

# **GRADE: AMBIENTE GRÁFICO DE DESENVOLVIMENTO PARA ENSINO DE COMPUTAÇÃO GRÁFICA**

Aluna: Natália Sens Weise

Orientador: Dalton Solano dos Reis

# Roteiro

- Introdução;
- Objetivos;
- Fundamentação Teórica:
  - Abstração do espaço 3D;
  - Computação Gráfica;
  - Fundamentos na criação de tutoriais;
- Trabalhos Correlatos;
- Versão Anterior do Software;
- Requisitos;
- Especificação;
- Implementação;
- Análise dos Resultados;
- Conclusões e Sugestões.

# Introdução

- Continuação do projeto VisEdu-CG;
- Dificuldade em abstração 3D por parte dos alunos;
- Computação Gráfica.

# Objetivos

- Disponibilizar uma nova versão do projeto para ser utilizado na disciplina de Computação Gráfica na forma de material de apoio;
- Específicos:
  - validar se o ambiente desenvolvido consegue representar objetos gráficos 3D definidos em um Grafo de Cena;
  - validar se estes objetos gráficos 3D podem ser manipulados por Transformações Geométricas;
  - avaliar se a utilização de exercícios, usando o ambiente desenvolvido, pode auxiliar no entendimento dos assuntos abordados em aula.

# Fundamentação Teórica

- **ABSTRAÇÃO DO ESPAÇO 3D:**
  - Settimy e Barral (2020);
  - Dificuldade abstração 3D – aprendizado de geometria raso;
  - Geometria é fundamental para aprender CG;
- **COMPUTAÇÃO GRÁFICA:**
  - Manssour e Cohen (2006) – conceito de CG;
  - Outros assuntos devem ser abordados:
    - Grafo de cena;
    - Objetos gráficos;
    - Transformações geométricas homogêneas;
    - Câmera sintética;
    - Iluminação;
- **FUNDAMENTOS NA CRIAÇÃO DE TUTORIAIS:**
  - Cieślak (2021) – Primeiro contato do usuário com a aplicação;
  - Ganha familiaridade;
  - Lista de oito dicas para montar um tutorial perfeito;
  - Exemplos: ser breve, entre cinco e nove etapas.

# Trabalhos Correlatos

## SCRATCHIM: UMA ABORDAGEM PARA O ENSINO DO PENSAMENTO COMPUTACIONAL PARA CRIANÇAS DE FORMA REMOTA E DESPLUGADA

Referência	Rodrigues, Gomes e Carneiro (2022)
Objetivos	Material físico Pensamento computacional para crianças
Principais funcionalidades	Encaixar blocos para atingir o objetivo de cada tarefa proposta
Ferramentas de desenvolvimento	Blocos de materiais acessíveis e coloridos (exemplo: EVA)
Resultados e conclusões	Sucesso Alunos levaram muito tempo

# Trabalhos Correlatos

## JORNADA QUÍMICA GENIAL

Referência	Barros, Sousa e Viana (2022)
Objetivos	Ensinar sobre a tabela periódica
Principais funcionalidades	<b>Três trilhas:</b> Germânio (Ge) – memorizar elemento; Níquel (Ni) – relacionar com cotidiano; Alumínio (Al) – relacionar posição da tabela com características
Ferramentas de desenvolvimento	Web – Next.js e React.js
Resultados e conclusões	Alto desempenho – alcançaram aprendizado Sentimento de satisfação por parte dos alunos

