

CURSO DE CIÊNCIA DA COMPUTAÇÃO – TCC		
( ) PRÉ-PROJETO	( X ) PROJETO	ANO/SEMESTRE: 2021/1

## REFATORANDO E ADICIONANDO NOVAS FUNCIONALIDADES NO VISEDU-CG COM MOTOR DE JOGOS UNITY

Douglas Eduardo Bauler

Prof. Dalton Solano dos Reis – Orientador

### 1 INTRODUÇÃO

A tecnologia está em constante evolução de maneiras muito significativas, melhorando o dia a dia, aumentando a produtividade e o entendimento de vários assuntos. Existem ferramentas interativas de ensino que dão interlúdio ao assunto a ser abordado, tornando-o de uma maneira mais lúdica na qual muitas vezes pode se dificultar o aprendizado.

A contribuição didática para uma pedagogia voltada para o sujeito requer assumir, entre outras coisas, o uso das mídias e das tecnologias da educação. O professor deve ser capaz de utilizar aparatos tecnológicos não apenas para seu uso próprio, mas trabalhar com esses recursos em sala de aula, em favor da aprendizagem dos alunos (SILVA, 2010, p.6).

Por meio dessas metodologias a ferramenta VisEdu-CG tem como objetivo trazer essas melhorias no aprendizado aos acadêmicos da matéria de Computação Gráfica. Conforme Bittenberg (2020, apud REIS, 2020, p. 1), “o VisEdu-CG é um projeto para desenvolver uma plataforma Web que permita os alunos da disciplina de Computação Gráfica do curso de Ciências da Computação praticarem os conceitos ministrados nesta disciplina”, estando atualmente na versão 5.0 do projeto.

Essa aplicação contou com o desenvolvimento de vários módulos específicos, dentre eles pode-se citar o motor de jogos, matemática, estatística, processamento de imagens, realidade aumentada e simulação. Para que a ferramenta tenha uma evolução constante foi realizado um processo de migração de linguagem e refatoração do código. Existe uma carência maior em relação às funcionalidades do WebGL, o que torna o VisEdu-CG um sistema igualmente limitado (BUTTENBERG, 2020).

Um processo de migração de uma ferramenta já consolidada numa linguagem, não é um processo simples de realizar, devido à complexidade com diversas funcionalidades da ferramenta, tendo também o curto espaço de tempo para seu desenvolvimento e a falta de estrutura do código para melhor entendimento e manutenção, fazendo com que a migração não seja realizada completamente. Em razão dessas dificuldades, este trabalho propõe continuar o processo de migração das funcionalidades, assim como a refatoração do código já migrado para melhor compreensão, manutenção e adição de novas funções, utilizando a motor de jogos Unity.

#### 1.1 OBJETIVOS

O objetivo deste trabalho é disponibilizar novas funcionalidades ao sistema atual e estender o trabalho intitulado “VisEdu-CG 5.0: Visualizador de Material Educacional” (BUTTENBERG, 2020).

Os objetivos específicos são:

- a) disponibilizar bloqueio de campos ao realizar importação/exportação de um projeto feito na ferramenta;
- b) disponibilizar novas peças do tipo Iteração, Polígonos e Spline.

### 2 TRABALHOS CORRELATOS

São apresentados três trabalhos correlatos com características semelhantes aos objetivos do trabalho proposto. O primeiro trabalho é uma ferramenta chamada Duolingo, aplicativo para auxílio de aprendizado de múltiplas linguagens e multiplataforma, sendo utilizado em forma de um jogo com desafios diários e metas e recompensas com objetivo de estimular o estudo de outras línguas. O segundo trabalho é o QuestMeter, conforme descreve Vieira (2019), é uma ferramenta de quiz construída com elementos de gamificação juntamente com o conceito de *Clickers*. E o terceiro trabalho é o Toweljs, conforme descreve Zanluca (2018), é um motor gráfico que utiliza JavaScript e WebGL, com objetivo de facilitar a implementação e abstrair o uso dessas duas ferramentas.

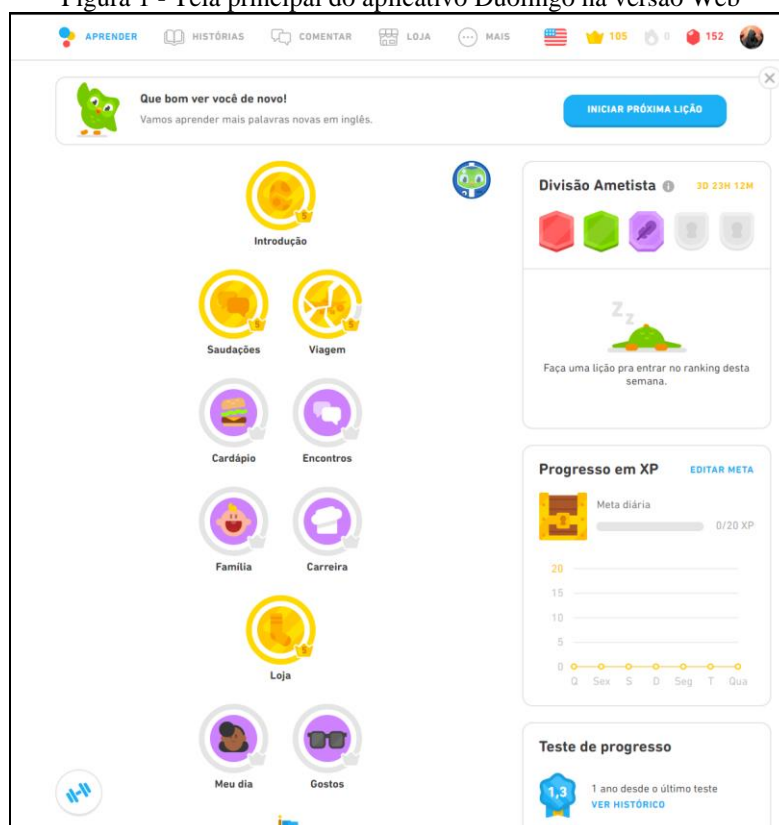
#### 2.1 DUOLINGO

O Duolingo é mundialmente conhecido por se tratar de um aplicativo de aprendizagem de idiomas com aparência de jogo. Assim, servindo-se de regras, pontuações, moedas e punições, estimulando atividades diárias

de curta duração (em torno de vinte minutos) premiando os usuários regulares. Além disso, a acessibilidade do aplicativo permite que as atividades sejam praticadas a qualquer hora e em qualquer lugar devido à portabilidade dos smartphones e tablets que servem de suporte para essa ferramenta, o que aumenta as chances de participação dos aprendizes que dispõem de pouco tempo para atividades extras (MELO, 2021).

