

| CURSO DE CIÊNCIA DA COMPUTAÇÃO – TCC |               |                      |
|--------------------------------------|---------------|----------------------|
| ( ) PRÉ-PROJETO                      | ( X ) PROJETO | ANO/SEMESTRE: 2023/2 |

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

Natália Sens Weise

Prof. Dalton Solano dos Reis – Orientador

### 1 INTRODUÇÃO

Conforme dito por Manssour e Cohen (2006, p. 1), a Computação Gráfica (CG) “é uma área da Ciência da Computação que se dedica ao estudo e desenvolvimento de técnicas e algoritmos para a geração (síntese) de imagens através do computador.”. Para realizar as devidas transformações nas imagens, é preciso fazer uso da matriz de transformação, que é responsável por proporcionar escala, rotação e translação aos objetos gráficos da cena. Também é necessário o conhecimento de outros assuntos dentro dessa temática, sendo eles: grafo de cena, objetos gráficos, transformações geométricas homogêneas (matriz de transformação), câmera sintética e iluminação. Contudo, ainda é preciso que se tenha uma boa fundamentação teórica em geometria, visto que os conceitos de CG se baseiam nessa área da matemática (Azevedo; Conci; Vasconcelos, 2022).

Como Settimy e Bairral (2020) observaram, os alunos possuem dificuldade na abstração do espaço 3D pelo fato do ensino básico não abordar a Geometria de forma mais clara e aprofundada. Segundo Settimy e Bairral (2020, p. 3), “a Geometria é um campo fértil para perceber e entender as formas geométricas presentes em nosso cotidiano, sendo possível desenvolver habilidades importantes como a experimentação, representação, descrição e argumentação [...]”, sendo fundamental para o entendimento de CG. Dentre as diversas ferramentas de apoio existentes, uma que se destaca no âmbito da Geometria é o Geogebra 3D, que permite criar objetos 3D e manipular os valores de suas propriedades, o que contribui muito para o aprendizado da matéria como visto por Fassarella e Rocha (2018).

Outro material de apoio que se sobressai é o VisEdu-CG, construído por Buttenberg (2020) com o objetivo de auxiliar os alunos da Fundação Universidade Regional de Blumenau (FURB) no entendimento dos assuntos abordados na disciplina de CG do curso de Ciência da Computação. O projeto apresenta uma tela dividida em quatro seções: Fábrica de Peças, na qual o usuário pega os blocos para programar; Renderer, em que o usuário deposita as peças que coletou na Fábrica de Peças; Ambiente Gráfico, em que é possível visualizar os eixos, grade e objetos colocados em cena; e Visualizador, que mostra o resultado da execução do que foi projetado pelo usuário. Todavia, nem todos os objetivos propostos por Buttenberg (2020) foram concluídos. Algumas funcionalidades propostas, como os objetos Polígono e Spline e algumas propriedades da câmera não foram implementadas. Além disso, o tutorial desenvolvido ficou limitado a poucas funções básicas.

Sendo assim, esse projeto visa auxiliar os alunos de CG a entenderem os assuntos abordados em aula continuando com o desenvolvimento do antigo VisEdu-CG (Buttenberg, 2020), implementando as funcionalidades faltantes e trazendo novas, como a interface com mudança de tema (claro e escuro) para o usuário escolher o que mais lhe agrada à vista, além de exercícios para fixação do conteúdo e um tutorial mais completo.

#### 1.1 OBJETIVOS

O objetivo principal deste trabalho é disponibilizar uma nova versão do VisEdu-CG, agora chamado de ambiente GRÁfico de Desenvolvimento para Ensino de computação gráfica (GRADE), para ser utilizado na disciplina de Computação Gráfica na forma de material de apoio.

Os objetivos específicos são:

- a) validar se o ambiente desenvolvido consegue representar objetos gráficos 3D definidos em um Grafo de Cena;
- b) validar se estes objetos gráficos 3D podem ser manipulados por Transformações Geométricas;
- c) avaliar se a utilização de exercícios, usando o ambiente desenvolvido, pode auxiliar no entendimento dos assuntos abordados em aula.

