

**INSTITUTO FEDERAL DE MATO GROSSO DO SUL
CAMPUS NOVA ANDRADINA**

**EDUCAÇÃO PROFISSIONAL TÉCNICA DE NÍVEL MÉDIO
INTEGRADO EM INFORMÁTICA**

BEATRIZ LANDIGRAF RIOS DOS SANTOS

**LABORATÓRIO CURIE: JOGO PARA O ENSINO DE QUÍMICA
EXPERIMENTAL E BOAS PRÁTICAS NO LABORATÓRIO**

**NOVA ANDRADINA – MS
2018**

BEATRIZ LANDIGRAF RIOS DOS SANTOS

LABORATÓRIO CURIE: JOGO PARA O ENSINO DE QUÍMICA
EXPERIMENTAL E BOAS PRÁTICAS NO LABORATÓRIO

Proposta de Trabalho de Conclusão de Curso
apresentada no curso de Educação Profissional
Técnica de Nível Médio Integrado em Informática
do Instituto Federal de Mato Grosso do Sul – IFMS
– como um dos requisitos para a obtenção do
título de Técnico em Informática.

Orientador: Me. Leonardo Bravo Estácio.
Coorientador: Me. Narcimário Pereira Coelho.

NOVA ANDRADINA – MS
2018

AGRADECIMENTOS

Agradeço a Deus, por ter sido meu pai, meu sustento, meu amigo e companheiro e principalmente por ter colocado pessoas tão especiais e maravilhosas na minha vida cujo merecem serem agradecidas nos próximos parágrafos.

Ao Instituto Federal de Mato Grosso do Sul, pela oferta desse curso, assim como seu corpo docente, em especial, ao meu orientador, Me. Leonardo Bravo Estácio, por ter sido o pilar desse trabalho, junto com o co-orientador, Me. Narcimário Coelho, a eles agradeço imensamente.

Agradeço a minha família, que sempre me apoiou, me aconselhou, mesmo estando grande parte do dia fora, em especial a minha mãe Abigayld Landigraf e meu irmão Carlos Henrique, vocês cumpriram o verdadeiro significado de família em minha vida.

Um destaque especial a professora Valdineia Garcia que sempre nos pressionava para ter novidades a toda semana e também ao professor convidado para a banca, Fernando Balbino, obrigada por todas as considerações.

Aquelas que foram mais do que colegas de sala, que sempre me ajudaram, com palavras, abraços, casas com moradia e alimentação, as minhas amigas, Ana Clara Oliveira, Caroline Cristina, Danielli Siqueira, Gabrielle Rios e Livia Lamar, minha eterna gratidão.

A minha amiga e assistente de alunos, Fernanda Soares Junglos, por todo ensinamento, conselho e apoio, você que se tornou como uma irmã para mim durante toda essa trajetória, muito obrigada.

E a todos aqueles que em algum momento, mesmo sem intenção me aconselharam, me motivaram, saibam que vocês contribuíram diretamente para a conclusão deste trabalho, muito obrigada.

RESUMO

Muitos dos estudantes do ensino médio têm enfrentado dificuldades no processo de aprendizagem da disciplina de química inorgânica, principalmente quando há falta de objetos didáticos que possam ilustrar e/ou exemplificar o conteúdo abordado pelo professor durante as aulas. Um jogo se enquadra como um objeto de aprendizagem, pois através dele é possível desenvolver diversas habilidades na vida do usuário, como agilidade, efeito motivador, habilidades cognitivas, aprendizado por descoberta, experiências de novas identidades, coordenação motora, além da capacidade de memorização. Desta forma, este trabalho tem o objetivo de apresentar o desenvolvimento de um jogo didático para o ensino de química e boas práticas no laboratório. Para tanto foi desenvolvido um jogo de três dimensões, onde os jogadores podem manipular os objetos e visualiza-los como na vida real. O desenvolvimento foi na *game engine* Unity3D. E as modelagens tridimensionais construídas no software 3ds Max. Para a construção de animações baseadas em dados de *motion capture* foi utilizado o Motion Builder e como plataforma unificada de recursos utilizamos inicialmente o armazenamento de dados e autenticação de usuários através do Firebase. O jogo construído pode ser considerado uma ferramenta que incentivará estudantes a conhecer o processo de reações em um laboratório, e também do uso adequado de cada equipamento.

Palavras-chaves: Jogo eletrônico no ensino, Química, Unity 3D.

SUMÁRIO

1	INTRODUÇÃO	7
2	OBJETIVOS	11
2.1	Objetivo Geral	11
2.2	Objetivos Específicos	11
3	FUNDAMENTAÇÃO TEÓRICA.....	12
3.1	Engenharia de Software para Jogos	12
1.1	3DS Max	17
1.2	MotionBuilder	18
1.3	Substance Painter	18
1.4	Unity 3D	19
1.5	Firebase	20
4	METODOLOGIA	22
4.1	Passo 1 – Levantamento bibliográfico.....	23
4.2	Passo 2 – Estudo das ferramentas e métodos.....	23
4.3	Passo 3 – Formulação das ideias	24
4.4	Passo 4 – Concepção inicial do <i>Game Design Document</i> - GDD	24
4.5	Passo 5 – Definição inicial da arquitetura	27
4.6	Passo 6 – Levantamento de requisitos	27
4.7	Passo 7 – Implementação de requisitos seleccionados.....	30
4.7.1	Concepção dos modelos 3D.....	30
4.7.2	Processo de Desenvolvimento na <i>Game Engine</i>	34
4.8	Passo 8 – Análise e realização de testes.....	35
1.6	Passo 9 – Revisão do GDD	36
1.7	Passo 10 – Refinamento da arquitetura	36
5	DESENVOLVIMENTO	37
5.1	Definição inicial da arquitetura	37

5.2	Levantamento de requisitos	38
5.3	Desenvolvimento de requisitos	39
5.4	Requisitos concluídos	40
5.5	Requisitos pendentes.....	45
5.6	Implementação da Autenticação	46
6	RESULTADO FINAL	47
6.1	Game Design Document – GDD	47
6.2	Personagem.....	47
6.3	Gameplay.....	49
6.4	Controles.....	49
6.5	Câmera	50
6.6	Universo do jogo	51
6.7	Interface	64
7	CONSIDERAÇÕES FINAIS	70
8	REFERÊNCIAS.....	Erro! Indicador não definido.