Nas versões mais atuais, o Duolingo oferece um ambiente interativo tendo uma progressão de atividades realizadas. Cada um com uma aula/tutorial explicando pronúncias e como será abordado as questões sobre cada assunto. A versão gratuita possui vidas onde geralmente são três, ou seja, errando três atividades será obrigatório ter que aguardar elas serem recarregadas para realizar novas tentativas, caso seja pago, suas vidas e tentativas serão ilimitadas. Ao longo de todos os testes é sempre possível realizar uma tentativa de aptidão da linguagem, para verificar seu nível de desempenho no conhecimento de novas palavras, por exemplo. Na Figura 1 é possível visualizar a tela principal do Duolingo na versão web.

Figura 1 - Tela principal do aplicativo Duolingo na versão Web



Fonte: elaborado pelo autor.

A experiência com o Duolingo no ensino formal de alemão durante um semestre letivo em um curso de Letras/Alemão com estudantes em níveis iniciais de aprendizagem mostrou-se relevante para ampliar o contato com o idioma e expandir o vocabulário dos aprendizes. Além de ser um recurso gratuito, a acessibilidade, a interatividade bem como a gamificação do aplicativo móvel, são recursos importantes que podem contribuir com a regularidade de estudo por meio dos lembretes diários para realização de tarefas, além de estimular a prática com pontuações e premiações que tendem a manter o usuário motivado e engajado no estudo (MELO, 2021). Nas suas versões atuais, possui mais tutoriais interativos explicando o funcionamento de cada lição auxiliando na execução dos exercícios.

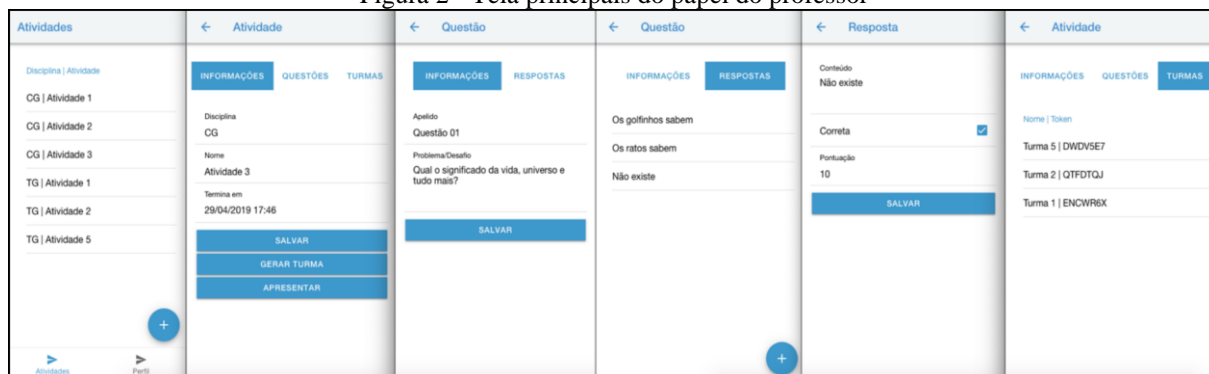
## 2.2 QUESTMETER

É uma ferramenta de quiz construída com elementos de gamificação com o conceito de *Clickers*. Tendo como objetivo auxiliar os professores na realização de atividades diversificadas para motivar e engajar os alunos em sala de aula, além disso, outro propósito da ferramenta é testar a interação dos alunos com ferramentas diferenciadas em sala. A ferramenta foi desenvolvida utilizando o framework Ionic e a plataforma Firebase (VIEIRA, 2019).

De acordo com Vieira (2019), a gamificação pode ser observada em dois pontos da ferramenta, nas pontuações recebidas pelos alunos ao realizarem as atividades e na apresentação dessas atividades. Dois papéis de usuário: o papel do professor e o papel do aluno. Cada um desses papéis possui uma visão diferente da

ferramenta. Conforme descreve o autor, o papel do professor na ferramenta é manter a funcionalidades como: atividades, questões dentro das atividades, respostas dentro das questões, gerar turmas dentro de atividades, apresentar a atividade criada para os alunos e controlar o andamento da apresentação, podendo prosseguir as questões. Na Figura 2 são mostradas as principais telas disponibilizadas para o papel do professor.