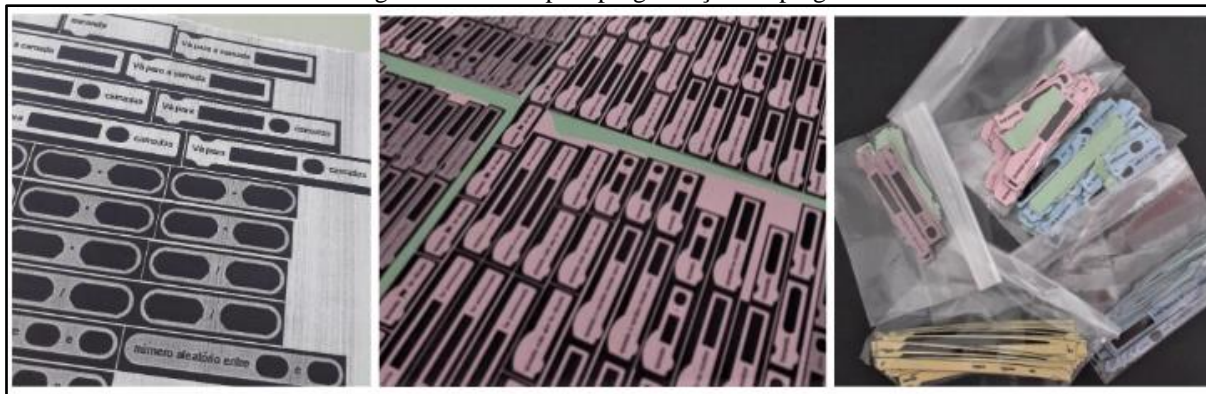
### 2 TRABALHOS CORRELATOS

Essa seção expõe três trabalhos selecionados com características em comum ao que se pretende desenvolver. A subseção 2.1 traz um jogo desplugado para ensinar pensamento computacional às crianças, proposto por Rodrigues, Gomes e Carneiro (2022). A subseção 2.2 aborda o jogo GeNiAl desenvolvido por Barros, Sousa e Viana (2022), que busca ensinar a tabela periódica para estudantes do ensino superior. A subseção 2.3 apresenta uma plataforma com jogos que ensinam astronomia projetada por Siedler *et al.* (2022).

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

Perante o cenário pandêmico vivido mundialmente e a falta de acesso à internet e equipamentos eletrônicos sofridos por algumas escolas, Rodrigues, Gomes e Carneiro (2022) sugeriram a utilização da computação desplugada para promover o aprendizado do Pensamento Computacional (PC) de forma remota. Para isso, o já conhecido Scratch, recurso tecnológico para o ensino de PC, foi trazido para o meio físico: a equipe do projeto desenvolveu os blocos para programação com materiais acessíveis e coloridos (Figura 1), este último para despertar o interesse das crianças. Além disso, foram gravadas aulas para auxiliar os alunos e disponibilizadas apostilas com maiores explicações sobre o kit (Rodrigues; Gomes; Carneiro, 2022).

Figura 1 - Blocos para programação desplugada



Fonte: Rodrigues, Gomes e Carneiro (2022).

Para realizar os exercícios e praticar o conhecimento, os estudantes tinham que encaixar os blocos conforme necessário para atingir o objetivo da tarefa. Com base nos resultados obtidos, notou-se que os alunos conseguiram concluir as atividades e adquiriram o conhecimento desejado. Contudo, os alunos levaram mais tempo por não terem apoio presencial dos professores para tirar dúvidas (Rodrigues; Gomes; Carneiro, 2022).

## 2.2 JORNADA QUÍMICA GENIAL

Barros, Sousa e Viana (2022) propuseram uma aplicação com o foco em ajudar estudantes de ensino superior, que estejam na área das ciências ou que apenas tenham interesse no assunto, a aprender sobre a tabela periódica. A necessidade foi observada ao notar como o assunto era abordado de forma cansativa e desgastante nas aulas, desmotivando o aluno no momento do aprendizado. Com isso, buscou-se trazer uma solução divertida e estimulante para os entusiastas de química.

O jogo foi desenvolvido para web em Next.js e React.js. Nele, existe um quiz e mais três trilhas para treinar diferentes conhecimentos da área, sendo que cada uma delas está ligada a um objetivo proposto (Figura 2): Germânio (Ge), com exercícios de agilidade para memorizar nome, símbolo e número atômico do elemento; Níquel (Ni), um minijogo da memória com o objetivo de relacionar elementos químicos com artigos do cotidiano; e Alumínio (Al), com atividades de lógica que buscam relacionar a posição do elemento na tabela com suas características (Barros; Sousa; Viana, 2022).

Figura 2 - Telas do jogo GeNiAl



Fonte: Barros, Sousa e Viana (2022).

Para verificar a eficácia do jogo em ensinar sobre a tabela periódica, foram disponibilizados formulários de pesquisa aos voluntários do projeto. Com base nas respostas obtidas, notou-se que a aplicação contribuiu com o fortalecimento e aprimoramento dos saberes dos alunos, visto que obtiveram alto desempenho nas atividades do

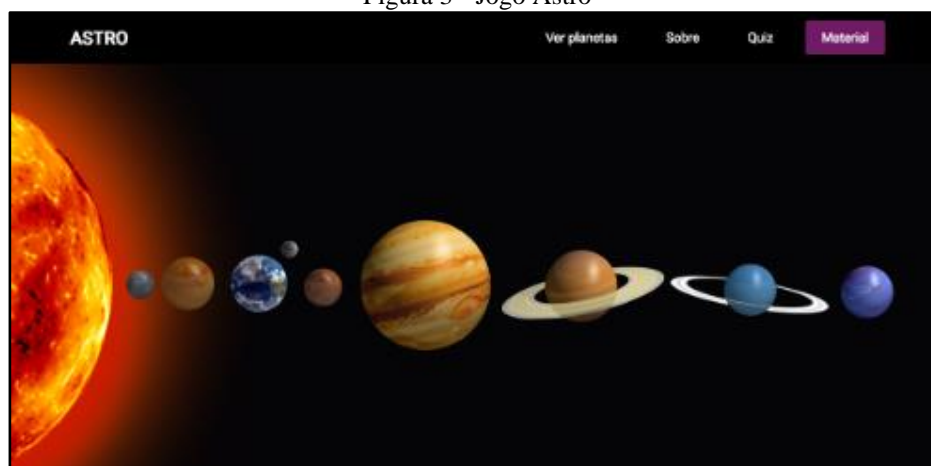
jogo. Os pesquisados também informaram que obtiveram sentimento de satisfação ao concluir as tarefas pré-estabelecidas com êxito (Barros; Sousa; Viana, 2022).