# Trabalhos Correlatos

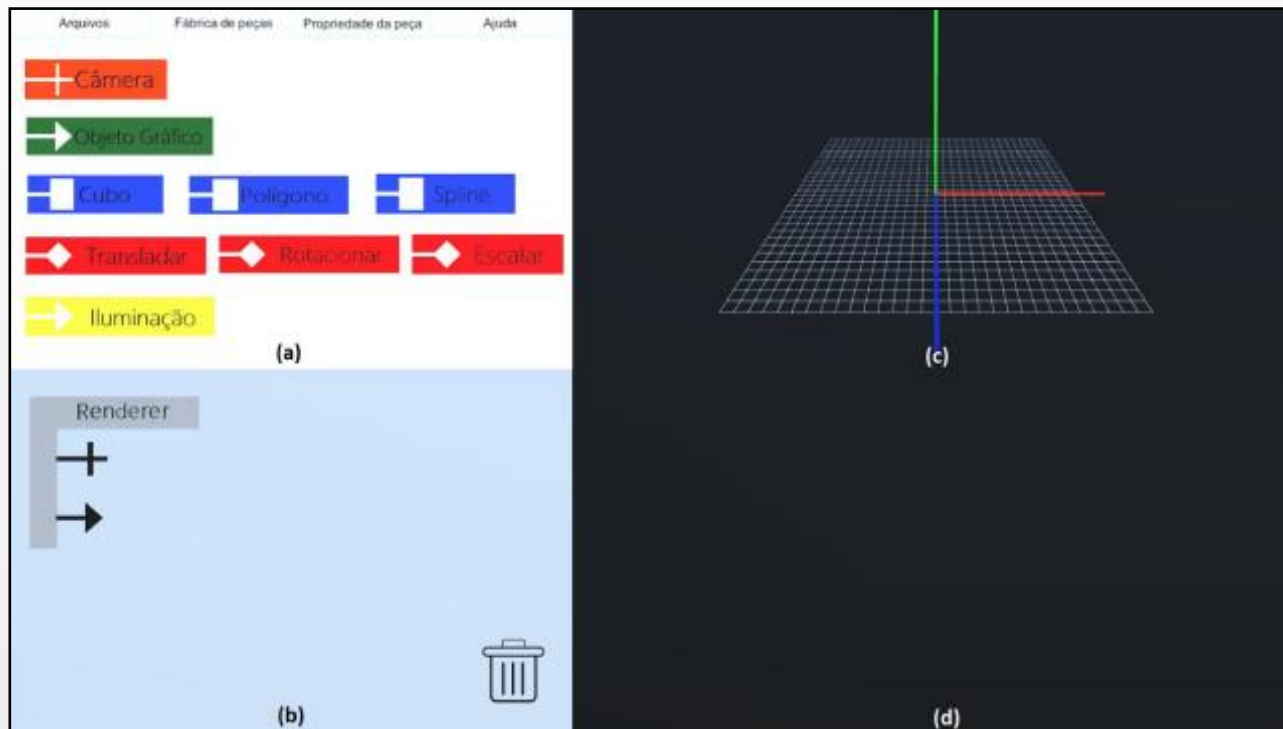
## ORBITANDO: UMA PLATAFORMA PARA ENSINO DE ASTRONOMIA DE OUTRO MUNDO

Referência	Siedler <i>et al.</i> (2022)
Objetivos	Ensinar astronomia de forma mais interessante Plataforma que ensinasse astronomia
Principais funcionalidades	<b>Primeiro jogo:</b> apresenta dois módulos, Professor e Aluno: Professor posta atividades, Aluno as faz e visualiza resultados e conteúdos <b>Segundo jogo:</b> dinâmica de fases (planetas). Tarefas para chegar no próximo planeta e ler as informações <b>Terceiro jogo:</b> com RA – visualiza planetas em 3D sobre o card, sem RA – vê o planeta em 2D
Ferramentas de desenvolvimento	<b>Primeiro jogo:</b> HyperText Markup Language 5 (HTML5), JavaScript, NodeJS e MongoDB <b>Segundo jogo:</b> Unity <b>Terceiro jogo:</b> Unity e Vuforia
Resultados e conclusões	Mais interesse e aprendizado Alunos se ajudaram no segundo jogo



# Versão Anterior do Software

- Tutorial de sete passos;
- Quatro telas: Fábrica de Peças, Renderer, Ambiente Gráfico e Visualizador;
- Nove tipos de objetos/componentes de cena;
- Faltou Spline, Polígono, propriedades da Câmera, tutorial limitado.



Fonte: Buttenberg (2020)

# Requisitos

## Requisitos Funcionais da Ferramenta

RF01	permitir que o usuário possa seguir um tutorial para auxiliar o entendimento da ferramenta
RF02	permitir que o usuário possa arrastar as peças e editar suas informações conforme for desejado
RF03	permitir que o usuário possa mexer no tema da aplicação (modo claro ou escuro) conforme melhor lhe agradar
RF04	permitir que o usuário possa fazer uso da câmera com todas as suas propriedades
RF05	permitir que o usuário possa realizar exercícios pré-definidos, a fim de treinar seus conhecimentos adquiridos
RF06	permitir que o usuário saiba se acertou o exercício de treinamento ou não e, caso tenha acertado, qual foi a porcentagem desse acerto
RF07	permitir que o usuário possa importar e exportar a cena criada com as peças em formato JSON

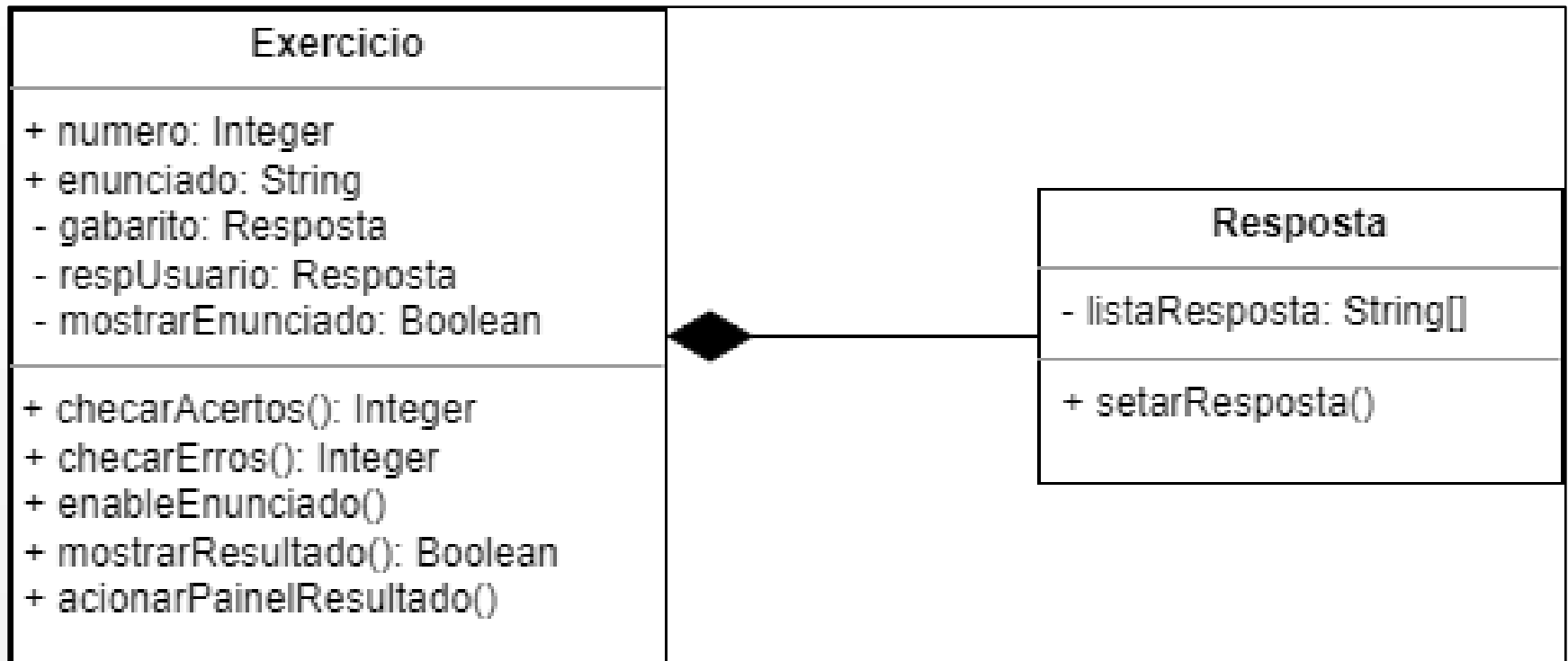
# Requisitos

## Requisitos Não Funcionais da Ferramenta

RNF01	utilizar o motor de jogos Unity em conjunto com a IDE Visual Studio
RNF02	utilizar a linguagem de programação C# para implementação
RNF03	ser desenvolvido para plataforma web