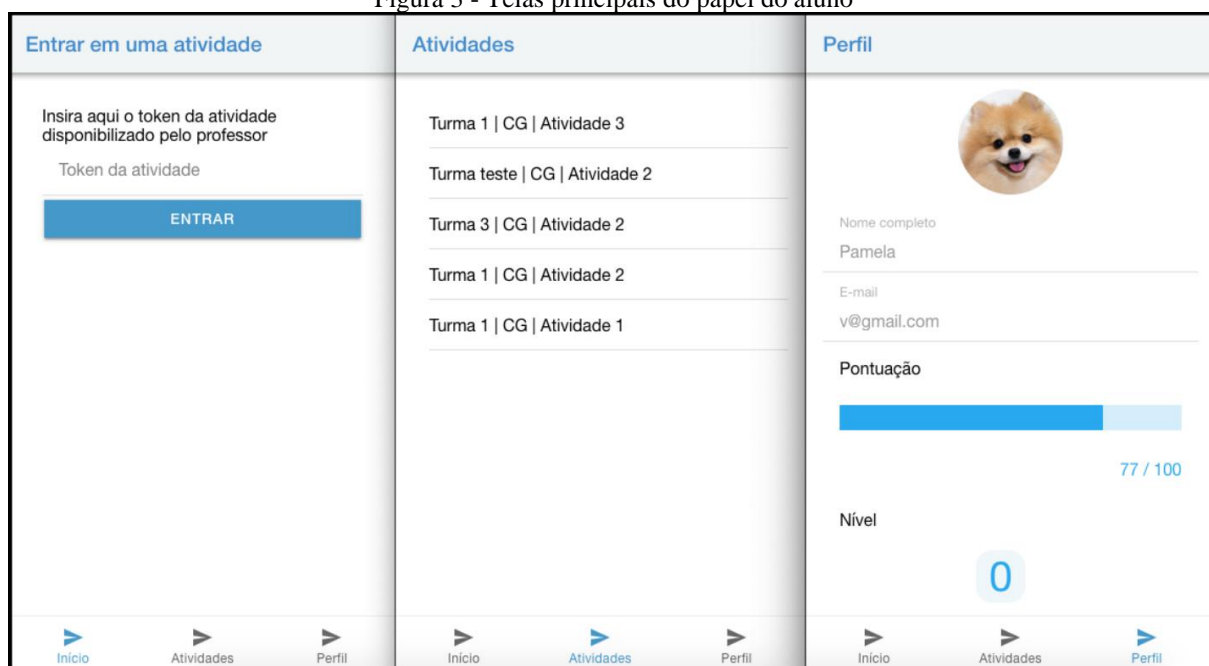
Figura 2 - Tela principais do papel do professor



Fonte: Vieira, 2019.

Segundo Vieira (2019), o papel do aluno é realizar em sala de aula as atividades apresentadas pelo professor. Dentre suas funcionalidades inclui ingressar em atividades disponibilizadas, escolher as opções oferecidas em cada questão da atividade, visualizar as respostas escolhidas, podendo identificar quais as respostas certas e erradas, verificando seu progresso na ferramenta na tela de perfil. A Figura 3 é mostrado as principais telas disponibilizadas para o papel do aluno.

Figura 3 - Telas principais do papel do aluno



Fonte: Vieira, 2019.

Vieira (2019) conclui que em seu projeto, foram cumpridos os objetivos definidos, embora os resultados de usabilidade, engajamento e motivação, obtidos através dos testes realizados com os alunos, tenham sido razoáveis. Os *feedbacks* recebidos dos alunos e dos professores durante os testes e comentários disponíveis nos formulários, foram positivos em sua maioria. Já os resultados dos questionários respondidos pelos professores, demonstraram que a ferramenta cumpriu com seu objetivo de motivar e engajar os alunos em atividades realizadas em sala de aula.

## 2.3 TOWELJS

Segundo Zanluca (2018), Toweljs é uma implementação de um motor de jogos utilizando JavaScript e WebGL, tendo como principal objetivo facilitar a implementação e aumentar o nível de abstração para aplicações desenvolvidas utilizando essas duas ferramentas. O motor por sua vez disponibiliza a criação de

objetos gráficos (cubos e esferas) e luzes permitido juntar tudo numa cena, permite também a criação de dois tipos diferentes de câmara sintética (perspectiva e ortogonal). Tudo isso feito utilizando uma arquitetura baseada em componentes, que ajudou na organização e facilitará futuras expansões do código. Implementou-se junto comportamentos comuns em motores de jogos com o intuito de isolá-los para que seja fácil de reaproveitá-los sem passar por um grande trabalho a cada nova aplicação.

Em suas conclusões do projeto, Zanluca (2018) descreve que dentre as dificuldades durante o desenvolvimento, vale destacar a forma como foram implementadas as transformações geométricas e as luzes. Para evitar o processamento desnecessário do recálculo da matriz de transformação do objeto a cada vez que o desenhasse, optou-se por recalculá-la somente quando realmente houvesse alguma alteração nos seus valores. O autor também afirma a respeito da arquitetura utilizada, baseada em componentes, na qual se mostrou eficiente para ser utilizada num motor de jogos, que poderá alcançar um baixo nível de acoplamento e um grande grau de reutilização dos componentes.

Durante os testes de usabilidade do motor de jogos desenvolvido por outro desenvolvedor, percebeu-se que o tutorial criado para que os auxiliasse, no uso do motor, precisava de melhoria. A principal dificuldade estava no entendimento da criação e manipulação dos componentes. Após melhorias no tutorial o motor se mostrou como uma ferramenta com potencial para ser usado por pessoas que não tenham um conhecimento tão aprofundado em computação gráfica. Visto que um nível de abstração conseguiu ser alcançado graças ao encapsulamento do código responsável pelo desenho e comportamento dos gameObject dentro dos seus respectivos componentes (ZANLUCA, 2018).