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

Para obter o aprimoramento das técnicas de ensino sobre astronomia em sala de aula, Siedler *et al.* (2022) criaram uma plataforma com jogos para auxiliar os professores a ensinarem o tema de forma mais interessante aos alunos, promovendo engajamento. Ao entrevistar os professores, foi requisitado que a plataforma OrbitAndo fosse multiplataforma, funcionasse em diversos aparelhos e de forma off-line, além de imergir o aluno em temas voltados ao Sistema Solar. A partir disso, também foram criados três jogos para compor o ambiente proposto: Astro, Jogo: Uma Volta pelo Sistema Solar e Orbit A.R. (Siedler *et al.*, 2022).

O primeiro jogo (Figura 3) foi desenvolvido utilizando HyperText Markup Language 5 (HTML5), JavaScript, NodeJS e MongoDB e apresenta dois módulos: Professor e Aluno. Em Professor, o docente pode inserir mais informações sobre o tema, aplicar questionários, coletar dados de desempenho dos discentes, entre outras funcionalidades. No modo Aluno, o estudante pode visualizar as informações postadas clicando em cada um dos planetas alinhados na tela, além de realizar questionários e salvar em arquivo no formato PDF tanto o conteúdo sobre planetas quanto as questões com suas respostas registradas (Siedler *et al.*, 2022).

Figura 3 - Jogo Astro



Fonte: Siedler *et al.* (2022).

O segundo jogo (Figura 4) foi desenvolvido no motor de jogos Unity e possui a dinâmica de fases. Cada fase é um planeta e, para ganhar o jogo, o usuário deve viajar de planeta em planeta, começando pelo Sol e terminando o trajeto em Netuno. Para alcançar ao próximo astro, o aluno deve completar tarefas e ao chegar no destino pode acessar informações sobre aquele planeta (Siedler *et al.*, 2022).

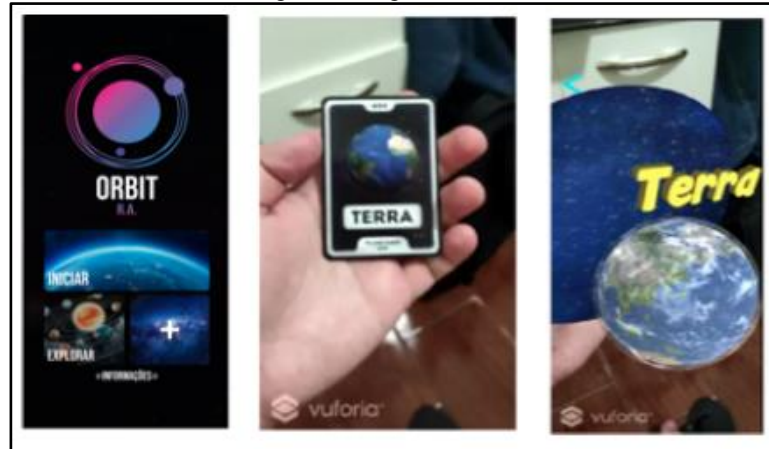
Figura 4 - Jogo: Uma Volta pelo Sistema Solar



Fonte: Siedler *et al.* (2022).

O terceiro jogo (Figura 5) faz uso da Unity em conjunto com o Vuforia, para trazer uma experiência mais imersiva. Nele, o usuário lê com a câmera do smartphone com sistema Android cartas que funcionam como marcadores. Ao ler a imagem, o aplicativo projeta o respectivo astro em 3D na tela. Caso o usuário não possua os cartões, pode visualizar as imagens em 2D (sem a experiência de Realidade Aumentada). Ao testar a plataforma com alunos do quinto ano, notou-se maior interesse e aprendizado do conteúdo. Além disso, as crianças fizeram uso de trabalho em equipe no segundo jogo, como estratégia para passar de fase (Siedler *et al.*, 2022).

Figura 5 - Jogo Orbit A.R.



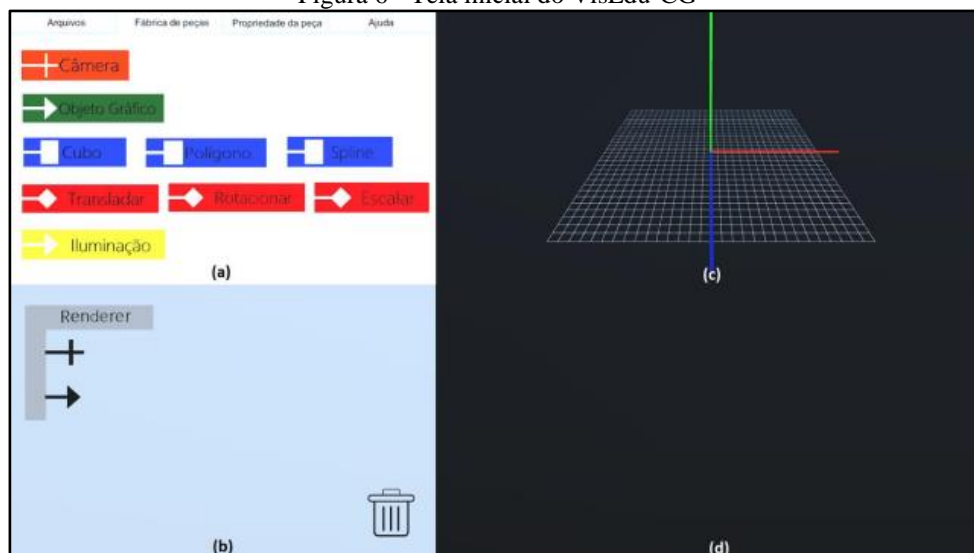
Fonte: Siedler *et al.* (2022).