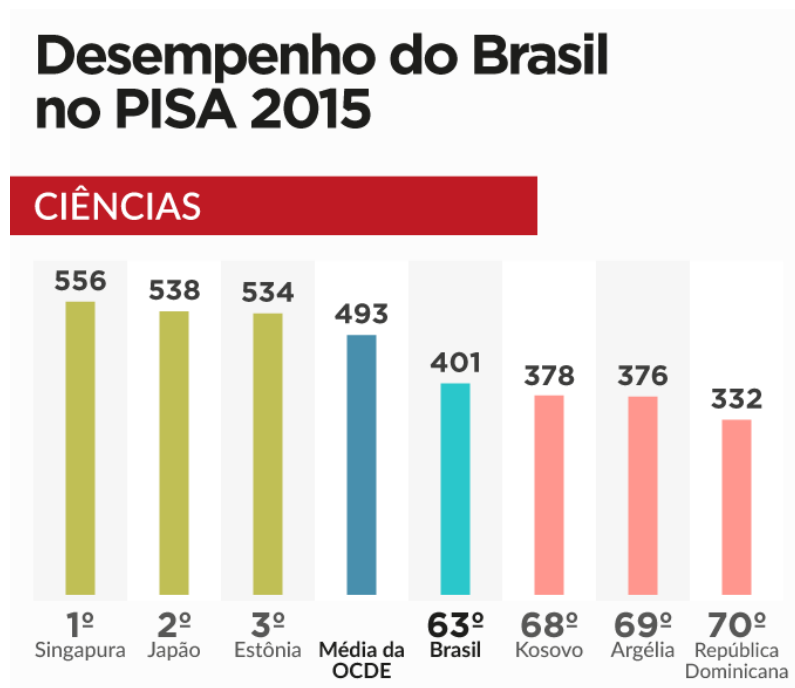
1 INTRODUÇÃO

Culturalmente, no Brasil, os conteúdos de ciências exatas são apresentados aos alunos como um discurso simbólico, abstrato e confuso. O ensino de ciências exatas tem se ocupado em garantir que os alunos dominem apenas técnicas e fórmulas, ao invés de desenvolverem também a compreensão acerca dos conteúdos.

Na Lei de Diretrizes e Bases (LDB) da educação nacional, há as finalidades sobre o ensino médio, e dentre elas, há uma que diz sobre a necessidade de existir uma relação entre a teoria e a prática no ensino de cada disciplina.

Segundo os resultados do PISA (Programa Internacional de Avaliação de Estudantes) do ano de 2015, o Brasil esteve entre os 8 piores países com uma média de 401 na área de ciências, ocupando a 63ª posição como apresentado na Figura 1.

Figura 1 - Desempenho do Brasil no PISA 2015

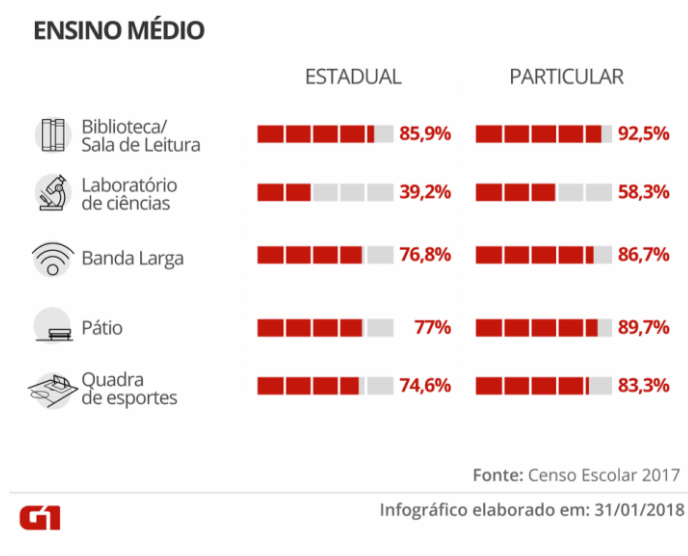


FONTE: REVISTA EXAME

Um dos fatores que leva o mal desempenho do Brasil no PISA se dá na não ocorrência da relação da teoria com a prática nas escolas brasileiras, principalmente nas disciplinas da área de ciências da natureza em que se é necessário um laboratório para que possam ser realizadas as aulas experimentais, minando os interesses dos estudantes.

Contudo, como apresentado na Figura 2, o infográfico retirado do site do G1, elaborado com os dados do censo escolar de 2017, das estruturas destinadas às escolas, o laboratório de ciências é o que apresenta maior escassez, com apenas 39,2% nas escolas estaduais e 58,3% nas escolas particulares.

Figura 1 - Estruturas das escolas brasileiras que contêm ensino médio



FONTE: SITE DO G1

Segundo Tarouco et. al (2014):

“Objeto de Aprendizagem (OA) apresenta-se como uma vantajosa ferramenta de aprendizagem e instrução, a qual pode ser utilizada para o ensino de diversos conteúdos e revisão de conceitos. A metodologia com a qual o OA é utilizado será um dos fatores-chave a determinar se a sua adoção pode ou não levar o aluno ao desenvolvimento do pensamento crítico. Flexibilidade e possibilidade de reutilização são algumas das características de um Objeto de Aprendizagem, que facilitam a disseminação do conhecimento, assim como sua

atualização. Salienta-se que, como em qualquer planejamento de aula, a adequada seleção de um OA para uso em atividade didática fica definida a partir do objetivo que se pretende alcançar na aprendizagem de um determinado conteúdo. Contemplando esse quesito, o Objeto de Aprendizagem pode ser um excelente aliado do professor em sala de aula”.

No modelo de nossa sociedade atual é comum ver o crescimento de OA, dentre eles estão os jogos eletrônicos educativos, graças ao acesso cada vez maior de tecnologias como *smartphones* e computadores pessoais. Lecheta (2016) diz que nos dias atuais ninguém consegue ficar longe de um celular, e diz que o *mobile* é um grande pilar na área de tecnologia, sendo a área que mais crescerá nos próximos anos.