### 3 SISTEMA ATUAL

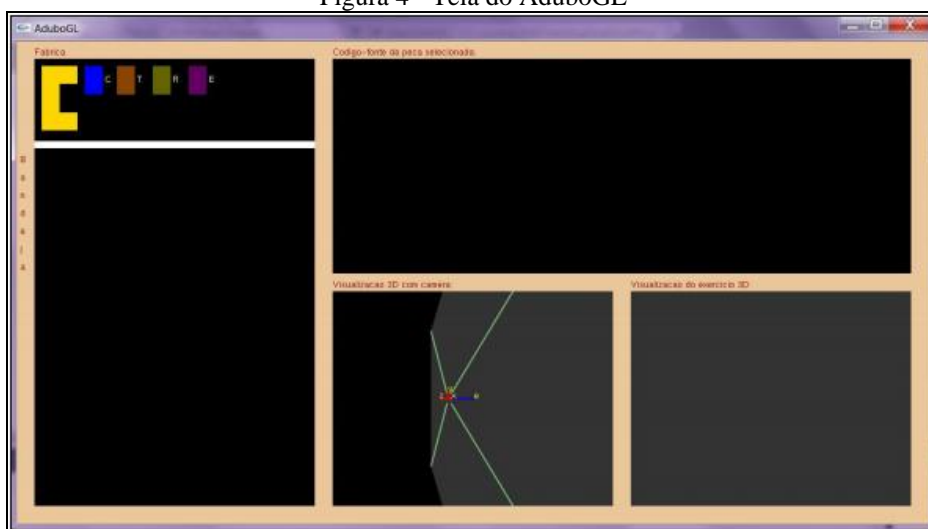
São apresentados três trabalhos, na qual foi dado continuidade. A seção 3.1 descreve a ferramenta AduboGL ferramenta voltada ao aprendizado da computação gráfica com foco nas transformações geométricas, utilizando a biblioteca OpenGL (ARAUJO, 2012), a seção 3.2 descreve VisEdu-CG 4.0: Visualizador de material educacional (KOEHLER, 2015) e a seção 3.2 descreve o VisEdu-CG 5.0: Visualizador de material educacional (BUTTENBERG, 2020), ambas aplicações utilizadas para metodologias de ensino-aprendizagem dos acadêmicos de computação gráfica.

#### 3.1 ADUBOGL

O AduboGL é uma ferramenta voltada ao aprendizado da computação gráfica com foco nas transformações geométricas. A biblioteca OpenGL é utilizada como ponto fundamental para montar o cenário 2D da aplicação e o resultado no espaço 3D. Implementado na linguagem C++ no ambiente do Microsoft Visual C++ 2010 Express para melhor integração com OpenGL. A parte fundamental da ferramenta é formada por peças que representam uma forma gráfica ou um comando importante da OpenGL para realizar uma interação no espaço 3D que será gerado pela junção das peças (ARAUJO, 2012).

Na Figura 4 é demonstrado a aplicação sendo executada. A primeira janela (esquerda), por sua vez, é dividida em duas partes, onde a primeira é a fábrica e a segunda a bandeja. A parte da fábrica apresenta as possíveis peças a serem utilizadas no desenvolvimento do exercício e a bandeja é usada para a montagem do exercício. A segunda janela é a de código-fonte que corresponde ao código da peça selecionada, sendo da fábrica ou da bandeja. As outras duas telas são utilizadas para a apresentação da cena em 3D, resultante do exercício montado (ARAUJO, 2012).

Figura 4 - Tela do AduboGL



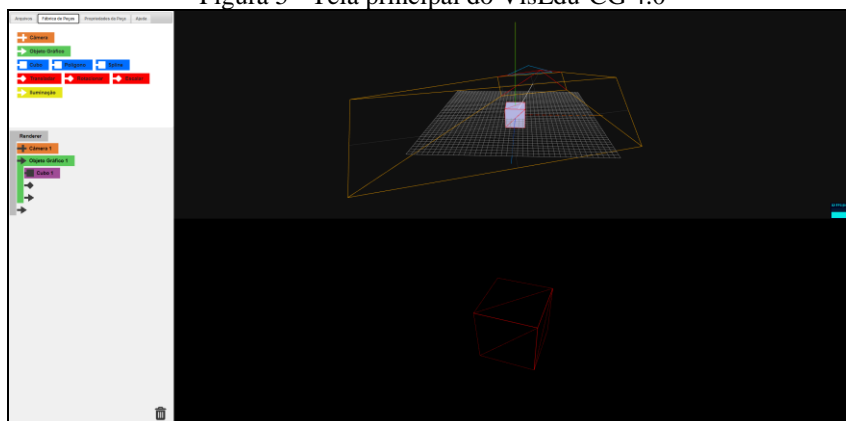
Fonte: Araújo, 2012.

Conforme afirma Araújo (2012), para que o aluno pudesse aprender os conceitos da Computação Gráfica de forma mais simplificada seria necessário disponibilizar além de um framework um ambiente onde o aluno pudesse treinar e trabalhar as funções disponíveis nele. Como não havia tempo hábil para o desenvolvimento dos dois, o framework e a aplicação, foi decidido focar no desenvolvimento da aplicação.

### 3.2 VISEDU-CG 4.0

Conforme descreve Koehler (2015), o VisEdu-CG 4.0 teve como principal objetivo implementar a ferramenta utilizando a biblioteca Tree.js e utilizar o motor de abstração WebGL, provendo uma forma fácil de manipular, criar elementos gráficos e melhorar sua usabilidade. O visualizador conta com HTML, JavaScript para montar sua interface de usuário, enquanto usa JQuery para definir os comportamentos mais complexos dela. Na Figura 5 é possível visualizar a interface da respectiva versão, onde são disponibilizadas quatro telas com o objetivo principal de melhorar o desempenho no aprendizado dos conceitos de Computação Gráfica.

Figura 5 - Tela principal do VisEdu-CG 4.0



Fonte: elaborado pelo autor.

Em suas conclusões Koehler (2015), conclui que neste processo, o motor foi adaptado para suportar a biblioteca Three.js e assim fornecer as capacidades tridimensionais necessárias (conforme objetivo proposto), onde o visualizador foi completamente refatorado para explorar outras técnicas, tecnologias e garantindo que ele opere de forma eficiente junto ao motor, sendo esta uma ferramenta útil e simples. Ele também relata que a utilização de HTML, CSS, JavaScript e JQuery permitiu um controle mais simples e eficaz sobre as peças, abas e janelas.