### 3 SOFTWARE ATUAL

Ao longo dos anos, o VisEdu-CG já passou por diversas versões: tendo as duas primeiras em C++ (Araújo, 2012; Schramm, 2012), as três seguintes em Three.js (Nunes, 2014; Montibeler, 2014; Koehler, 2015) e a atual em Unity (Buttenberg, 2020), cuja tecnologia se manterá nessa nova versão proposta. Inicialmente chamado de Adubo e posteriormente de VisEdu-CG, a ferramenta surgiu com o objetivo de auxiliar os alunos da disciplina de Computação Gráfica do curso de Ciência da Computação da FURB a compreender melhor os temas abordados em aula, os principais sendo: grafo de cena, objetos gráficos, transformações geométricas homogêneas, câmera sintética e iluminação.

Buttenberg (2020) projetou a última versão do antigo nomeado VisEdu-CG em Unity, na versão 2018.2.6f1, a fim de aprimorar para uma ferramenta mais popular. Ao inicializá-la, o usuário pode optar por um tutorial de sete passos para aprender a usar a ferramenta. Nesta ferramenta são apresentadas quatro seções de tela distintas: Fábrica de Peças (Figura 6 (a)), na qual o usuário pega os blocos para programar; Renderer (Figura 6 (b)), em que o usuário deposita as peças que coletou na fábrica; Ambiente Gráfico (Figura 6 (c)), no qual é possível visualizar os eixos, grade e objetos colocados em cena; e Visualizador (Figura 6 (d)), que mostra o resultado da execução do que foi programado pelo usuário sem a presença da grade e dos eixos de orientação presentes na tela de Ambiente Gráfico.

Figura 6 - Tela inicial do VisEdu-CG



Fonte: Buttenberg (2020).

A Fábrica de Peças (Figura 6 (a)) apresenta nove diferentes tipos de objetos e componentes de cena, sendo eles: Câmera, Objeto Gráfico, Cubo, Polígono, Spline, Transladar, Rotacionar, Escalar e Iluminação. O Objeto Gráfico, Cubo, Polígono e Spline são formas geométricas para dispor no espaço gráfico. Os objetos Transladar, Rotacionar e Escalar são responsáveis pela matriz geométrica (transformações geométricas homogêneas), podendo mudar a posição, orientação e o tamanho no espaço do objeto em que forem aplicados. A Câmera e Iluminação são fundamentais para o funcionamento da aplicação, visto

que a Câmera possibilitará a visualização do resultado e a Iluminação permitirá que os objetos sejam vistos em cena (Buttenberg, 2020).

Ao selecionar o bloco desejado, o usuário deve arrastá-lo até o *Renderer* (Figura 6(b)), encaixando conforme formato da peça. Ao inserir um objeto geométrico, é possível adicionar tanto a iluminação quanto os objetos da matriz geométrica. Ao selecioná-los, é possível excluir o objeto ou editar suas propriedades, que aparecerem no canto superior esquerdo. Enquanto o aluno vai adicionando blocos, é possível pré-visualizar o resultado na tela de *Ambiente Gráfico* (Figura 6(c)), podendo fazer alterações nos valores de *Transladar*, *Rotacionar* e *Escalar* para obter o resultado desejado, que é apresentado na tela *Visualizador* (Figura 6(d)) (Buttenberg, 2020).

Ao concluir o projeto, Buttenberg (2020) demonstra que os objetivos específicos foram parcialmente cumpridos, visto que algumas funcionalidades propostas, os objetos *Polígono* e *Spline* e algumas propriedades da câmera, não foram implementados. Além disso, o tutorial desenvolvido ficou limitado a poucas funções básicas. Ademais, o projeto não possui a funcionalidade de se fazer exercícios de treinamento do conteúdo relacionado a Computação Gráfica.

#### 4 PROPOSTA DA FERRAMENTA

Nesta seção será apresentada a justificativa do projeto (subseção 4.1) que será desenvolvido, bem como seus principais requisitos (subseção 4.2) e metodologia (subseção 4.3) que será utilizada.

##### 4.1 JUSTIFICATIVA

No Quadro 1 é apresentado um comparativo das características entre os trabalhos correlatos. Nas linhas são descritas as características e nas colunas os trabalhos.

Quadro 1 - Comparativo dos trabalhos correlatos e última versão do projeto

| Trabalhos Correlatos<br>Características | Scratchim<br>(Rodrigues;<br>Gomes;<br>Carneiro, 2022) | GeNiAl<br>(Barros;<br>Sousa; Viana,<br>2022) | OrbitAndo<br>(Siedler <i>et al.</i> ,<br>2022) | VisEdu-CG 5.0<br>(Buttenberg,<br>2020) |
|---|---|--|--|--|
| Existe interação por meio de peças      | X   |  |  | X                                      |
| É um software educacional               |   | X  | X  | X                                      |
| Apresenta exercícios para validação     | X   | X  | X  |  |
| Apresenta tutorial explicando o seu uso | X   |  |  | (+/-)                                  |
| Apresenta conteúdos teóricos            | X   |  | X  |  |
| Possui acesso off-line                  | X   |  | X  |  |
| Foi desenvolvido em Unity               |   |  | Dois dos três jogos                            | X                                      |
| Disponibilidade                         | Físico  | Web  | Multiplataforma                                | Web                                    |

Fonte: elaborado pelo autor.