# Especificação

Diagrama de classes: Exercício e Resposta



Redundância: lista pronta

# Especificação

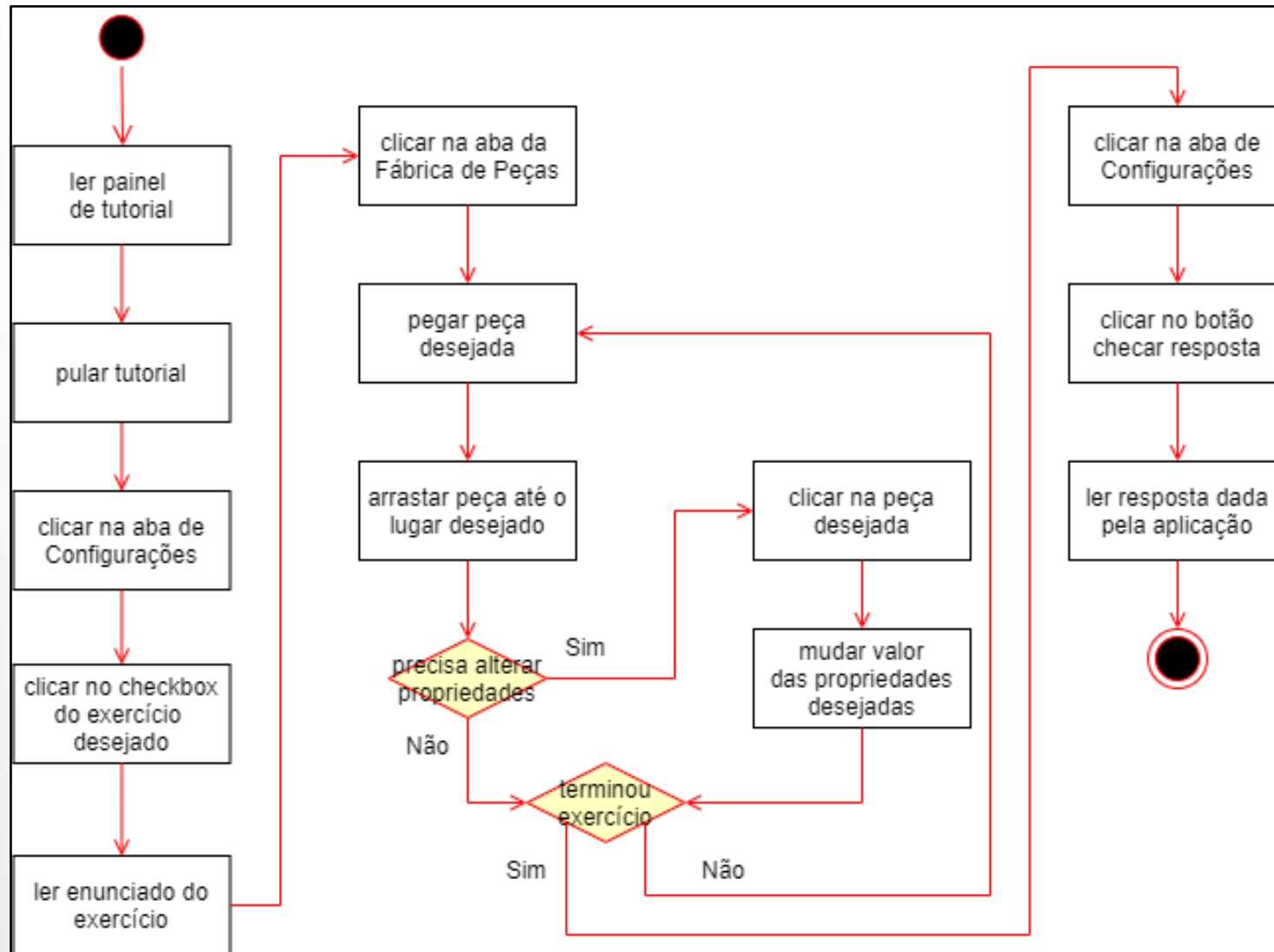
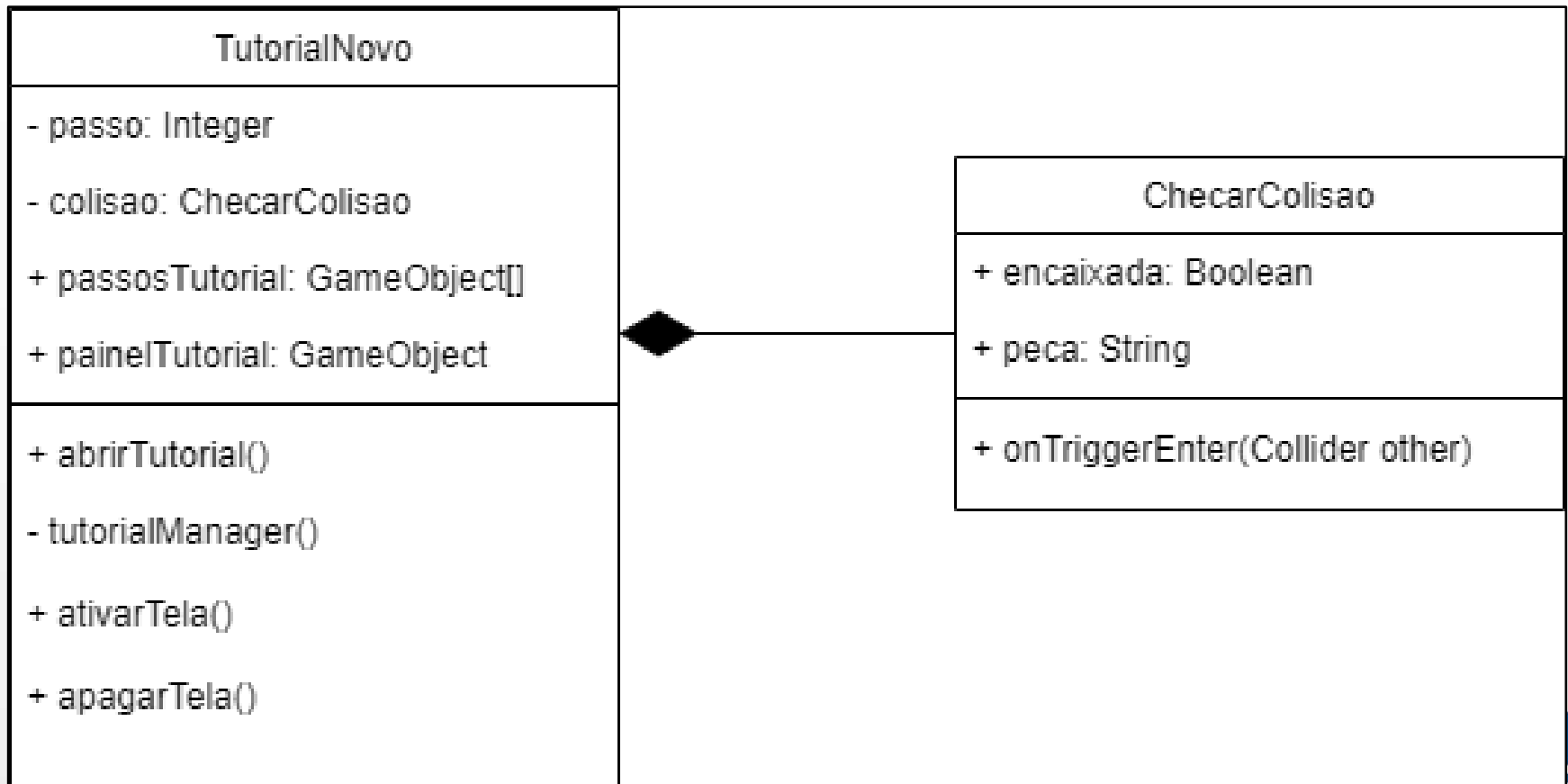