Koehler (2015), afirma que seus objetivos propostos foram cumpridos. Quanto as principais melhorias, o autor também destaca que devem ser aplicadas ao visualizador incluindo a criação de mais tipos peças, a fim de cobrir uma gama cada vez maior de conceitos gráficos, bem como a correção de alguns problemas que surgiram na utilização das propriedades das peças, principalmente com os polígonos.

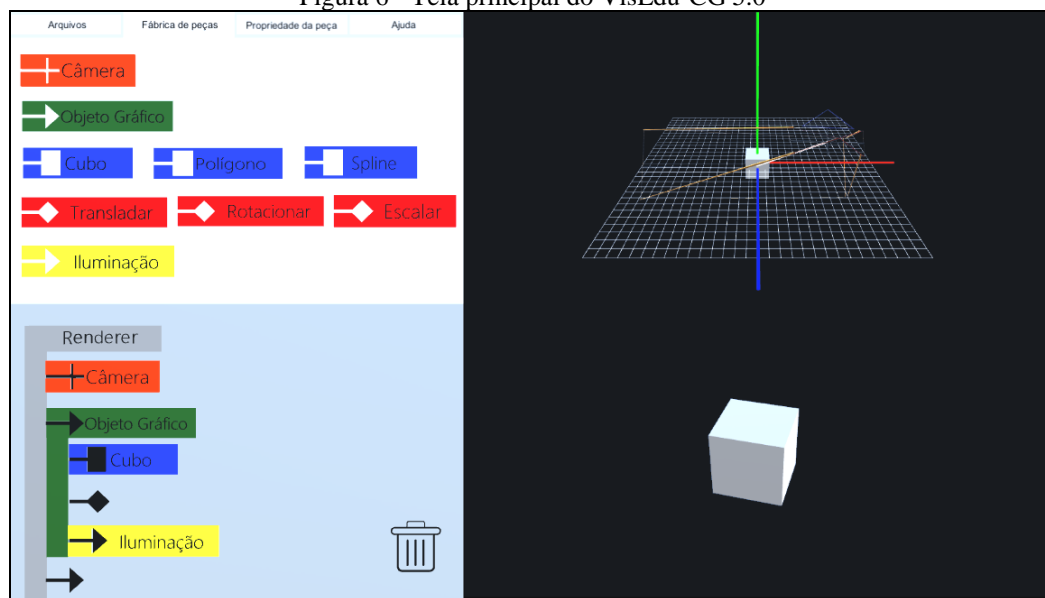
### 3.3 VISEDU-CG 5.0

O objetivo principal da versão 5.0 da ferramenta VisEdu-CG, foi realizar a migração para utilizar o motor gráfico Unity. Apesar de apresentar problemas de visualização em alguns objetos, como por exemplo, a iluminação *spot*, a plataforma teve resultado satisfatório na construção de cenários contendo conceitos básicos de computação gráfica, como as transformações geométricas e em conceitos com maior complexidade como é o caso das iluminações (BUTTENBERG, 2020).

O segundo objetivo foi criar uma proposta de tutorial informativo de maneira simples, onde a criação de uma cena destacasse os conceitos essenciais. Buttenberg (2020) salienta que o tutorial pode ser melhorado, como por exemplo a forma de exibição dos passos no tutorial. Além disso, algumas peças ainda não foram migradas como: o Polígono, o Spline e as melhorias na implementação da peça iluminação.

O terceiro objetivo, de utilizar representações visuais a partir de peças de encaixe para gerar uma cena gráfica, foi atingido. As peças importadas de uma ferramenta de criação de modelos 3D se comportaram adequadamente no Unity e os encaixes das peças nos slots foram bem-sucedidos. Quase todas as peças tiveram suas representações gráficas efetuadas, com exceção das peças spline e polígono (Figura 6). Ele destaca essas peças a serem adicionadas nas próximas versões. Além das funções de *look at*, *near* e *far* da câmera (BUTTENBERG, 2020).

Figura 6 - Tela principal do VisEdu-CG 5.0



Fonte: elaborado pelo autor.

Com relação a conclusão a respeito do resultado da ferramenta, Buttenberg (2018), destaca um nível no consumo de memória maior na maioria dos navegadores, com exceção do Google Chrome sendo estável. E a melhor plataforma em desempenho foi na versão desktop Windows. Por este motivo ele recomenda ser gerado executáveis não apenas para a versão web, e sim para as demais plataformas, visando a melhor resolução de desempenho da ferramenta e tornando-a multiplataforma. Sendo assim, também disponibilizou um tutorial com base nas funcionalidades disponíveis da versão, atendendo seus principais requisitos do projeto, como a migração, podendo manipular peças como: a câmera, o objeto gráfico, o cubo e iluminação, com exceção do Polígono, Spline e outras funcionalidades, como exportação/importação de projetos e a guia de ajuda. Além de que o tutorial inicialmente implementado não trata de todas as funções já desenvolvidas.

## 4 PROPOSTA DA FERRAMENTA

Neste capítulo tem o objetivo de apresentar a justificativa, requisitos e metodologia que será elaborado para o desenvolvimento.

### 4.1 JUSTIFICATIVA

No Quadro 1 é demonstrado um comparativo entre os trabalhos correlatos para entender melhor o trabalho proposto. Uma das características descritas são as transformações geométricas (translação, escala e rotação), utilizado apenas pelo Toweljs. Quanto a característica de motor de jogos, apenas o próprio Toweljs

trabalha sendo o próprio motor, tendo como objetivo aumentar o nível de abstração para aplicação que utilizam o JavaScript e WebGL, disponibilizando objetos gráfico, luzes e câmera na criação de uma cena.