Com base nas informações apresentadas no Quadro 1, tanto o VisEdu-CG quanto o Scratchim apresentam programação por blocos (interação por meio de peças), o primeiro de forma digital e o segundo de forma desplugada. Apesar deste último ser do âmbito educacional, ele não se enquadra na categoria software; portanto, não possui a segunda característica, diferente dos outros trabalhos apresentados acima. Em relação a possuir exercícios para validação do aprendizado, todos os trabalhos correlatos validam esse tópico, com exceção do VisEdu-CG (versão anterior do projeto). Já sobre os tutoriais, apenas o Scratchim e a última versão do VisEdu-CG (de forma limitada) entram nesta categoria. Sobre os conteúdos teóricos, Scratchim os apresenta por meio da apostila que acompanha o kit e o OrbitAndo apresenta tanto no primeiro quanto no segundo jogo de sua plataforma. Além disso, ambos podem ser utilizados de maneira off-line. Quanto a ser desenvolvido em Unity, o VisEdu-CG e a plataforma de astronomia preenchem os requisitos, este último apresentando dois dos três jogos feitos nesse mesmo motor de jogos. Por fim, o jogo desplugado foi disponibilizado apenas em um modelo físico, enquanto o jogo GeNiAl e o VisEdu-CG são web e o OrbitAndo é multiplataforma.

A partir das conclusões destes trabalhos correlatos, se verifica a continuidade da eficácia dos materiais didáticos em contribuir para com o aprendizado. Dentre as técnicas utilizadas, destacam-se a presença conteúdo teórico e exercícios de fixação/validação do assunto. Além delas, se evidencia a importância de um manual explicando o funcionamento da ferramenta que se é apresentada.

Também, em conversa com o professor de Computação Gráfica Reis (2023), verificou-se que os alunos têm uma grande dificuldade em aprender o conteúdo desta disciplina por não conseguirem fazer uma abstração do espaço 3D. Para contribuir com o aprendizado de CG, esse projeto propõe continuar com o antigo VisEdu-CG, implementando as funcionalidades faltantes e trazendo novas, como a interface com mudança de tema (claro e escuro) para o usuário escolher o que mais lhe agrada à vista, além de exercícios para fixação do conteúdo. Por meio desse projeto, busca-se auxiliar o entendimento dos alunos sobre os assuntos abordados durante as aulas de CG. Sendo assim, com base nas principais características e funcionalidades observadas dos trabalhos correlatos e



da última versão do projeto, foram extraídos os pontos mais importantes para a versão proposta por esse projeto. Com base nelas, decidiu-se trazer ao projeto exercícios de validação, um tutorial mais sucinto, conteúdo teórico na justificativa de resposta dos exercícios e acesso off-line para aqueles que fizerem uso do acesso via desktop no sistema operacional Windows.

E assim, se espera como contribuição tecnológica disponibilizar uma aplicação Web com recursos gráficos que reúna conteúdo de CG, com a possibilidade de praticar exercícios relacionados ao conteúdo desta disciplina. E como contribuição social, disponibilizar uma forma mais interativa e visual do espaço 3D para trabalhar com os conteúdos de CG.

#### 4.2 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO

Os requisitos do projeto definem que o ambiente gráfico deve:

- a) permitir que o usuário possa seguir um tutorial para auxiliar o entendimento da ferramenta (Requisito Funcional – RF);
- b) permitir que o usuário possa arrastar os blocos e editar suas informações conforme for desejado (RF);
- c) permitir que o usuário possa mexer no tema da aplicação (modo claro ou modo escuro) conforme melhor lhe agradar (RF);
- d) permitir que o usuário tenha os blocos *Spline* e Polígono disponibilizados na Fábrica de Peças (RF);
- e) permitir que o usuário possa fazer uso da câmera com todas as suas propriedades para que funcione correta e completamente (RF);
- f) permitir que o usuário possa realizar atividades pré-definidas, a fim de treinar seus conhecimentos adquiridos (RF);
- g) permitir que o usuário saiba se acertou a atividade de treinamento ou não e que tenha uma explicação do porquê do erro (RF);
- h) utilizar o motor de jogos Unity em conjunto com a Integrated Development Environment (IDE) Visual Studio (Requisito Não Funcional – RNF);
- i) utilizar a linguagem de programação C# para implementação (RNF);
- j) ser desenvolvido para plataforma desktop (apenas para Windows) e web (RNF).