Os jogos eletrônicos, inicialmente, eram vistos apenas como forma de entretenimento. Contudo, atualmente estão cada vez mais presentes na educação, desde o ensino básico até o superior, sendo gradativamente mais explorados nesse aspecto educacional, visto que os jogos cooperam para o desenvolvimento de diversos fatores, sejam objetivos específicos, habilidades ou valores (FALKEMBACH, [s.d]).

Savi et al. (2008) dizem que os benefícios dos jogos são classificados em: efeito motivador; facilitador do aprendizado; desenvolvimento de habilidades cognitivas; aprendizado por descoberta; experiências de novas identidades; socialização; coordenação motora; comportamento *expert*. Silveira et al (2012) relatam que além do potencial de prender a atenção do estudante, os jogos fazem com que eles apresentem maior interesse na disciplina, além de ser uma maneira prazerosa de aprender.

Através de ambientes virtuais 3D, os jogadores podem presenciar as cenas de maneira virtual, possibilitando a utilização de materiais como se fosse de forma presencial (MARÇAL, et al., 2009).

Diante do exposto, foi desenvolvido um jogo eletrônico 3D que simula um laboratório tendo como personagem principal Marie Currie. Com isso, pretende-se auxiliar o aluno e o professor quando há falta de laboratório de química, visto que através de um ambiente 3D o aluno poderá manipular objetos, representando a maneira presencial. Este trabalho explica o processo de

desenvolvimento deste jogo, e, traz um GDD – *Game Design Document*, documento que possui todos os aspectos do jogo.

A estrutura se dispõe em 6 capítulos: o capítulo 2 apresenta os objetivos deste trabalho; no capítulo 3 é apresentada a fundamentação teórica referente à engenharia de *software* voltada para jogos, o *software* de modelagem 3DS Max, o de animação *MotionBuilder*, o de texturização *Substance Painter*, o motor de jogos *Unity 3D* e o *Firebase*; o capítulo 4 traz o percurso metodológico atravessado para a materialização deste projeto; o capítulo 5 traz o desenvolvimento adotado; no capítulo 6 é apresentado o resultado; e por fim, no capítulo 7, as considerações finais.

2 OBJETIVOS

2.1 Objetivo Geral

Desenvolver um jogo eletrônico 3D educativo, para dispositivos móveis que utilizem os sistemas operacionais Android e IOS, que auxilie no ensino de química e boas práticas no laboratório.

2.2 Objetivos Específicos

- realizar levantamento bibliográfico;
- elaborar o *Game Design Document*;
- construir a modelagem 3D de cada objeto do jogo;
- desenvolver uma fase que trabalhe o uso de equipamento de segurança utilizando uma *game engine*;
- realizar testes.

3 FUNDAMENTAÇÃO TEÓRICA

3.1 Engenharia de Software para Jogos

A engenharia de um jogo começa pela concepção de um desafio central. A partir desse, são realizadas seções de *brainstorming* – são realizados debates em buscas de ideias que possam ser a possível solução. Os *brainstorming* geram diversas ideias, e a partir daí vem o primeiro obstáculo, escolher as que serão desenvolvidas.

Segundo Schell (2014) várias equipes tentam estruturar as “ideias perfeitas”, as quais podem nunca aparecer. Visando evitar contratempo, Schell (2014) fala sobre a necessidade de um teste de ideias, em que é necessário passar as ideias por oito filtros, e assim, quando uma ideia falhar em algum desses filtros, deverá ser reavaliada e readequada.

1º filtro - Impulso artístico: “Este jogo é agradável? ” Este é um filtro pessoal, sendo resumido apenas a pergunta chave, caso a resposta seja não, algo necessita ser modificado.

2º filtro – Demografia: “O jogo atende o público alvo?” Jogos costumam ter um público-alvo, podendo ser para uma faixa etária, determinado sexo, ou algum grupo de pessoas específico. Nesta etapa, deve considerar se o *design* é ideal para o grupo demográfico que será direcionado.

3º filtro - *Design* de experiência: “O jogo é bem projetado? ” Leve em consideração todas as experiências da equipe de criação, incluindo estética, curvas de interesse, tipo do jogo e equilíbrio no *gameplay*.

4º filtro – Inovação: “O jogo apresenta originalidade? ” Mesmo o jogo sendo algo inédito, precisa haver uma novidade, algo que os jogadores não viram antes.

5º filtro - Negócios e marketing: “Este jogo será lucrativo? ” Jogos possuem valor e *game designers* que desejam a venda de seus jogos, devem

considerar as seguintes questões: o tema e a história são atraentes para os consumidores?; o jogo é facilmente explicável e pode ser entendido apenas analisando uma visão geral?; quais são as expectativas que os consumidores terão sobre esse jogo com base em sua classificação?; como os recursos deste jogo se comparam a outros jogos similares no mercado?; será que o custo de produzir este jogo será tão alto que o tornará não lucrativo?; o modelo de negócio para o jogo faz sentido? As respostas a essas questões terão um impacto no projeto pois as ideias que impulsionaram o projeto inicial podem revelar-se completamente insustentáveis quando visualizada através deste teste.

6º filtro - Limite tecnológico: "É tecnicamente possível construir este jogo? " Antes do desenvolvimento, as ideias do jogo não são concretas, assim, não estão vinculadas pelas restrições do que é possível ou prático. Esse filtro pode conduzir a direção de um jogo, visto que no processo de aplicação é previsível perceber que os recursos para que o jogo funcione corretamente estão ou não estão disponíveis. Caso aparece outras ideias durante a aplicação deste filtro, seriam particularmente valiosas, já que há certeza de que elas são práticas.