Com relação a exportação de atividades, apenas o QuestMeter tem uma boa integração, pois é possível que o professor crie atividades e envie-as aos alunos. A plataforma QuestMeter, por mais que não possua um tutorial explícito, quando seguido passo a passo de cada etapa conforme a suas necessidades, professor e aluno entendem seu funcionamento, porque a ferramenta torna-se intuitiva. Conforme pesquisas realizadas pelo autor, demonstrou-se que a maioria dos usuários conseguiram realizar suas tarefas sem nenhum auxílio. Com relação a característica de multiplataforma, os aplicativos QuestMeter e Duolingo possuem, sendo o QuestMeter desenvolvido no framework Ionic e na plataforma Firebase.

Quadro 1 - Comparativo dos trabalhos correlatos

Trabalhos Correlatos Características	Duolingo (2011)	QuestMeter (2019)	Toweljs (2018)
transformações geométricas	Não	Não	Sim
motor de jogos	Não	Não	Sim
exportação de atividades	Não	Sim	Não
tutoriais interativos	Sim	Sim	Não
múltiplas plataformas	Sim	Sim	Não

Fonte: elaborado pelo autor.

Nota-se a relevância da utilização de motor de jogos, ter funcionalidades como exportação de atividades e tutoriais interativos para melhor aproveitamento e entendimento das funcionalidades disponíveis da ferramenta. Propõe-se continuar o processo de migração da versão atual, realizar a refatoração de código para melhor manutenção e adicionar novas funcionalidades como novas peças e bloqueios de campos específicos ao exportar um projeto para realização de exercícios. Em função destas características conclui-se que a ferramenta VisEdu-CG possui grande importância no aprendizado da disciplina de Computação Gráfica. Feitas melhorias a cada versão, aperfeiçoando a exibição e entendimento dos alunos a respeito à dos objetos gráficos, manipulação de câmera, todos encapsulados em cenas.

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

A aplicação desenvolvida deve:

- permitir importar/exportar um projeto com a opção de bloqueio de campos (Requisito Funcional - RF);
- disponibilizar guia de “Ajuda” o qual descreve todas as funcionalidades disponíveis da ferramenta (RF);
- desenhar novos componentes dos tipos Iteração, Polígono e Spline (RF);
- ser desenvolvido na linguagem C# (Requisito Não Funcional – RNF);
- utilizar motor de jogos Unity (RNF).

#### 4.3 METODOLOGIA

O trabalho será desenvolvido observando as seguintes etapas:

- levantamento bibliográfico: realizar levantamento bibliográfico sobre refatoração para o motor de jogos Unity, pesquisar sobre tutoriais utilizados no ensino de Computação Gráfica, e sobre os seus conceitos básicos;
- elicitação de requisitos: detalhar e reavaliar os requisitos e, se necessário, especificar outros a partir das necessidades observadas durante a revisão bibliográfica;
- especificação e análise: utilizar as funcionalidades da ferramenta Draw.io, para elaborar diagramas de classe e atividades conforme Unified Modeling Language (UML);
- refatoração de código: realizar processo de refatoração do código implementado na versão atual afim de melhorar compreensão e manutenção da ferramenta;
- implementação da ferramenta: continuar implementação da ferramenta proposta por Buttenberg (2018) migrando para o motor gráfico Unity, disponibilizando funcionalidades como importar/exportar projetos com a opção de campos com seu valor bloqueados. Disponibilizar a guia de Ajuda, onde é descrito as funcionalidades disponíveis da ferramenta. Adicionar novas peças dos

- tipos Iteração, Polígono e Spline;
- f) testes: elaborar testes de usabilidade de aplicação afim de validar a implementação da migração assim como novas funcionalidades, buscando novas melhorias para o desenvolvimento da ferramenta.

Quadro 2 - Cronograma

etapas / quinzenas	2021									
	ago.		set.		out.		nov.		dez.	
	1	2	1	2	1	2	1	2	1	2
levantamento bibliográfico										
elicitação de requisitos										
especificação e análise										
refatoração da versão atual										
implementação da ferramenta										
testes										

Fonte: elaborado pelo autor.

## 5 REVISÃO BIBLIOGRÁFICA

Este capítulo descreve brevemente os assuntos que serão abordados no trabalho: refatoração de sistemas, o motor de jogos Unity e estender o tutorial da ferramenta parcialmente implementado para melhor aprendizado de matéria de Computação Gráfica do curso de Ciência da Computação.

### 5.1 REFATORAÇÃO DE CODIGO

A refatoração é o processo de modificar um sistema de software de modo que não altere o comportamento externo do código, embora melhore a sua estrutura interna. É uma maneira disciplinada de reorganizar o código, minimizando as chances de introduzir *bug*. Em sua essência, ao refatorar, aperfeiçoará o design do código depois que ele foi escrito (FOWLER, 2020).

*Refactorings* são transformações de código que melhoram a manutenibilidade de um sistema, mas sem afetar o seu funcionamento. A definição de transformações de código, está referindo a modificações no código, como dividir uma função em duas, renomear uma variável, mover uma função para outra classe, extrair uma interface de uma classe etc. Segundo melhorar a manutenibilidade do sistema, isto é, melhorar sua modularidade, melhorar seu projeto ou arquitetura, melhorar sua estabilidade, tornar o código mais legível, mais fácil de entender e modificar etc. Por fim, coloca-se uma restrição: não adianta melhorar a manutenibilidade do sistema e prejudicar o seu funcionamento. Ou seja, *refactoring* deve entregar o sistema funcionando exatamente como antes das transformações. Uma outra maneira de dizer isso é afirmando que *refactorings* devem preservar o comportamento ou a semântica do sistema (VALENTE, 2020).

É arriscado ser feito uma refatoração, se exige mudanças que podem introduzir a *bugs* sutis em um código que está funcionando. Senão for feita de forma adequada, pode-se fazer atrasar em dias ou até semanas. Ao começa a explorar o código, logo se descobre novas oportunidades para alterá-lo, à medida que o analisa com mais detalhes. Quanto mais se avalia, mais detalhes se nota a mais mudanças a serem feitas. Em algum momento, é potencial risco de cavar a própria sepultura e não conseguir sair dela. Para evitar que isso aconteça, a refatoração deve ser feita sistematicamente (FOWLER, 2020).