#### 4.3 METODOLOGIA

O trabalho será desenvolvido observando as seguintes etapas:

- a) revisão bibliográfica: rever os conceitos abordados no presente trabalho, complementando o que for necessário;
- b) elicitação de requisitos: rever os requisitos e, se necessário, adicionar novos;
- c) modelagem de diagramas: diagramar modelos Unified Modeling Language (UML) requisitados para o desenvolvimento da estrutura do projeto com a ferramenta Astah;
- d) atualizar projeto: refatorar o projeto anterior para uma versão mais recente do Unity;
- e) aprimorar funcionalidades do projeto: implementar as funcionalidades que não puderam ser realizadas no projeto anterior e outras cuja necessidade surgir durante o desenvolvimento, através do Unity e Visual Studio, por meio da linguagem de programação C#;
- f) aprimorar aparência do projeto: desenvolver funcionalidade de escolha de tema (claro ou escuro);
- g) adicionar exercícios: implementar fase de exercícios para treinamento do assunto, usando as mesmas ferramentas e linguagem do item (e);
- h) teste das funcionalidades: utilizar listas de exercícios do conteúdo da disciplina de CG para avaliar as funcionalidades desenvolvidas;
- i) teste com usuários: testar o uso da ferramenta pelos alunos durante uma aula da disciplina de CG;
- j) implementar melhorias: implementar o que for apontado no teste com usuários.

As etapas serão realizadas nos períodos relacionados no Quadro 2.

Quadro 2 - Cronograma

| etapas / quinzenas                   | 2024 |   |      |   |      |   |      |   |      |   |
|--------------------------------------|------|---|------|---|------|---|------|---|------|---|
|                                      | fev. |   | mar. |   | abr. |   | maio |   | jun. |   |
|                                      | 1    | 2 | 1    | 2 | 1    | 2 | 1    | 2 | 1    | 2 |
| revisão bibliográfica                |      |   |      |   |      |   |      |   |      |   |
| elicitación de requisitos            |      |   |      |   |      |   |      |   |      |   |
| modelagem de diagramas               |      |   |      |   |      |   |      |   |      |   |
| atualizar projeto                    |      |   |      |   |      |   |      |   |      |   |
| aprimorar funcionalidades do projeto |      |   |      |   |      |   |      |   |      |   |
| aprimorar aparência do projeto       |      |   |      |   |      |   |      |   |      |   |
| adicionar exercícios                 |      |   |      |   |      |   |      |   |      |   |
| teste das funcionalidades            |      |   |      |   |      |   |      |   |      |   |
| teste com usuários                   |      |   |      |   |      |   |      |   |      |   |
| implementar melhorias                |      |   |      |   |      |   |      |   |      |   |

Fonte: elaborado pelo autor.

## 5 REVISÃO BIBLIOGRÁFICA

Nessa seção serão descritos os principais conceitos que servirão como base para esse projeto: abstração do espaço 3D (subseção 5.1) e Computação Gráfica (subseção 5.2).

### 5.1 ABSTRAÇÃO DO ESPAÇO 3D

Segundo Azevedo, Conci e Vasconcelos (2022, p. 35), “[...] a abstração matemática dita Sistema de Coordenadas é explorada pela Computação Gráfica como ferramenta que permite escolher e alterar a representação de objetos gráficos de maneira que for mais conveniente a cada operação de processamento visual.”. Como Settimy e Bairral (2020) observaram, os alunos possuem dificuldade na abstração do espaço 3D pelo fato do ensino básico não abordar a geometria de forma mais clara e aprofundada. Segundo Settimy e Bairral (2020, p. 3), “[...] a Geometria é um campo fértil para perceber e entender as formas geométricas presentes em nosso cotidiano, sendo possível desenvolver habilidades importantes como a experimentação, representação, descrição e argumentação.”.

Entre as ferramentas de auxílio para aprendizado de Geometria se tem o Geogebra 3D. Com o Geogebra 3D, os usuários podem construir qualquer figura geométrica 3D e manipular seus valores de escala, rotação e translação. Dessa forma, utilizando esse espaço de visualização 3D, contribui mais com o aprendizado, conforme observado por Fassarella e Rocha (2018). Ao considerar que a Computação Gráfica se baseia fundamentalmente em Geometria, esta se torna indispensável para o aprendizado de CG.

Conforme observado por Settimy e Bairral (2020) e Azevedo, Conci e Vasconcelos (2022) isso se dá pelo fato de que, além de abstrair o espaço 3D, também é necessário entender o conceito de matriz de transformação homogênea e aplicá-la a objetos gráficos da cena, sendo necessário conhecimentos da área da geometria

### 5.2 COMPUTAÇÃO GRÁFICA

Conforme dito por Manssour e Cohen (2006, p. 1), computação gráfica “[...] é uma área da Ciência da Computação que se dedica ao estudo e desenvolvimento de técnicas e algoritmos para a geração (síntese) de imagens através do computador.”. E, como visto por Azevedo, Conci e Vasconcelos (2022, p. 183), “isso se faz a partir da descrição da geometria dos objetos [...], dos materiais associados às suas superfícies [...], das fontes de luz e do modelo de representação da iluminação adotado, da definição de uma câmera virtual que estabelece a posição de observação de cena, [...]”. Sendo assim, para maior entendimento do assunto, é necessário o conhecimento de outros conceitos dentro dessa temática, sendo eles: grafo de cena, objetos gráficos, transformações geométricas homogêneas, câmera sintética e iluminação.

Conforme Silva, Raposo e Gattas (2004, p. 3), “[...] grafos de cena são ferramentas conceituais para representação de ambientes virtuais tridimensionais nas aplicações de computação gráfica.”. Isso significa que o grafo é uma espécie de mapa para a cena construída, mostrando quais objetos gráficos fazem parte dela, quais objetos possuem filhos, quais suas características (cor, textura, posicionamento etc.). Azevedo, Conci e Vasconcelos (2022, p. 183) também afirmam que “[...] é comum que os objetos sejam descritos como malhas poligonais, compostas por conjuntos de vértices e arestas.”. Sendo assim, objetos gráficos são formas compostas por coordenadas que são mapeadas e representadas no mundo gráfico.