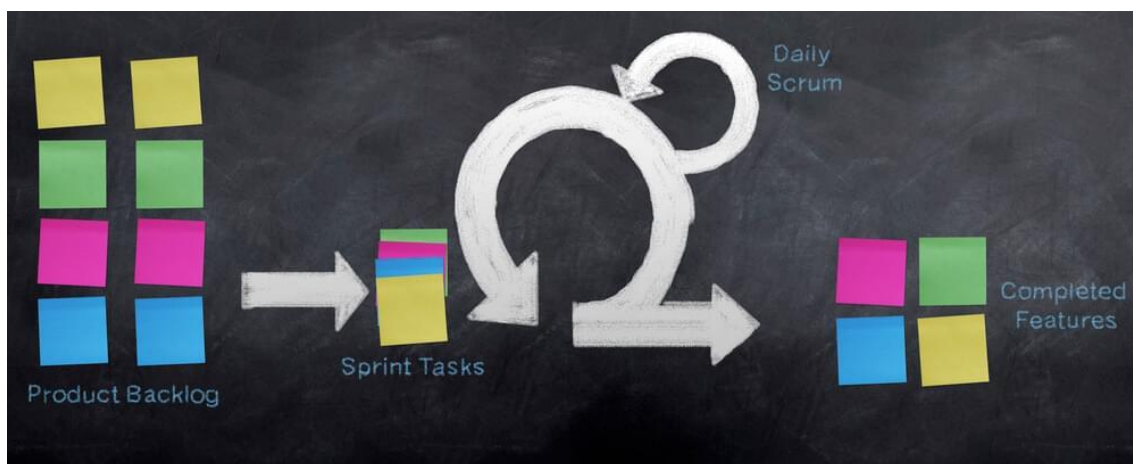
7º filtro - Social / Comunidade: "Este jogo atende aos objetivos sociais e comunitários?" Às vezes, não é suficiente para um jogo ser divertido. Alguns dos objetivos de design podem exigir um componente social forte, um componente viral forte ou a formação de uma comunidade próspera em torno de seu jogo. O design do seu jogo terá um forte impacto sobre essas coisas.

8º filtro – Testes: "Os jogadores aproveitam o jogo? " Uma vez que seja possível realizar testes, é necessário aplicar o mais importante de todos os filtros, afinal, uma coisa é imaginar o que será um jogo, e totalmente outra é jogar, e o mais importante é ser jogado pelo público-alvo. Há o desejo do jogo ser jogável o mais rápido possível, pois quando o jogo está em ação, as mudanças importantes tornar-se-ão óbvias. Além de modificar o próprio jogo, a aplicação deste filtro geralmente muda e sintoniza os outros à medida em que começa a aprender mais sobre a mecânica em desenvolvimento e a psicologia do público-alvo.

Esse processo de selecionar as ideias para compor um jogo eletrônico é chamado de *Game Design*. As decisões finais da direção de um jogo, geralmente, é feita por apenas uma pessoa, mas nada impede de ser feita pela equipe. O processo de *Game Design* é todo sobre tomadas de decisões – nele, é preciso tomar as melhores decisões possíveis, o mais rápido possível, com intuito de ter o melhor resultado final.

Schell (2014) diz que existe uma peculiaridade da natureza humana - mudanças, talvez seja algo que já tenha passado pelos oito filtros, como o público alvo, ou questões de design. Pensando nisso, é comum no desenvolvimento de jogos a adoção de *looping* utilizando a metodologia ágil *Scrum* para o desenvolvimento do jogo. Essa metodologia é representada na Figura 3 em que a cada *looping* é necessária a revisão dos oito filtros e ainda estar sempre atendendo os seguintes métodos e processos ágeis.

Figura 3 - Metodologia Scrum



FONTE: SCHELL, 2014

Objetivos flexíveis: noção de que não é possível saber exatamente o que terá tempo para criar, por isso é necessário planejar um conjunto mais flexível de objetivos e não apenas tolerar mudanças no plano e sim planejar mudanças no plano.

Backlog priorizado: *backlog* é uma lista de recursos ordenados por prioridade. Sempre que alguém tiver uma nova ideia para um recurso, ele pode ser adicionado ao *backlog*. Em cada *sprint*, a equipe revisita o *backlog* e ordena

os recursos por prioridade. Isso facilita a decisão sobre o que trabalhar no próximo - basta ver o topo do *backlog*. Não há garantia de que tudo no *backlog* seja feito - só garante que as coisas mais importantes sejam feitas com qualquer tempo disponível.

Sprints: em vez de se concentrar em direção de um objetivo de longo prazo, os desenvolvedores ágeis trabalham em uma série de "*sprints*", onde cada *sprint* tem algumas semanas e tem um trabalho concreto no final. Os prazos têm uma maneira especial de fazer as coisas acontecerem, e essa é precisamente a filosofia por trás do *sprint*: mais prazos significa que mais coisas são feitas.

Reuniões de *Scrum*: encontros diários de "*scrum*", projetados para brevidade e eficácia, são reuniões tipicamente rápidas. Durante essas reuniões, cada membro explica exatamente três coisas: o que eles realizaram ontem, o que eles planejam realizar hoje e quais os problemas que enfrentam. As soluções para os problemas são discutidas individualmente pelos membros da equipe apropriada após a conclusão da reunião. Neste sistema, cada membro da equipe fica ciente do que os outros estão fazendo e dá-lhes a chance de obter ajuda dos membros.

Dia de demonstração: no final de cada *sprint*, são realizadas reuniões para ver e experimentar o que foi produzido. Com base nessa nova linha de base, a equipe faz análise de risco e trabalha em conjunto para planejar o próximo *sprint*.

Retrospectivas: Também no final de cada *sprint*, a equipe tem uma reunião de retrospectiva, visando não o produto em que estão trabalhando, mas sim sobre o processo que eles estão usando. Esta é uma chance para a equipe discutir o que estão fazendo corretamente, o que estão fazendo de errado e como devem ajustar seu processo para o próximo *sprint*.

Tendo as ideias que compõem do *Game Design* bem estabelecidas, pode-se avançar para o próximo passo, o projeto básico da arquitetura. Essa parte é muito importante principalmente quando o jogo em questão não for

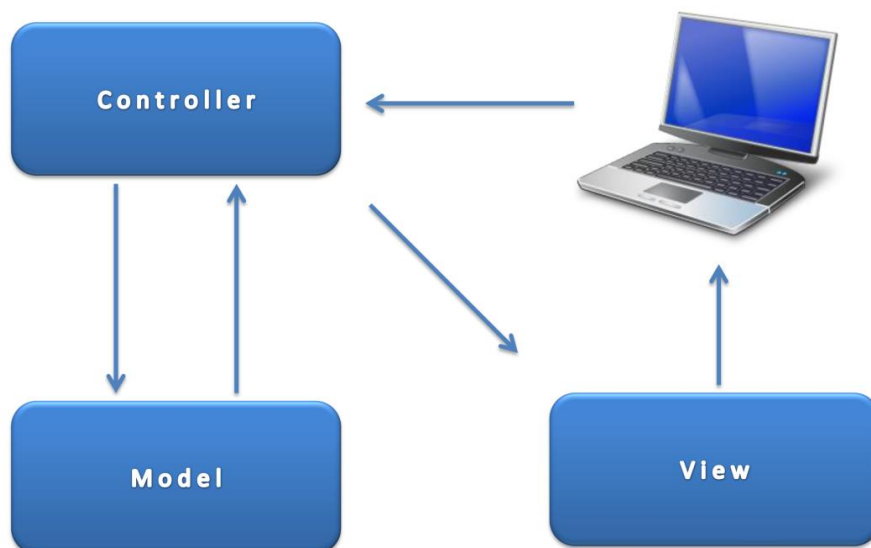
simples de ser implementado. Entretanto, se o tempo de desenvolvimento e testes for considerado bastante curto, a etapa de projeto de arquitetura pode não ser desconsiderada.