Quando os coautores e Flower escreveram o livro *Padrões de Projeto*, mencionaram que os padrões de projeto oferecem alvos para as refatorações. No entanto, identificar o alvo é somente uma parte do problema e transformar o código para atingir seus objetivos é outro desafio (FOWLER, 2020).

Padrões projetos e refatoração estão amplamente conectados, visando garantir uma melhor compreensão e fácil manutenção de código de um projeto específico. Conforme afirma Rapeli (2006), os padrões de projetos favorecessem a implementações mais eficientes tendo clareza e fácil entendimento do código, em casos de sistemas não projetados em seu uso, é possível aplicá-los sem alterar suas funcionalidades existentes.



## 5.2 TUTORIAIS INTERATIVOS

A acelerada expansão e disseminação das tecnologias digitais permitiu inovar, a nível conceitual e pedagógico, os espaços formais de ensino e aprendizagem e as práticas que dentro deles se organizam. Ao analisar esses espaços formais, nomeadamente as escolas, contém os requisitos e infraestruturas necessárias para reagir às mudanças científicas e tecnológicas vigentes, de modo a se adaptarem às exigências educativas do século XXI, onde a mudança nas práticas de ensino e aprendizagem em sala de aula se torna cada vez mais necessário (PEDRO; BAETA, 2017).

As ferramentas tecnológicas são instrumentos que podem auxiliar no acompanhamento e nas orientações dos alunos, oferecendo soluções como: facilitar o acompanhamento das atividades em execução por cada aluno de maneira síncrona e/ou assíncrona; facilitar a difusão de esclarecimento de dúvidas coletivas e individuais; integrar espaços coletivos e individuais de produção; e oferecer auxílios a cada aluno conforme o estágio de aprendizagem (SANTOS, 2014).

Tutoriais interativos podem ser implementados de forma de assistentes visuais, na qual contribuem para um melhor aproveitamento dos recursos e entendimento a respeito da ferramenta a ser utilizada. Essas novas tecnologias são de grande ajuda no processo de ensino aprendizagem, fazendo com que os usuários se sintam mais próximos e motivados para aproveitar todos os recursos disponíveis.

A instrução através do tutorial visa potencializar o processo de construção do conhecimento. Para tanto, o aprendiz terá que ler e exercitar o conhecimento adquirido. A conexão com outros *sites* disponíveis na internet, pode apresentar informações auxiliares que nortearão no processo de ensino-aprendizagem (SANTOS; BEZERRA, 2018).

Conforme afirma Santos e Bezerra (2018), na internet há uma predominância de tutoriais informais, uma vez que se pode encontrar em *sites* como o Youtube, com diversos vídeos de tutoriais ensinando sobre os mais variados assuntos, feitos por quaisquer pessoas independentemente de ter um conhecimento científico. Todavia, a qualidade vai depender da sua capacidade de expressão e conhecimento das características dessa ferramenta. Portanto tutoriais podem ser considerados mecanismos informais de auxílio aprendizagem, destacando seu caráter didático descrevendo por uma sequência de passos a utilização de uma ferramenta.

## REFERÊNCIAS

- ARAÚJO, Luciana Pereira de. **AduboGL - Aplicação didática usando a biblioteca OpenGL**. 2012. 76 f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) – Centro de Ciências Exatas e Naturais, Fundação Universidade Regional de Blumenau, Blumenau.
- BUTTENBERG, Peterson Boni. **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, Fundação Universidade Regional de Blumenau, Blumenau.
- FOWLER, Martin. **Refatoração: Aperfeiçoando o design de códigos existentes**. Novatec Editora, 2020.
- KOEHLER, William Fernandes. **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, Fundação Universidade Regional de Blumenau, Blumenau.
- MELO, Telma de Macedo. O emprego do aplicativo móvel Duolingo no processo de ensino e aprendizagem de alemão como língua estrangeira em contexto de ensino presencial universitário. **Pandaemonium Germanicum**, São Paulo, v. 24, no. 42, p. 78-107, jan./abr. 2021.
- PEDRO, Neuza; BAETA, Patrícia. Práticas educativas nas salas de aula do futuro: Análise focalizada nas metodologias de ensino aprendizagem. In: **Actas da X Conferência Internacional de TIC na Educação - Challenges 2017**. 2017.
- RAPELI, Leide Rachel Chieusi. **Refatoração de sistema Java utilizando padrões de projeto: um estudo de caso**. 2006. 130 f. Dissertação de mestrado (Pós-Graduação em Ciência da Computação) – Centro de Ciências Exatas e de Tecnologia, Universidade Federal de São Carlos, São Carlos.
- REIS, Dalton Solano. **Grupo de Pesquisa em Computação Gráfica, Processamento de Imagens e Entretenimento Digital**. [S.l.], [201- ], Disponível em: <<http://gcg.inf.furb.br/?pageid=1147>>. Acesso em: 13 abr. 2021.

SILVA, Adriana Santos da. **A tecnologia como nova prática pedagógica**. 2011. Monografia apresentada ao curso de pós-graduação em Supervisão escolar, Vila Velha.

VIEIRA, Pâmela Carolina. **QuestMeter: Ferramenta de quiz com conceitos de clickers e gamificação**. 2019. 20 f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) – Centro de Ciências Exatas e Naturais, Fundação Universidade Regional de Blumenau, Blumenau.

ZANLUCA, Gabriel. **Toweljs: Engine 3D em JavaScript usando arquitetura baseada em componentes**. 2018. 80 f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) – Centro de Ciências Exatas e Naturais, Fundação Universidade Regional de Blumenau, Blumenau.