Para entender o conceito de transformações geométricas homogêneas, primeiro é preciso conceituar transformação em si. Conforme dito por Azevedo, Conci e Vasconcelos (2022, p. 52), transformação “[...] é qualquer função  $f$  que realiza um mapeamento de um conjunto de entrada, dito domínio, em um conjunto de saída, dito contradomínio.”. Dito isso, transformações geométricas homogêneas são funções que alteram o valor inicial das coordenadas e são aplicadas igualmente a todos os pontos de um objeto gráfico. Dentre as transformações

existentes, destacam-se: rotação, escalamento e translação. A função de rotação é responsável por rotacionar os pontos, podendo ser no sentido horário ou anti-horário, para uma nova posição a partir da sua origem. Escalamento seria a transformação usada para alterar o tamanho de um objeto gráfico, podendo tanto aumentar quanto diminuir sua escala. Já a translação, é o módulo necessário para mudar a posição de um objeto a partir de sua origem. Esses três tipos de transformações são comumente usados em conjunto para se obter o resultado desejado e, por isso, acabam sendo complementares umas das outras (Azevedo; Conci; Vasconcelos, 2022).

Uma câmera sintética, também conhecida como câmera virtual, “[...] define um ponto de vista sob o qual a cena será visualizada e com isso cria uma representação no sistema de Computação Gráfica para o observador da cena.” (Azevedo; Conci; Vasconcelos, 2022, p. 38). Dessa forma, ela é necessária para a visualização dos objetos gráficos na cena. Vale ressaltar que apenas serão vistos em cena os objetos gráficos alinhados com o volume de visão da câmera, que seria toda a área visível a partir da sua localização. Para dispor devidamente a câmera, é preciso ter sua localização e orientação no espaço, o tipo de projeção que realizará e como ela interpretará os dados das imagens que serão visualizadas (Azevedo; Conci; Vasconcelos, 2022).

Como a câmera fica na cena junto com os outros objetos, ela também é um objeto gráfico, sendo preciso definir suas coordenadas e sua orientação (para onde ela está olhando). A projeção trata sobre como o objeto gráfico será visto em cena, podendo ser do tipo paralela, que mantém a linha de projeção seguindo os pontos de forma paralela entre si (muito usado em projeção 2D), ou sob perspectiva, fazendo com que objetos mais próximos apareçam maiores do que os mais distantes do ponto de visualização (projeção mais usada no 3D). A forma em que a câmera interpretará se relaciona com os outros dois aspectos anteriores. Para projetar a imagem, é preciso saber seu centro (para inserir no lugar correto) e sua escala (para ficar do tamanho desejado), para então mostrá-la de forma adequada (Azevedo; Conci; Vasconcelos, 2022).

Para que os objetos gráficos sejam percebidos em cena, é preciso a presença de iluminação para a percepção de suas cores e texturas. Existem quatro tipos de luz: a ambiente, a direcional, a pontual e a holofote. A luz ambiente é a mais comum e simples de se utilizar. Ela funciona como uma luz global, iluminando a cena de forma igualitária, permitindo que todos os objetos sejam visualizados, mas sem produzir grandes efeitos de reflexão e sombreamento. A luz direcional é a utilizada para simular a luz solar: ela vem de um ponto específico e segue a angulação, traçando raios paralelos de luz entre si. É importante lembrar que esse tipo de iluminação considera que todos os raios emitem quantidade equivalente de luz. A luz pontual é um ponto no espaço que ilumina em todas as direções e apresenta intensidades de luz diferentes conforme afastamento da origem. Por se tratar de um ponto, é usada para representar lâmpadas, explosões, entre outros tipos de objetos com pontos luminosos. Por fim, a luz holofote, como o próprio nome diz, é a luz proveniente de uma lâmpada do tipo holofote, iluminando apenas a região abrangente pelo seu ângulo de abertura, reduzindo de intensidade conforme afastamento (Azevedo; Conci; Vasconcelos, 2022).

## REFERÊNCIAS

ARAÚJO, Luciana P. de. **Adubogl: Aplicação Didática usando a Biblioteca Open GL**. 2012. 76f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau, 2012.

AZEVEDO, Eduardo; CONCI, Aura; VASCONCELOS, Cristina. **Computação Gráfica: Teoria e Prática: Geração de Imagens**. 1. ed. rev. Rio de Janeiro: Alta Books, 2022.

BARROS, Gabriel C.; SOUSA, Janyeid K. C.; VIANA, Davi. Jornada Química GeNiAl: um jogo sério para o ensino da tabela periódica e seus elementos. *In: CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO*, 11., 2022, Manaus. **Anais** [...]. Manaus: Publication chair, 2022. p. 1-12. Disponível em: <https://sol.sbc.org.br/index.php/sbie/article/view/22432/22256>. Acesso em: 27 nov. 2023.

BUTTENBERG, Peterson B. **VisEdu-CG 5.0: Visualizador de material educacional**. 2020. 19 f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau, 2020. Disponível em: [https://www.furb.br/dsc/arquivos/tccs/monografias/2020\\_1\\_peterson-bonibuttenberg\\_monografia.pdf](https://www.furb.br/dsc/arquivos/tccs/monografias/2020_1_peterson-bonibuttenberg_monografia.pdf). Acesso em: 27 nov. 2023.