Essa arquitetura básica deve ser um esboço de como as partes que compõem um jogo se comunicarão. Como dito, é só uma base, uma vez que o processo de desenvolvimento de jogos deve ser iterativo (Schell, 2014). É impossível inicialmente planejar com precisão a arquitetura e quantas interações realmente serão necessárias antes do jogo ser considerado “bom o suficiente” e isso torna o desenvolvimento um segmento bastante arriscado.

Model-view-controller (MVC) é um padrão de arquitetura de software que separa a representação da informação da interação do usuário, dividindo uma aplicação em três partes interconectadas.

Quando se utiliza o MVC em jogos eletrônicos, divide o software em: *Model* (dados, modelos 3D, *sprites*, etc.), *View* (como o usuário enxerga o jogo) e *Controller* (ações e decisões baseadas em acontecimentos). Essa divisão pode ser observada na Figura 4.

Figura 4- Arquitetura MVC



FONTE: O AUTOR, 2018

O desenvolvimento do jogo e da interface do usuário tem um fluxo de trabalho que normalmente aguarda a entrada do usuário ou outra condição de acionamento (inteligência artificial de algum inimigo, ação pré-programada, etc.), enviando a notificação desses eventos em algum lugar apropriado, decidindo o que fazer em resposta e atualizando os dados adequadamente. Essas ações mostram claramente a compatibilidade desses aplicativos com o MVC.

Ao dividir o processo em dados, interfaces e decisões, os desenvolvedores podem organizar muito melhor a arquitetura de um jogo eletrônico e permitir assim um crescimento sustentável do código.

Todas as decisões tomadas nessa etapa de *Game Design* constituem o GDD – *Game Design Document*. Um documento que possui todos os aspectos do jogo em desenvolvimento.

1.1 3DS Max

O 3DS Max, originalmente chamado por 3D Studio Max, foi desenvolvido pela Autodesk e assim como os outros softwares de modelagem tridimensional, foi desenvolvido com o intuito de auxiliar modeladores que iniciavam todo projeto do zero, tendo que dominar diversos conceitos matemáticos para a criação de uma malha e a partir dela, sua manipulação, o que delimitava a expansão do mundo 3D (GUIMARÃES; PARENTE, 2008).

Por meio desses softwares, a modelagem passou a ser por meio da modificação dos polígonos, arestas e vértices, tendo como ponto inicial objetos primitivos, como cubo, esfera ou cilindro e neles são feitas as alterações e aplicações de modificadores disponíveis até a obtenção do modelo desejado.

Polycount é o número de polígonos de um objeto, sendo *highpoly* e *lowpoly* quando possui muitos e poucos polígonos, respectivamente. Quanto mais polígonos um objeto possuir, mais suave serão suas fases, ao contrário, terão aparências robustas (LARSSON et al., 2013).

Jogos desenvolvidos para dispositivos móveis, devem exigir pouco custo computacional, por isso, vários autores, assim como Marçal et al., (2009) empregaram técnicas de *lowpoly*, com o intuito de diminuir o esforço computacional ao mesmo tempo mantendo o maior detalhamento do objeto possível.

1.2 MotionBuilder

O MotionBuilder é um software desenvolvido pela Autodesk destinado à criação de animações em tempo real. Ele será utilizado para a animação da personagem Marie Curie, que foi modelada no 3DS Max e as texturas criadas no Substance Painter.

Em jogos eletrônicos que utilizam arte 3D as animações podem ser feitas utilizando duas abordagens. A primeira é dada através de sequências de deformações realizadas nas malhas do modelo, realizadas de forma manual, quando a segunda, consiste na utilização de *bones* (ossos) para auxiliar na deformação e manipulação dos polígonos.

1.3 Substance Painter

O Substance Painter é um software desenvolvido pela Allegorithmic destinado a texturização 3D e 2D. Ele possui um vasto conteúdo de texturas, o que facilita no processo de criação, assim como todo o conjunto de ferramentas e filtros, como ferramentas para corte e ferramenta de equalizador de cores (ALLEGORITHMIC, 2017).

1.4 Unity 3D

A Unity 3D é uma *game engine*, para o desenvolvimento de aplicações de duas ou três dimensões, desenvolvido pela Unity Technologie. Como uma boa *game engine*, ela conta com o oferecimento de diversos módulos que garantem a efetividade e agilidade do processo de desenvolvimento. Na Tabela 1 estão expostos os módulos e a descrição de cada um.

Tabela 1: Módulos disponíveis no motor de jogos Unity 3D

Módulo	Descrição
Gráfico	Oferece um motor de renderização customizado, <i>DirectX</i> ou <i>OpenGL</i> , dependendo da plataforma selecionada. Também são oferecidos quarenta tipos de efeitos <i>shader</i> , bem como a opção da criação de novos efeitos.
Física e Detecção de colisão	Implementa propriedades da física como massa, gravidade, inércia e colisão por meio do motor de física <i>PhysX</i> da NVIDIA 6.
Entrada/Saída	Não somente oferece suporte a dispositivos convencionais usados em jogos (teclado, <i>mouse</i> , <i>joystick</i> e <i>gamepad</i>), mas também a telas sensíveis ao toque, acelerômetro e posição geográfica de dispositivos móveis.
Som	Utiliza o do motor de som FMOD 7, oferecendo funcionalidades como modelos customizados de atenuação por distância, filtros básicos e efeitos <i>Doppler</i> .
Inteligência Artificial	Fornece o sistema 'NavMesh', utilizado na implementação da movimentação dos Non Playing Characters (NPC) dentro do ambiente virtual.
Rede	Implementa o modelo de comunicação em rede Cliente-Servidor, onde Clientes Unity conectados a um Servidor Unity podem se comunicar de acordo com a dinâmica do jogo.