### **ASSINATURAS**

(Atenção: todas as folhas devem estar rubricadas)

Assinatura do(a) Aluno(a): \_\_\_\_\_

Assinatura do(a) Orientador(a): \_\_\_\_\_

Assinatura do(a) Coorientador(a) (se houver): \_\_\_\_\_

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

## FORMULÁRIO DE AVALIAÇÃO – PROFESSOR TCC I

Acadêmico(a): \_\_\_\_\_

Avaliador(a): \_\_\_\_\_

ASPECTOS AVALIADOS <sup>1</sup>		atende	atende parcialmente	não atende
ASPECTOS TÉCNICOS	1. INTRODUÇÃO O tema de pesquisa está devidamente contextualizado/delimitado?			
	O problema está claramente formulado?			
	2. OBJETIVOS O objetivo principal está claramente definido e é passível de ser alcançado?			
	Os objetivos específicos são coerentes com o objetivo principal?			
	3. JUSTIFICATIVA São apresentados argumentos científicos, técnicos ou metodológicos que justificam a proposta?			
	São apresentadas as contribuições teóricas, práticas ou sociais que justificam a proposta?			
	4. METODOLOGIA Foram relacionadas todas as etapas necessárias para o desenvolvimento do TCC?			
	Os métodos, recursos e o cronograma estão devidamente apresentados?			
ASPECTOS METODOLÓGICOS	5. REVISÃO BIBLIOGRÁFICA (atenção para a diferença de conteúdo entre projeto e pré-projeto) Os assuntos apresentados são suficientes e têm relação com o tema do TCC?			
	6. LINGUAGEM USADA (redação) O texto completo é coerente e redigido corretamente em língua portuguesa, usando linguagem formal/científica?			
	A exposição do assunto é ordenada (as ideias estão bem encadeadas e a linguagem utilizada é clara)?			
	7. ORGANIZAÇÃO E APRESENTAÇÃO GRÁFICA DO TEXTO A organização e apresentação dos capítulos, seções, subseções e parágrafos estão de acordo com o modelo estabelecido?			
	8. ILUSTRAÇÕES (figuras, quadros, tabelas) As ilustrações são legíveis e obedecem às normas da ABNT?			
	9. REFERÊNCIAS E CITAÇÕES As referências obedecem às normas da ABNT?			
	As citações obedecem às normas da ABNT?			
	Todos os documentos citados foram referenciados e vice-versa, isto é, as citações e referências são consistentes?			

### PARECER – PROFESSOR DE TCC I OU COORDENADOR DE TCC (PREENCHER APENAS NO PROJETO):

O projeto de TCC será reprovado se:

- qualquer um dos itens tiver resposta NÃO ATENDE;
- pelo menos 4 (quatro) itens dos **ASPECTOS TÉCNICOS** tiverem resposta ATENDE PARCIALMENTE; ou
- pelo menos 4 (quatro) itens dos **ASPECTOS METODOLÓGICOS** tiverem resposta ATENDE PARCIALMENTE.

**PARECER:** (    ) APROVADO (    ) REPROVADO

Assinatura: \_\_\_\_\_ Data: \_\_\_\_\_

<sup>1</sup> Quando o avaliador marcar algum item como atende parcialmente ou não atende, deve obrigatoriamente indicar os motivos no texto, para que o aluno saiba o porquê da avaliação.

## FORMULÁRIO DE AVALIAÇÃO – PROFESSOR AVALIADOR

Acadêmico(a): \_\_\_\_\_

Avaliador(a): \_\_\_\_\_

ASPECTOS AVALIADOS <sup>1</sup>		atende	atende parcialmente	não atende
ASPECTOS TÉCNICOS	1. INTRODUÇÃO O tema de pesquisa está devidamente contextualizado/delimitado?			
	O problema está claramente formulado?			
	1. OBJETIVOS O objetivo principal está claramente definido e é passível de ser alcançado?			
	Os objetivos específicos são coerentes com o objetivo principal?			
	2. TRABALHOS CORRELATOS São apresentados trabalhos correlatos, bem como descritas as principais funcionalidades e os pontos fortes e fracos?			
	3. JUSTIFICATIVA Foi apresentado e discutido um quadro relacionando os trabalhos correlatos e suas principais funcionalidades com a proposta apresentada?			
	São apresentados argumentos científicos, técnicos ou metodológicos que justificam a proposta?			
	São apresentadas as contribuições teóricas, práticas ou sociais que justificam a proposta?			
	4. REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO Os requisitos funcionais e não funcionais foram claramente descritos?			
	5. METODOLOGIA Foram relacionadas todas as etapas necessárias para o desenvolvimento do TCC?			
	Os métodos, recursos e o cronograma estão devidamente apresentados e são compatíveis com a metodologia proposta?			
	6. REVISÃO BIBLIOGRÁFICA (atenção para a diferença de conteúdo entre projeto e pré-projeto) Os assuntos apresentados são suficientes e têm relação com o tema do TCC?			
ASPECTOS METODOLÓGICOS	As referências contemplam adequadamente os assuntos abordados (são indicadas obras atualizadas e as mais importantes da área)?			
	7. LINGUAGEM USADA (redação) O texto completo é coerente e redigido corretamente em língua portuguesa, usando linguagem formal/científica?			
	A exposição do assunto é ordenada (as ideias estão bem encadeadas e a linguagem utilizada é clara)?			

### PARECER – PROFESSOR AVALIADOR: (PREENCHER APENAS NO PROJETO)

O projeto de TCC ser deverá ser revisado, isto é, necessita de complementação, se:

- qualquer um dos itens tiver resposta NÃO ATENDE;
- pelo menos **5 (cinco)** tiverem resposta ATENDE PARCIALMENTE.

**PARECER:** (     ) APROVADO (     ) REPROVADO

Assinatura: \_\_\_\_\_ Data: \_\_\_\_\_

<sup>1</sup> Quando o avaliador marcar algum item como atende parcialmente ou não atende, deve obrigatoriamente indicar os motivos no texto, para que o aluno saiba o porquê da avaliação.