FASSARELLA, Lucio S.; ROCHA, Rosângelo J. da. Geogebra 3D: Relato de uma experiência na superação de dificuldades de aprendizagem em geometria espacial. **Kiri-kerê**, São Mateus, v. 3, n. 5, p. 261-275, nov. 2018. Disponível em: <https://periodicos.ufes.br/kirikere/article/view/20347/14547>. Acesso em: 28 nov. 2023.

KOEHLER, William F. **VisEdu-CG 4.0: Visualizador de Material Educacional**. 2015. 90 f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau, 2015. Disponível em: [https://www.furb.br/dsc/arquivos/tccs/monografias/2015\\_1\\_william-fernandes-koehler\\_monografia.pdf](https://www.furb.br/dsc/arquivos/tccs/monografias/2015_1_william-fernandes-koehler_monografia.pdf). Acesso em: 28 nov. 2023.

MANSSOUR, Isabel H.; COHEN, Marcelo. Introdução à computação gráfica. **Revista de Informática Teórica e Aplicada**, Rio Grande do Sul, v. 13, n. 2, p. 1-25, 2006. Disponível em: <https://www.inf.pucrs.br/manssour/Publicacoes/TutorialSib2006.pdf>. Acesso em: 1 out. 2023.



- MONTIBELER, James P. **VisEdu-CG: Aplicação Didática para Visualizar Material Didático, Módulo de Computação Gráfica**. 2014. 106 f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau, 2014. Disponível em: [https://www.furb.br/dsc/arquivos/tccs/monografias/2014\\_1\\_james-perkison-montibeler\\_monografia.pdf](https://www.furb.br/dsc/arquivos/tccs/monografias/2014_1_james-perkison-montibeler_monografia.pdf). Acesso em: 28 nov. 2023.
- NUNES, Samuel A. **VisEdu-CG 3.0: Aplicação Didática para Visualizar Material Didático: Módulo de Computação Gráfica**. 2014. 89 f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau, 2014. Disponível em: [https://www.furb.br/dsc/arquivos/tccs/monografias/2014\\_1\\_samuel-anderson-nunes\\_monografia.pdf](https://www.furb.br/dsc/arquivos/tccs/monografias/2014_1_samuel-anderson-nunes_monografia.pdf). Acesso em: 28 nov. 2023.
- REIS, Dalton S. **Entrevista sobre aulas de Computação Gráfica**. Entrevistador: Natália Sens Weise. Blumenau. 2018. Entrevista feita através de conversação – não publicada.
- RODRIGUES, Amanda K. M.; GOMES, Kamily C. O.; CARNEIRO, Murillo G. Scratchim: uma abordagem para o ensino do Pensamento Computacional para crianças de forma remota e desplugada. In: CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 11., 2022, Manaus. **Anais** [...]. Manaus: Publication chair, 2022. p. 1-12. Disponível em: <https://sol.sbc.org.br/index.php/sbie/article/view/22515/22339>. Acesso em: 27 nov. 2023.
- SCHRAMM, Elizandro J. **Adubogl ES 2.0: Aplicação Didática usando a Biblioteca OpenGL EE 2.0 no iOS**. 2012. 64f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau, 2012.
- SETTIMY, Thaís F. de O.; BAIRRAL, Marcelo A. Dificuldades envolvendo a visualização em geometria espacial. **VIDYA**, Santa Maria, v. 40, n. 1, p. 177-195, jan./jun. 2020. Disponível em: [https://www.researchgate.net/publication/343556166\\_DIFICULDADES\\_ENVOLVENDO\\_A\\_VISUALIZACAO\\_EM\\_GEOMETRIA\\_ESPACIAL](https://www.researchgate.net/publication/343556166_DIFICULDADES_ENVOLVENDO_A_VISUALIZACAO_EM_GEOMETRIA_ESPACIAL). Acesso em: 28 nov. 2023.
- SIEDLER, Marcelo S. *et al.* OrbitAndo: uma plataforma para ensino de Astronomia de outro mundo. In: CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 11., 2022, Manaus. **Anais** [...]. Manaus: Publication chair, 2022. p. 1-11. Disponível em: <https://sol.sbc.org.br/index.php/sbie/article/view/22434/22258>. Acesso em: 27 nov. 2023.
- SILVA, Romano J. M. da; RAPOSO, Alberto B.; GATTAS, Marcelo. **Grafo de Cena e Realidade Virtual**. Rio de Janeiro: PUC, 2004. Disponível em: [https://web.tecgraf.puc-rio.br/~abraposo/INF1366/2007/02\\_GrafoDeCena\\_texto.pdf](https://web.tecgraf.puc-rio.br/~abraposo/INF1366/2007/02_GrafoDeCena_texto.pdf). Acesso em: 27 nov. 2023.

## PROJETO: OBSERVAÇÕES – PROFESSOR ORIENTADOR

Observações do orientador em relação a itens não atendidos do pré-projeto (se houver):

No quadro de “Comparativo dos trabalhos correlatos e última versão do projeto” (Quadro 1) optamos por manter a coluna do trabalho anterior (VisEdu-CG) a ser estendido neste projeto por achar importante também fazer uma correção com este projeto.