FONTE: POPOLIN NETO ET AL., 2015

O site oficial do Unity3D¹ conta com materiais que explicam todo o funcionamento do software – desde manuais que auxiliam na instalação a tutoriais. As informações contidas nessa seção foram retiradas desse site.

Game objects, são os objetos presente na cena. Imagine a cena como uma fase de um jogo, nessa fase devem ser adicionados diversos itens, como os obstáculos, personagens, mobília, câmera do jogo entre outros. Esses itens adicionados são os *game objects*.

Com o Unity é possível desenvolver jogos sem anos de experiência na criação de jogos, isso graças aos módulos já apresentados na Tabela 1 e também ao conceito de arquitetura baseada em *game assets* (componentes).

Cada objeto tem ações diferentes, podem emitir som, andar, rotacionar, isso dependerá do componente adicionado, por exemplo, o componente *transform* é responsável pelas transformações desse objeto, para isso ele conta com os atributos - posição, rotação e escala do objeto e cada modificação nesse item implicará diretamente no objeto.

Além dos componentes padrões, é possível adicionar outros, como o *script*. Os scripts são usados para a criação de efeitos gráficos, controle de comportamento físico de objetos ou até mesmo implementar uma função personalizada.

1.5 Firebase

O Firebase, após a integração ao Google, se tornou uma grande plataforma de construção de aplicativos mobile e web por entre recursos que favorecem o processo de desenvolvimento. Essa plataforma possui diversos recursos, inclusive a integração com o Unity. Através do serviço Firebase Authentication, é realizada a autenticação do usuário vinculada a uma conta

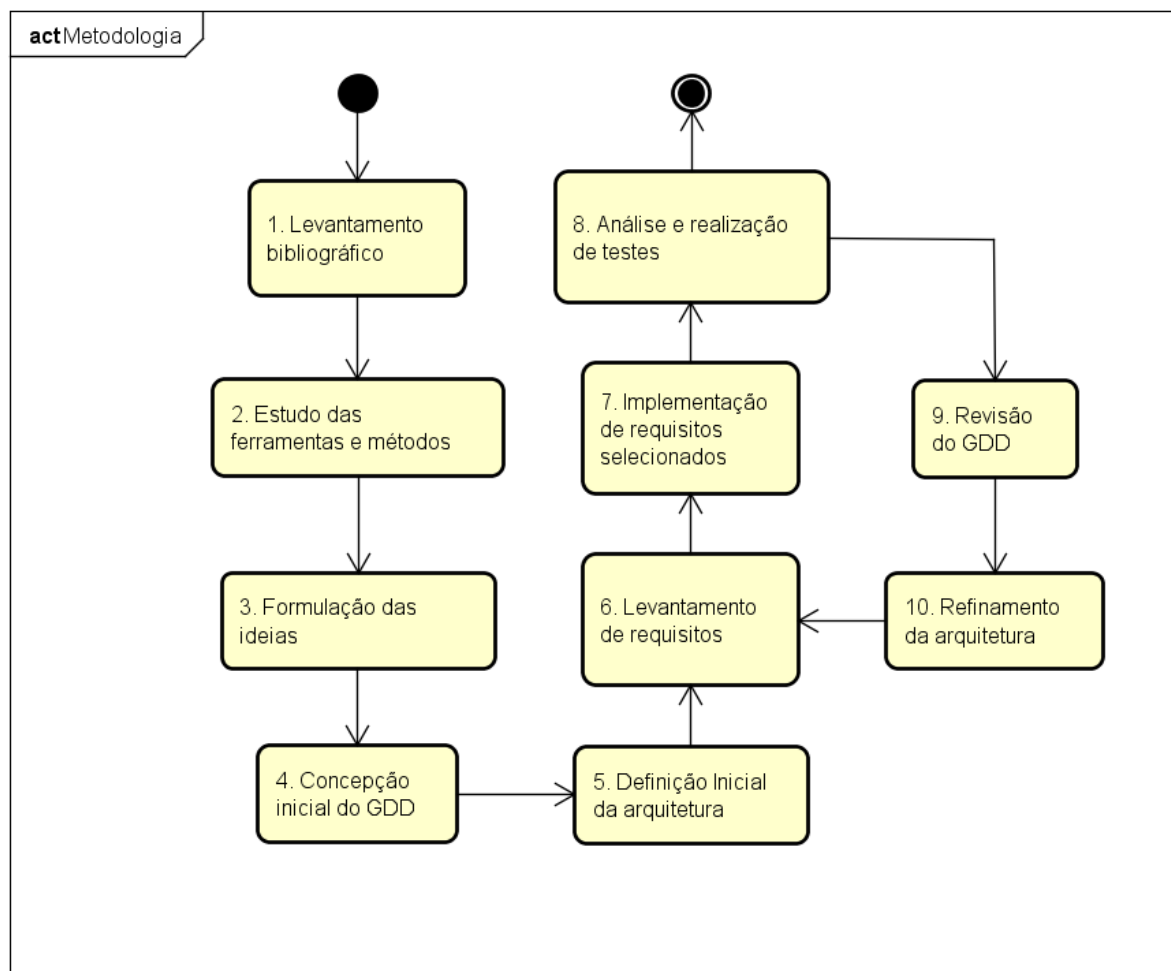
¹ <https://unity3d.com/>

Google, para que através do *login*, seja possível salvar o seu progresso (FIREBASE, 2017).

4 METODOLOGIA

Na Figura 5 encontram-se as etapas de desenvolvimento através de um fluxograma geral. Vale ressaltar que nas etapas 1, 2 e 3 não é necessário o término da anterior para iniciar a posterior, estes passos podem ser feitos e refeitos em todos os momentos do projeto. Porém, principalmente qualquer alteração no passo 3 resulta em grandes impactos nos subsequentes.

