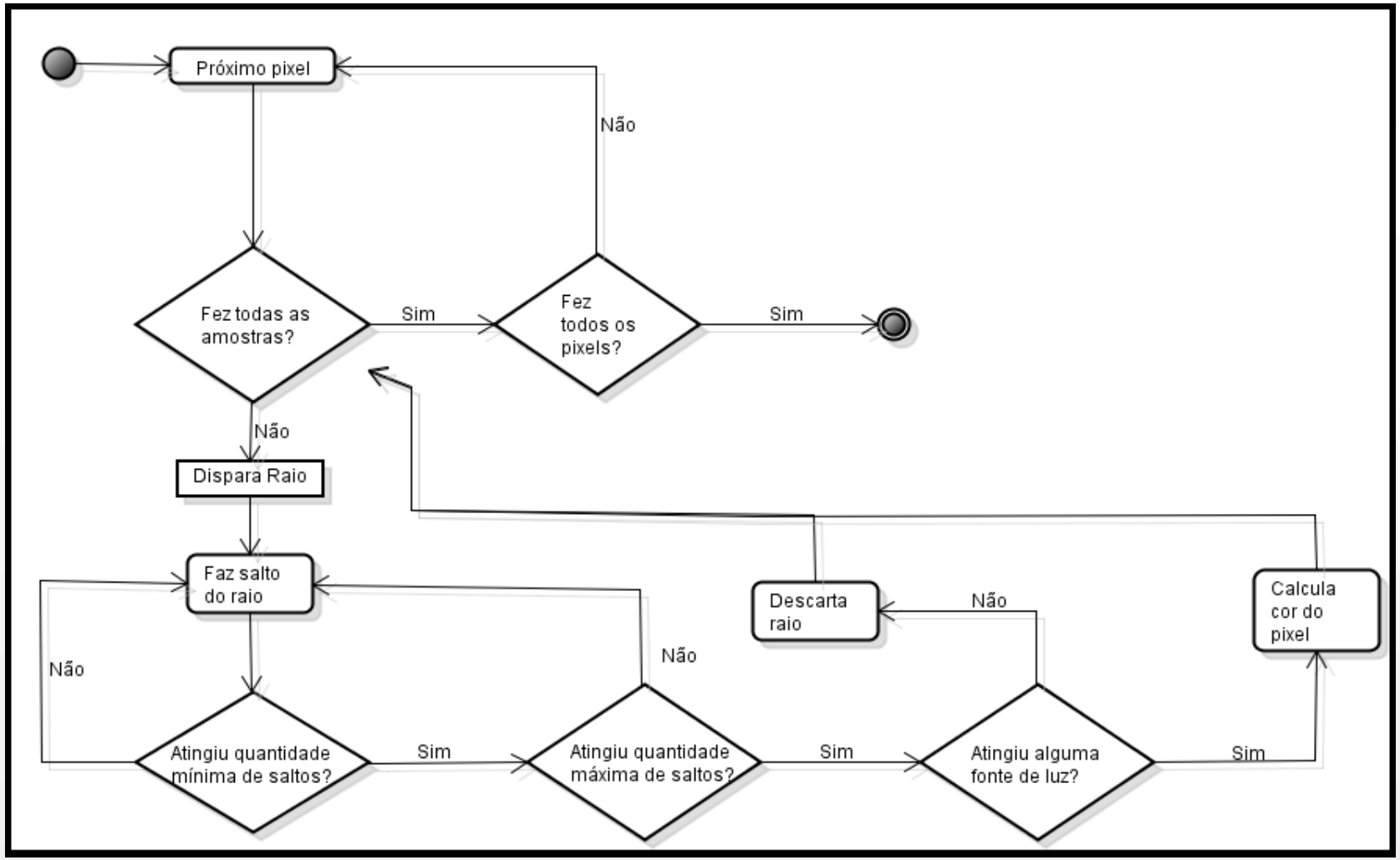
Especificação



Especificação



Especificação

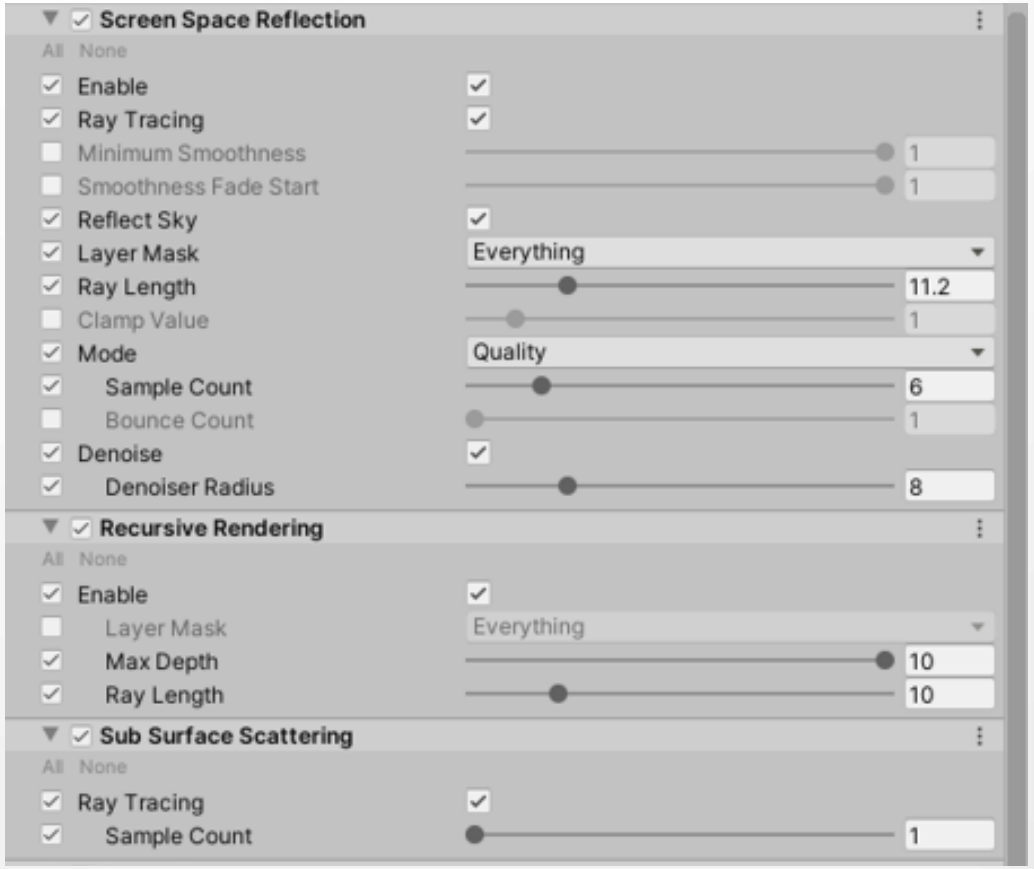


Implementação

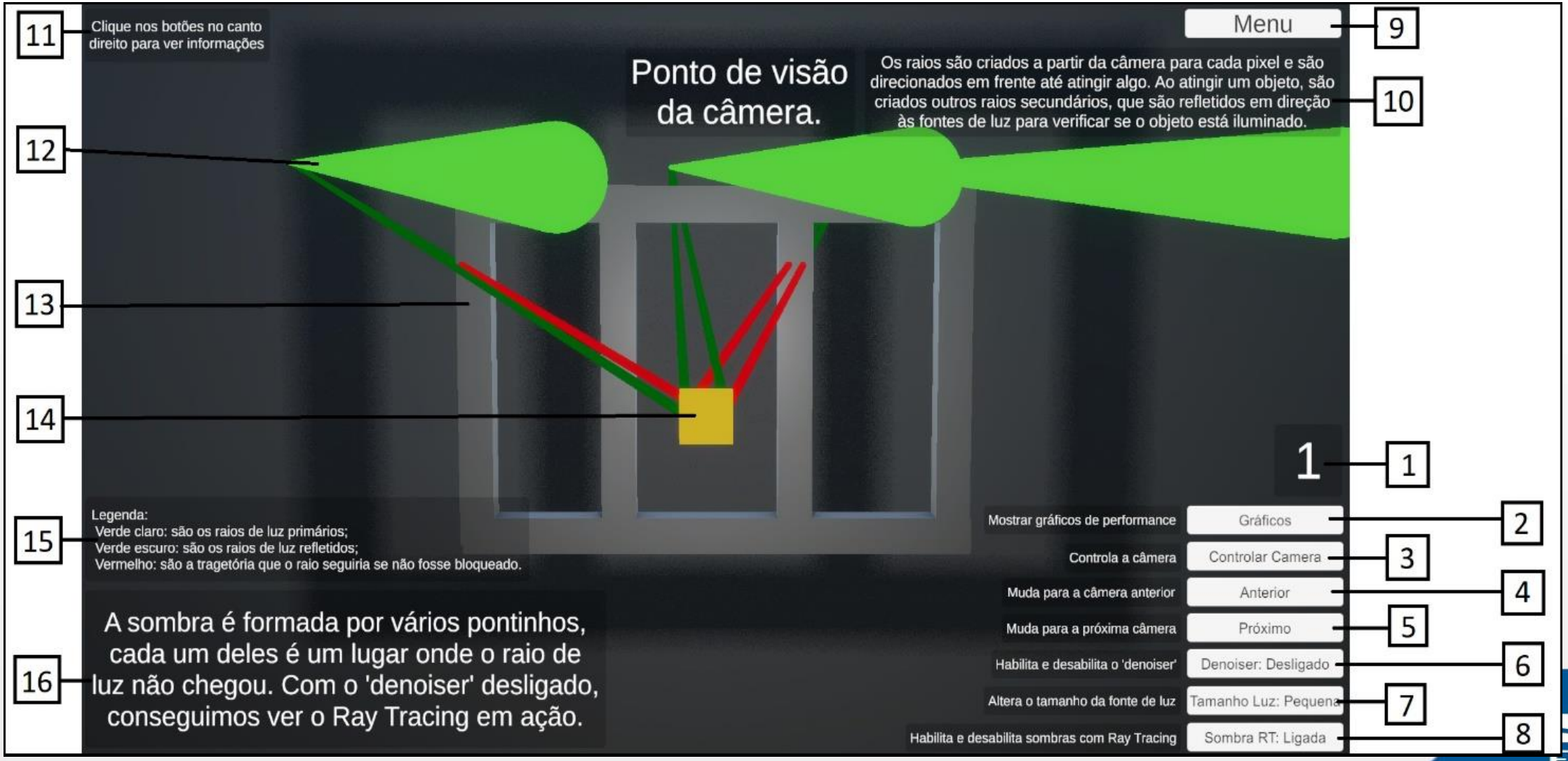
- Assets
  - MS Sports Car;
  - Snaps Prototype | Asian Residencies;
  - Snaps Prototype | Construction Site;
  - Snaps Art HD | Construction Site;
  - Asset Swap Tool.

Implementação

- Unity;
- Projeto HDPR;
- 4 cenas;
- 4 scripts;



Implementação



Resultados e Discussões

- Testes realizados
  - Sala `Sombras`;
  - Sala `Técnicas`;
  - Sala `PathTracing`;
- Correções efetuadas
  - Controle da câmera;
  - Gráficos de performance;
  - Refração;

Conclusões e Sugestões

- É possível utilizar *ray tracing* com Unity;
- Performance insatisfatória;
- Introduz o *ray tracing* à estudantes;

Conclusões e Sugestões

- Alterar e adicionar objetos nas cenas;
- Permitir uso em computadores com menor processamento;
- Explicações mais complexas;
- Adicionar as explicações aos objetos;

Demonstração