Diagrama de atividades:  
realizar exercício

# Especificação

Diagrama de classes: TutorialNovo e ChecarColisao



# Especificação

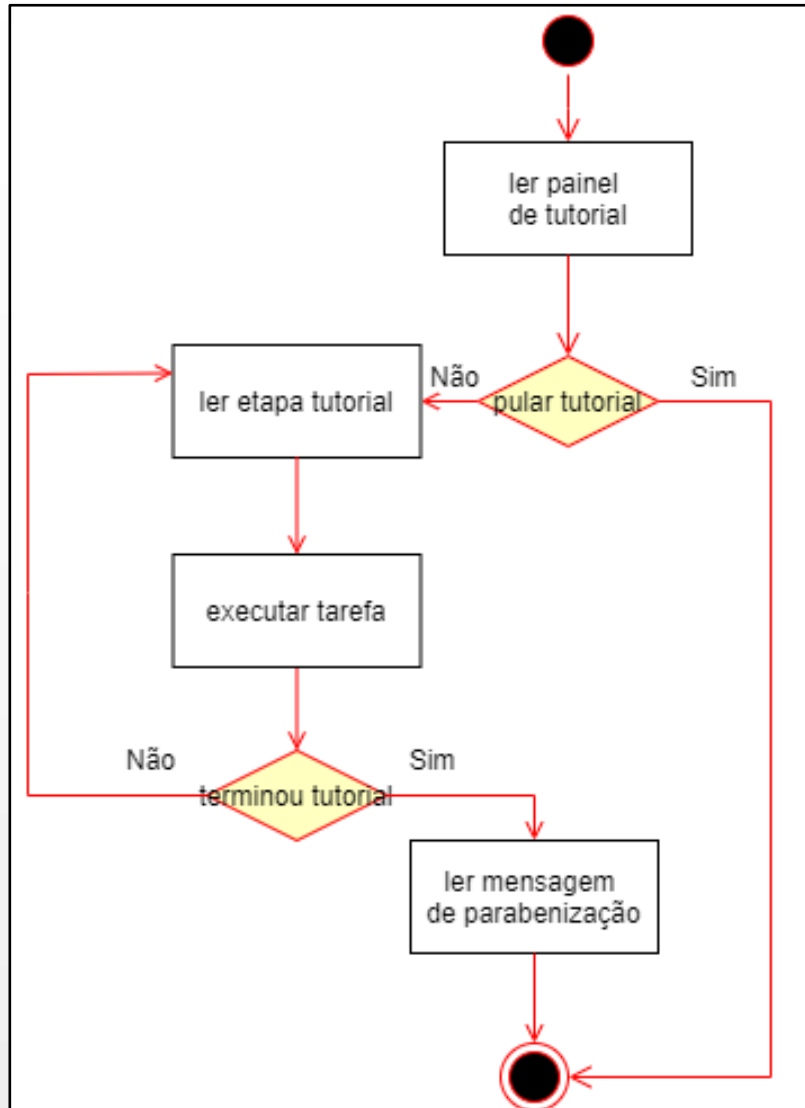


Diagrama de atividades:  
realizar tutorial

# Especificação

Diagrama de classes: Arquivo

Arquivo
<ul style="list-style-type: none"><li>+ cena: JSONObject</li><li>+ importando: Boolean</li><li>- numObjetoAtual: Integer</li><li>- objetoAtual: MeuObjetoGrafico</li><li>- objetoPai: MeuObjetoGrafico</li><li>- nomeObjetoAtual: String</li><li>- nomeObjetoPai: String</li></ul>
<ul style="list-style-type: none"><li>+ ordenarCena(List&lt;GameObject&gt; lista): List&lt;GameObject&gt;</li><li>+ exportar()</li><li>+ importar()</li></ul>