Figura 5 – Fluxograma da metodologia geral do projeto



FONTE: O AUTOR, 2018

4.1 Passo 1 – Levantamento bibliográfico

Inicialmente foi realizado um levantamento bibliográfico acerca do tema do projeto e constatada nossa lacuna de pesquisa, como apresentado na introdução: a dificuldade enfrentada pelos estudantes na disciplina de química, escassez de laboratórios e a utilização de jogos eletrônicos educativos como forma de solucionar esse problema.

Posteriormente, o levantamento se deu em função de arquiteturas e metodologias eficientes para o desenvolvimento de jogos eletrônicos. Isso permitiu que fosse criada uma simplificação do método ágil de desenvolvimento de software SCRUM. A metodologia proposta segue as boas práticas de desenvolvimento de jogos eletrônicos discutidas no capítulo 2 do livro de Schell (2014).

4.2 Passo 2 – Estudo das ferramentas e métodos

As ferramentas escolhidas para a execução do projeto foram selecionadas com base em conhecimentos prévios dos autores desse projeto, sendo elas: 3DS Max para modelagem tridimensional e animação; Substance Painter para texturização e coloração dos modelos; MotionBuilder para animação com base em dados de *motion capture*; Unity 3D como motor de jogos, Firebase como plataforma unificada de recursos que foram utilizados para o armazenamento de dados e autenticação de usuários e Microsoft Visual Studio 2017 como IDE de programação, todas descritas anteriormente no Capítulo 3 deste trabalho.

Estudos mais aprofundados dessas ferramentas se deram principalmente por vídeo aulas nas plataformas: YouTube, Udemy e Pluralsight, além de páginas de Internet especializadas em conteúdos para desenvolvedores: Stackoverflow e os fóruns do Unity 3D.

4.3 Passo 3 – Formulação das ideias

Como fundamentado no Capítulo 2, a formulação das ideias geralmente se dá através de seções de *brainstorming*, como utilizado para o levantamento das ideias deste projeto. Tais seções foram agendadas com os membros do projeto e ocorreram entre os meses de agosto e novembro de 2017, no Instituto Federal de Educação, Ciências e Tecnologia do Mato Grosso do Sul – *Campus Nova Andradina*. Definindo assim o escopo do projeto, fundamentando os seis seguintes itens:

1. Plataformas nas quais o jogo iria ser voltado;
2. Quais seriam os personagens;
3. O modo que seria a interação homem computador;
4. O universo que o jogo eletrônico se passaria;
5. Quais os tipos de desafios que o jogador teria que superar;
6. Como será o processo de aprendizagem pelo jogo.

Os assuntos dessas discussões foram compilados em arquivos de texto e então disponibilizados a todos participantes via *Google Drive*.

4.4 Passo 4 – Concepção inicial do *Game Design Document* - GDD

Tendo concluído a etapa de formulação das ideias pode dar início ao documento norteador do projeto. Esse documento deve conter a descrição dos personagens do jogo, seu tema, forma de se jogar (*gameplay*), controles, movimentação das câmeras no jogo, o universo, sua interface, etc. Tal documento é chamado de GDD (*Game Design Document*) (FULLERTON, 2004).

Pela metodologia empregada neste trabalho, esse passo consiste apenas em formular a concepção inicial do GDD. Sua versão final só virá quando o processo de desenvolvimento de todo o jogo for atingido. Isso se dá pelo fato

que sua escrita é feita de maneira iterativa para se adaptar a novas mudanças incorporadas durante o desenvolvimento de todo o jogo.

Abaixo é apresentada uma descrição de itens genéricos, ou seja, podem ser empregados em jogos de diferentes temas. Esses itens seguem recomendações descritas por Rogers (2014) e Schell (2014). Através desses itens, foi selecionado os que contemplavam aspectos do jogo, foi escrito um documento que está exposto de maneira detalhada no Capítulo 6 desse trabalho.

- Nome do jogo
 - Qual o nome do jogo?
 - Qual o logotipo do jogo? Logotipo é uma assinatura institucional, a representação gráfica de uma marca.
- História
 - Descrição detalhada da história (lembre-se que toda história deve conter um começo, meio e fim);
 - A descrição da história deve conter uma breve descrição do ambiente onde o jogo acontece e também dos principais personagens envolvidos na história.
- Personagens
 - Descrição das características dos personagens principais (nome, idade, tipo...);
 - História do passado dos personagens;
 - Personalidade dos personagens;
 - Habilidades características de cada personagem (poderes especiais, golpes especiais, armas...);
 - Ilustração visual dos personagens;
 - Ações que os personagens podem executar (andar, correr, pular, pulo duplo, escalar, voar, nadar...).
- Inimigos
 - Quais inimigos são encontrados no universo do jogo?
 - O que os torna únicos?
 - Como o jogador os supera?

- Controles
 - Como o jogador controla o personagem principal?
- *Gameplay*
 - Descrição da mecânica do jogo;
 - Quais são os desafios encontrados pelo jogador e quais os métodos usados para superá-los?
 - Como o jogador avança no jogo e como os desafios ficam mais difíceis?
 - Como o *gameplay* está relacionado com a história? O jogador deve resolver quebra-cabeças para avançar na história? Ou deve vencer chefões para progredir?
 - Como funciona o sistema de recompensas? Pontos, dinheiro, experiência, itens colecionáveis, armas, poderes? Quais os benefícios que o jogador tem com cada um desses itens?
 - Qual é a condição de vitória? Salvar o universo? Matar todos os inimigos? Coletar 100 estrelas? Todas as alternativas acima?
 - Qual é a condição de derrota? Perder 3 vidas? Ficar sem energia?
- Câmera
 - Como é a câmera do jogo? Como o jogador visualiza o jogo?
 - Ilustre visualmente como o jogo será visualizado;
- Universo do jogo
 - Descrição e ilustração dos cenários do jogo;
 - Como as fases do jogo estão conectadas?
 - Qual a estrutura do mundo?
 - Qual a emoção presente em cada ambiente?
 - Que tipo de música deve ser usada em cada fase?
 - Inclua ilustrações de todos os mapas e fases do jogo;
- Interface
 - Design e ilustração do HUD (*head-up display*);
 - Posicionamento dos elementos do HUD;
 - Design e ilustração das interfaces do jogo: tela inicial, menu de opções, tela de pause, menu de itens, tela de loading, etc...