# Especificação

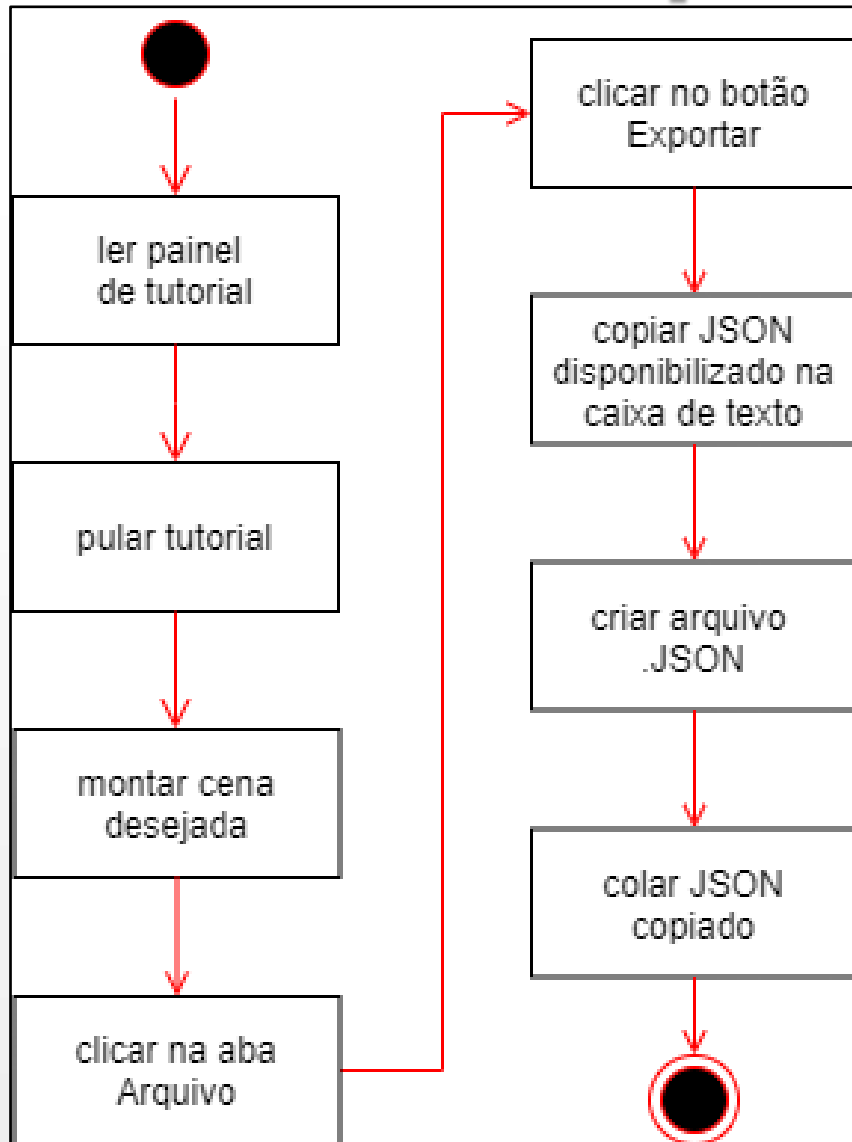


Diagrama de atividades:  
exportar cena em formato JSON

# Especificação

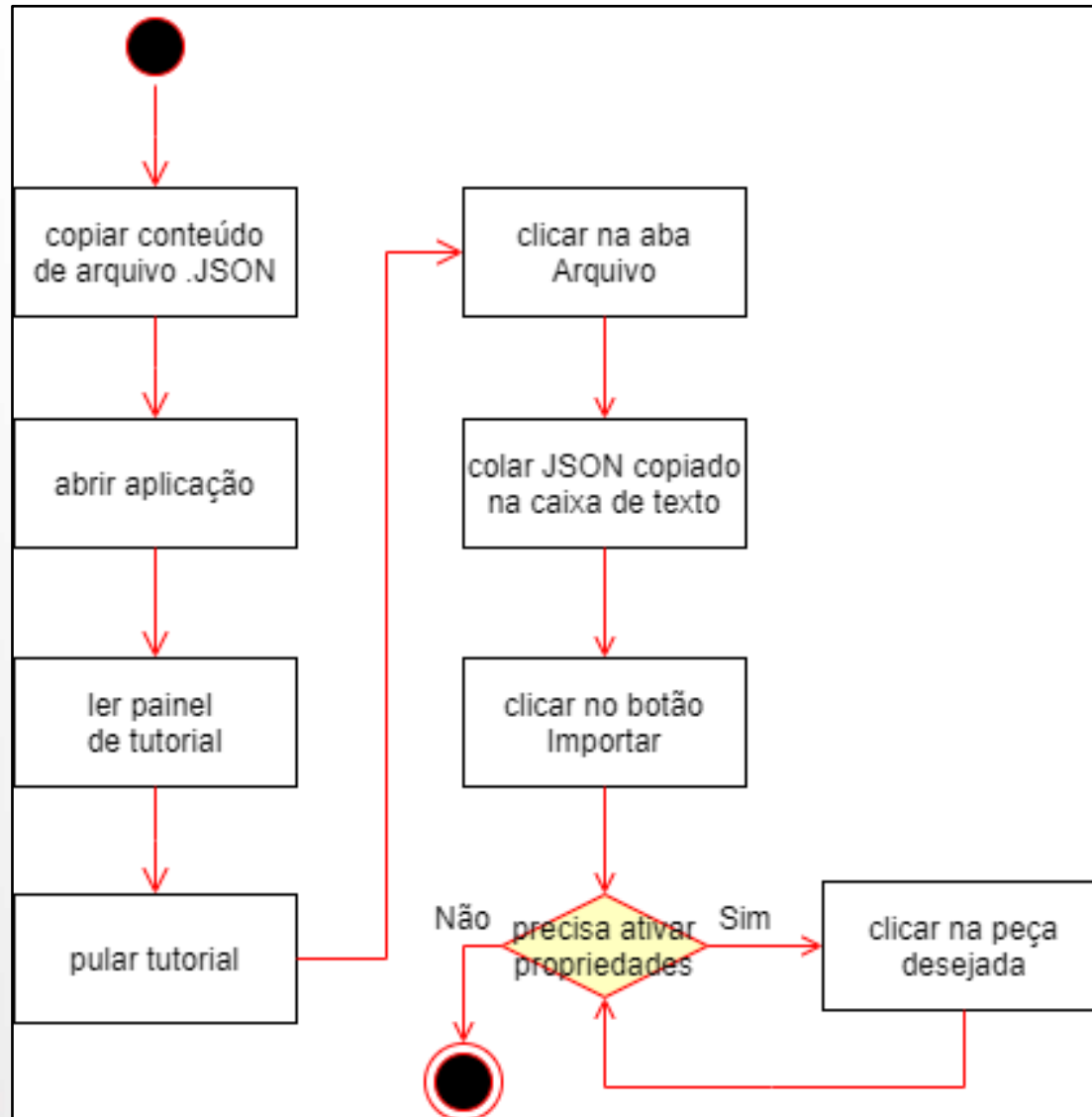


Diagrama de atividades:  
importar cena em JSON

# Especificação

Diagrama de classes: TrocaDeTema

TrocaDeTema
- control: Boolean + claro: Material + escuro: Material
+ trocandoTema() - trocandoFundo()

# Especificação

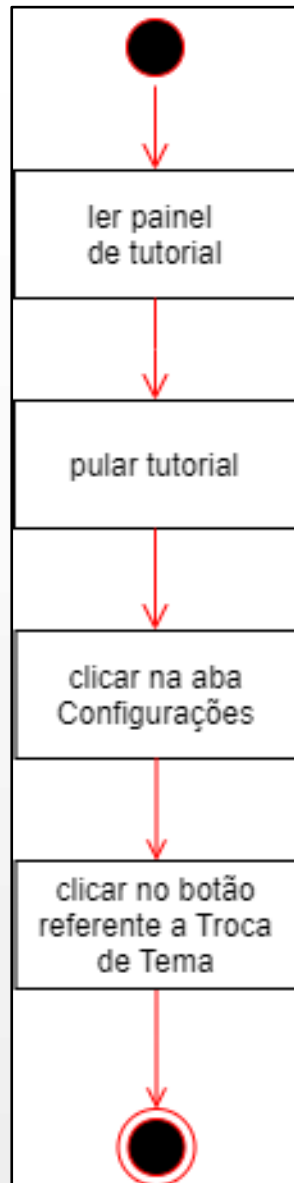


Diagrama de atividades:  
trocar tema

# Implementação

- Softwares:
  - Unity 2022.1.13f;
  - IDE Visual Studio 2019 16.11.34;
  - Draw.io 1.0 e Paint 3D;
- Principais funcionalidades:
  - Tutorial;
  - Troca de tema;
  - Exercícios;
  - Importação/Exportação;
  - Hierarquia de objetos.

# Implementação

- **Tutorial**

- Seguiu as dicas apontadas por Cieślak (2021);
- Telas foram criadas a partir de imagens já existentes dentro do projeto com o software Paint 3D;
- Dois tutoriais de nove etapas cada: oito são ações e uma é parabenização;
  - Um ensina como montar uma cena básica e o outro ensina como montar hierarquia pai-filho.

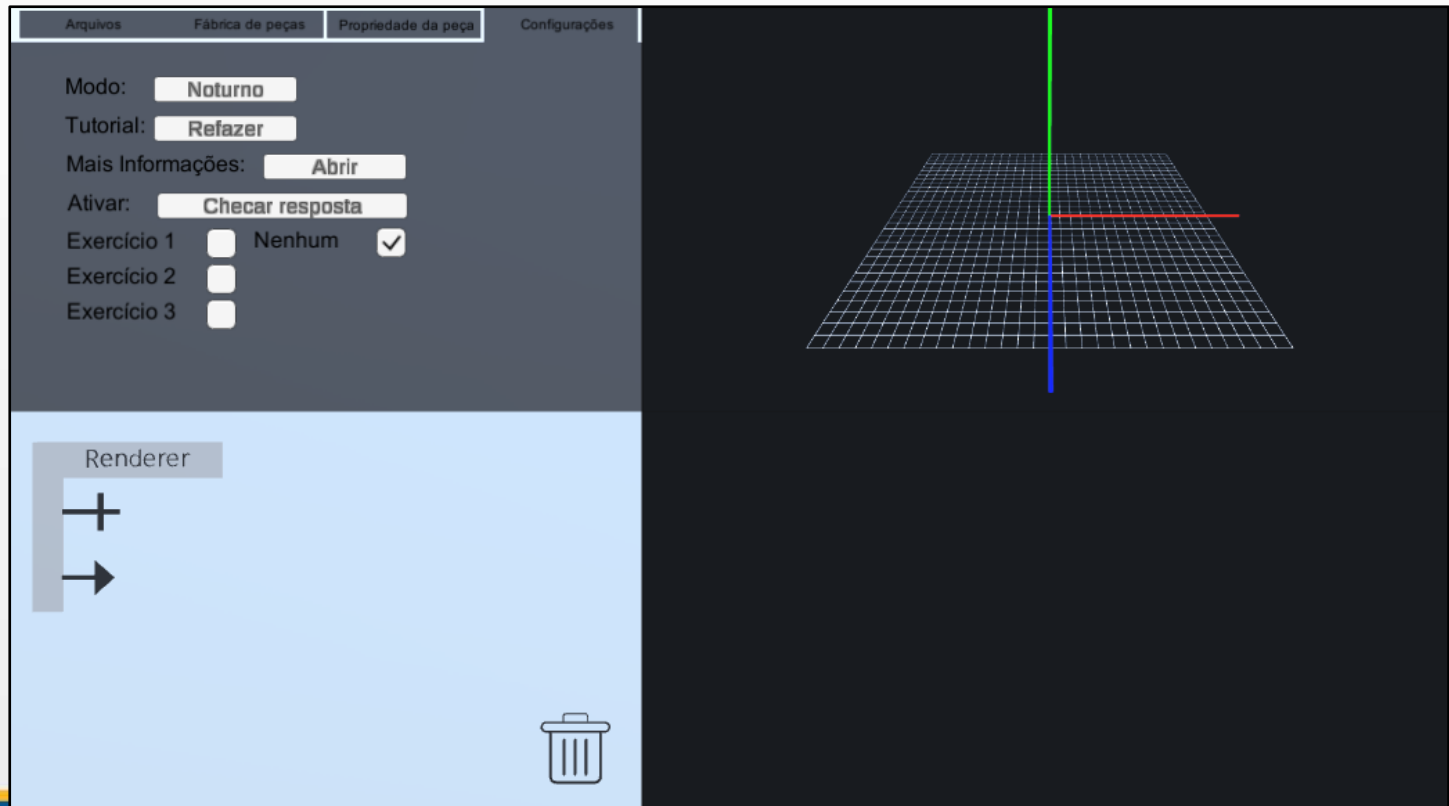
### Função principal da classe TutorialNovo

```
public void abrirTutorial()  
{  
    switch (passo)  
    {  
        case 0 when render.activeSelf && !grade.isOn:  
        case 1 when colisao.encaixada && colisao.peca == "Camera":  
        case 2 when colisao.encaixada && colisao.peca == "Objeto":  
        case 3 when colisao.encaixada && colisao.peca == "Iluminacao":  
        case 4 when colisao.encaixada && colisao.peca == "Cubo":  
        case 5 when colisao.encaixada && colisao.peca == "Escala":  
        case 6 when escala.activeSelf && escalarTexto.text == "3":  
        case 7 when Global.listaEncaixes.Count == 0:  
            tutorialManager(passosTutorial);  
            break;  
        case 8:  
            StartCoroutine(apagarTela(passosTutorial[passo]));  
            StartCoroutine(apagarTela(painelTutorial));  
            passo = 0; grade.isOn = true; break;  
    }  
}
```

# Implementação

- **Troca de Tema:**
  - Troca apenas do background;
  - Problemas na troca da cor das letras.

GRADE no modo noturno





# Implementação

- **Exercícios:**

- Três atividades;

- Mensagens:

- Verifique a ordem das peças em cena ou se alguma está faltando;
    - Verifique se as propriedades mencionadas no enunciado foram alteradas corretamente.

# Implementação

- **Importação/Exportação de cena:**
  - **Exportar:**
    - Aba Arquivo - Exportar;
    - SimpleJSON;
    - WebGL Copy And Write;
  - **Importar:**
    - Aba Arquivo - Importar;
    - Clicar nas peças para ativar as propriedades visualmente.

## Exemplo de cena exportada em JSON

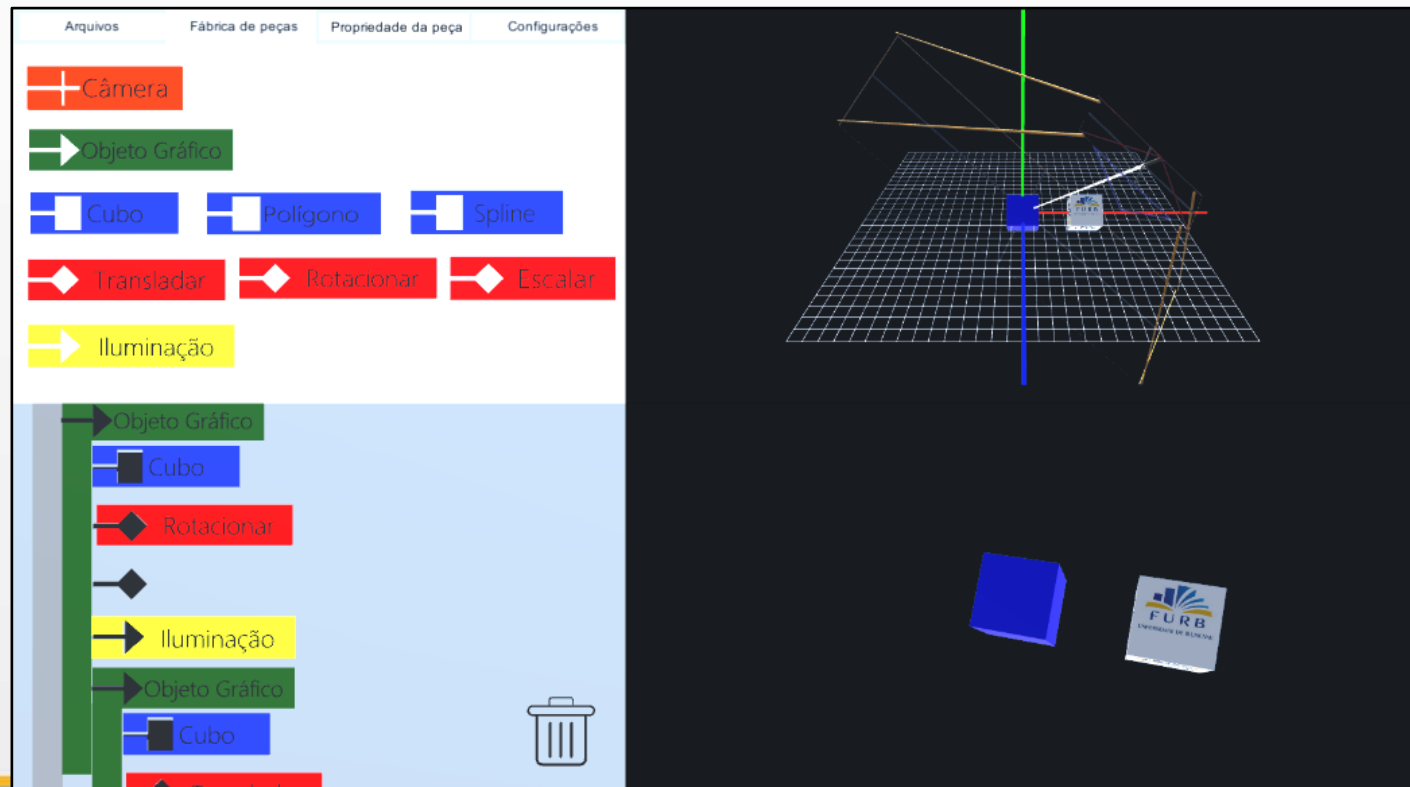
```
{
  "CameraP": {
    "nome": "Câmera",
    "posicao": ["100", "300", "300"],
    "lookAt": ["0", "0", "0"],
    "fov": "45",
    "near": "100",
    "far": "600",
    "posPeca": [696.351135253906, 624.93212890625, -870.424987792969]
  },
  "ObjetoGraficoP": {
    "nome": "ObjetoGraficoP",
    "ativo": true,
    "children": [
      {
        "Cubo": {
          "nome": "Cubo",
          "tamanho": ["2", "2", "2"],
          "posicao": ["0", "0", "0"],
          "cor": "RGBA(1.000, 0.000, 0.000, 1.000)",
          "textura": "FURB",
          "ativo": true,
          "posPeca": [697.420593261719, 618.979125976563, -870.424987792969]
        }
      },
      {
        "Iluminacao": {
          "nome": "Iluminacao",
          "tipoLuz": "Ambiente",
          "posicao": ["100", "300", "0"],
          "cor": "RGBA(1.000, 1.000, 1.000, 1.000)",
          "ativo": true,
          "posPeca": [698.855163574219, 613.408264160156, -870.403076171875]
        }
      }
    ]
  },
  "posPeca": [697.301147460938, 621.630798339844, -870.403076171875]
}
```

# Implementação

- **Hierarquia de objetos**

- Permite que o usuário possa fazer com que um objeto gráfico tenha um filho, fazendo com que o objeto filho herde as propriedades da matriz de transformação do pai.

Exemplo de cena com hierarquia pai e filho



# Análise dos Resultados

- Testes de funcionalidade saíram conforme esperado;
- Analisando as respostas obtidas pelo questionário, grande maioria conseguiu usar a ferramenta e acredita que ela seja muito útil no aprendizado de CG;
- Diversos pontos positivos e negativos foram apontados pelos alunos, além de sugestões.

Pontos positivos da aplicação apontados pelos alunos de CG

Palavras-chave/termos usados para descrever a aplicação	Qtd. ocorrências
Fácil/simples/prática de usar	10
Facilitou no entendimento/aprendizado dos assuntos abordados em aula	10
Intuitiva/fácil de entender	9
Visualização 3d/ apoio visual ajuda a entender/visualização da cena	8
Scratch/sketch/ <i>drag-drop</i> das peças/interação com a cena	6
Educativa/didática	4
Exercícios bons para fixação de conteúdo/exercícios interessantes e correspondem ao assunto de cg	2
Web	2
Tutorial ajuda muito	2
Divertido	2
Rápida	2
Interessante	2
Funcional	1
Leve	1
Legal	1
Permite autoavaliação e revisão de conteúdo	1

Palavras-chave/termos usados para descrever a aplicação	Qtd. ocorrências
Texto fica invisível no campo de texto das propriedades após o uso da tecla <i>backspace</i>	19
Visual pouco elaborado/baixa resolução/problemas na responsividade	4
Problemas na acentuação de palavras	3
<i>Bug</i> na porcentagem de acerto do exercício quando clica mais de uma vez no botão	3
<i>Bugs</i>	3
Quando há muitos objetos em cena, demora para fazer a rolagem de tela	2
Aplicação um pouco travada	2
Visualização dos ângulos dos objetos está presa	2
Funcionamento da câmera é estranho/complicado de entender	2
Textos pequenos	1
Poderia utilizar o localStorage para armazenar informações de sessão do usuário.	1
Não é mostrado ao usuário qual o intervalo de números que podem ser inseridos nas propriedades	1
Deveria ter uma explicação mais detalhada sobre as outras funcionalidades da aba Configurações	1
Modo escuro restrito a apenas uma das quatro janelas	1
Na hierarquia pai-filho, quando o filho já herdou algo do pai e é clicado no filho para editar, ele perde na visualização o que herdou do pai e é preciso clicar novamente no pai para a propriedade reaparecer na visualização	1
Tutorial não permite pular etapas	1

Pontos negativos da aplicação apontados pelos alunos de CG – parte 2

Palavras-chave/termos usados para descrever a aplicação	Qtd. ocorrências
Métricas de avaliação dos exercícios são confusas	1
O <i>check</i> sai do <i>checkbox</i> do exercício selecionado quando o botão de checar é clicado	1
Os enunciados dos exercícios podem ser confusos para alguns, deveria ter dicas de como resolver	1
Visualização não renderiza	1
Confusa	1
Talvez mais exercícios em sala de aula	1
Importação não aplica propriedades corretamente, mesmo estando certo no JSON	1
<i>Bug</i> na exclusão de objetos	1
Tela de visualização dos objetos é pequena	1
A aplicação não permite o acesso ao gabarito	1



Sugestões para a aplicação apontadas pelos alunos de CG

<b>Termos e elogios usados/dados para descrever a aplicação</b>	<b>Qtd. ocorrências</b>
Ficará ótimo se arrumar <i>bugs</i> anteriormente apontados	4
Melhorar modo noturno	3
Parabéns pelo trabalho!	2
Corrigir <i>bug</i> dos <i>inputs</i> das propriedades	2
Fazer uso de setas no tutorial para indicar onde encaixar as peças, onde pegá-las, fazendo com que fique mais fácil e interativo	2
Remover Spline e Polígono da Fábrica de Peças, visto que não funcionam	1
Permitir acesso ao gabarito dentro da aplicação	1
Poder mover objetos utilizando o mouse como na Unity	1
Arrumar bugs	1
Aumentar tela de visualização	1
Arrumar acentuação	1

Comparação entre os correlatos e aplicação desenvolvida

<b>Características\Trabalhos</b>	<b>Scratchim (Rodrigues; Gomes; Carneiro, 2022)</b>	<b>GeNiAI (Barros; Sousa; Viana, 2022)</b>	<b>OrbitAndo (Siedler et al., 2022)</b>	<b>GRADE</b>
Existe interação por meio de peças	X			X
É um software educacional		X	X	X
Apresenta exercícios para validação	X	X	X	X
Apresenta tutorial explicando o seu uso	X			X
Apresenta conteúdos teóricos	X		X	X
Possui acesso off-line	X		X	Apenas após carregamento completo da página
Foi desenvolvido em Unity			Dois dos três jogos	X
Disponibilidade	Físico	Web	Multiplataforma	Web

# Conclusões e Sugestões

- Satisfação por parte dos alunos;
- Evolução ao longo dos exercícios;
- Softwares escolhidos foram adequados e empenharam bem seu papel;
- Polígono e Spline não foram implementados e Câmera deixou a desejar.

# Conclusões e Sugestões

- Possíveis extensões:
  - construir as peças `Spline` e `Polígono`;
  - fazer com que as propriedades `LookAt`, `Near` e `Far` da `Câmera` funcionem de forma satisfatória;
  - corrigir validação dos campos de propriedades para que o texto não perca a visibilidade;
  - corrigir cálculo da porcentagem de acertos dos exercícios;
  - arrumar as propriedades das peças, para que não seja necessário clicar nelas quando forem importadas nem herdadas;
  - fazer um esquema de temporização ao longo do exercício e, quando o aluno estiver levando muito tempo para fazer, disponibilizar acesso a dicas de como resolver;
  - trazer o modo noturno para a tela de `Renderer`;
  - trazer setas indicando o caminho durante o tutorial.