- Filmes e *cutscenes*
 - Seu jogo tem filmes ou cenas?
 - Como eles serão apresentados ao jogador?

4.5 Passo 5 – Definição inicial da arquitetura

No Capítulo 3 foi apresentada a arquitetura de software utilizada neste passo para fazer sua definição inicial, o MVC. Para concluir esse passo foi utilizado o software A²stah para montar um diagrama de classes bastante genérico com a função de guiar o projeto nos estágios iniciais da programação. Tal diagrama foi construído com base em informações discutidas na fundamentação das ideias e no documento GDD inicial.

Assim como o GDD, a arquitetura do software também é feita de maneira iterativa. Porém, o software Astah não foi utilizado para ficar constantemente atualizando os diagramas. Para essas mudanças usou uma ferramenta do Microsoft Visual Studio 2017 que transforma o código implementado em um diagrama de classes.

4.6 Passo 6 – Levantamento de requisitos

Pela metodologia iterativa proposta, o levantamento de requisitos foi repetido muitas vezes, mas, inicialmente, foi gerada uma lista de requisitos baseados na estrutura de classes, que foi construída no passo de definição inicial da arquitetura, e utilizadas informações do passo que construiu o documento inicial do GDD. É importante lembrar que para um jogo eletrônico não existirão só requisitos de programação, mas também de arte, sons, animações, imagens, etc.

² <http://astah.net/>

O processo de pegar ideias e transformá-las em requisitos é bastante complexo, já que, em muitos casos, os membros da equipe não estão cientes das possibilidades que existem e podem se limitar por suas concepções. Por exemplo, pedir a uma pessoa do século 19 que liste as funcionalidades necessárias para um veículo que transporta vários passageiros, teria resultado em uma carruagem puxada por cavalos.

Além disso, precisa ter cuidado com o velho ditado, "não irei listar determinado requisito porque não existe uma ideia clara que o fundamente". Se encontrar um requisito que for necessário, ele deve ser posto em discussão para fomentar novas ideias e talvez até voltar alguns passos.

Para criar a lista de requisitos, foram utilizados os seguintes itens:

- Tipo do requisito (programação, modelagem 3D, pintura 3D e animação, arte 2D, vídeo, áudio);
- Descrição do requisito;
- Dia que o requisito foi proposto;
- Número da *sprint*;
- Data que o requisito foi concluído.

Tal lista foi disposta em uma tabela como apresentada na Figura 6:

Figura 6 – Lista de requisitos

[illegible]

FONTE: O AUTOR, 2018

À medida que novos requisitos vão surgindo, essa lista é atualizada e ordenada de forma que os itens mais importantes sempre fiquem como próximos. Tal lista é o *Product Backlog* da metodologia de desenvolvimento ágil SCRUM.

Como dito anteriormente, a metodologia de desenvolvimento que está sendo utilizada é baseada no SCRUM. Cabe ressaltar as diferenças. O método utilizado na parte de desenvolvimento desse trabalho é apresentado na Figura 7.

Figura 7: Metodologia ágil SCRUM modificada



FONTE: O AUTOR, 2018

Observando a Figura 7, é possível perceber semelhanças com a apresentada no Capítulo 2, assim como algumas mudanças. Foi removido o *daily SCRUM* e o tempo de execução da *sprint* não é fixo. Tais modificações foram necessárias devido a complicações de horários que os participantes do projeto estão inseridos. Calendário escolar, semanas de prova, semanas acadêmicas, viagens técnicas, tempo para preparações de aulas, etc. são alguns exemplos

de acontecimentos que não permitem manter um calendário fixo para conclusão de *sprints*, nem para manter reuniões diárias.

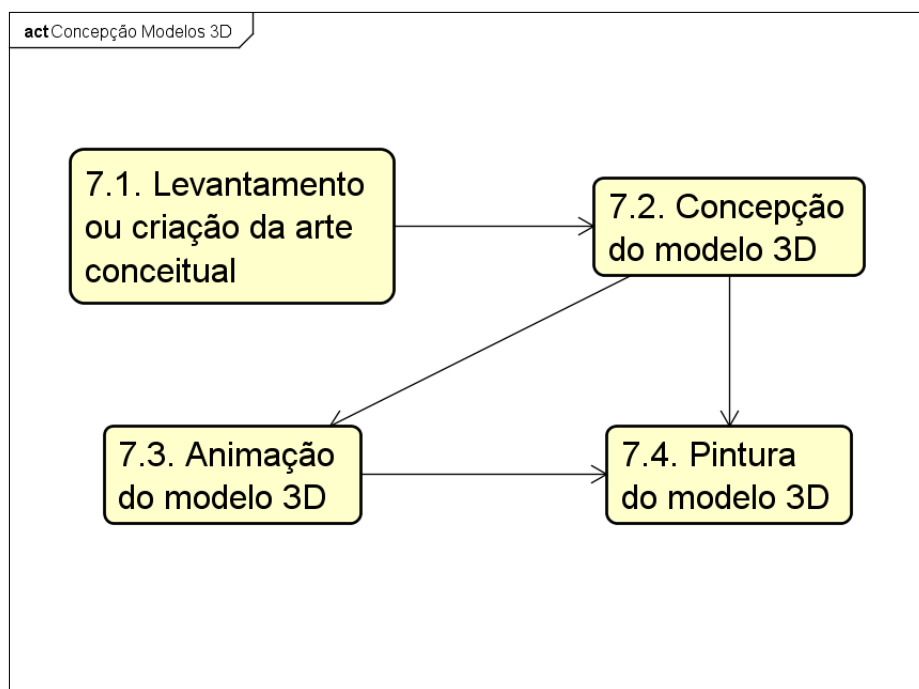
4.7 Passo 7 – Implementação de requisitos selecionados

Esse é o passo principal para criação de um jogo. É aqui que tudo o que foi planejado começa a ser codificado, criadas as artes, sons, imagens. E, para cada uma dessas partes existe um *workflow* (fluxo de trabalho) específico.

4.7.1 Concepção dos modelos 3D

Para a concepção de todos os modelos 3D foram seguidas as etapas apresentadas na Figura 8. Na figura, a etapa 7.3. só é realizada em objetos que necessitam de animações, como humanoides.

Figura 8 – Metodologia para concepção dos modelos 3D

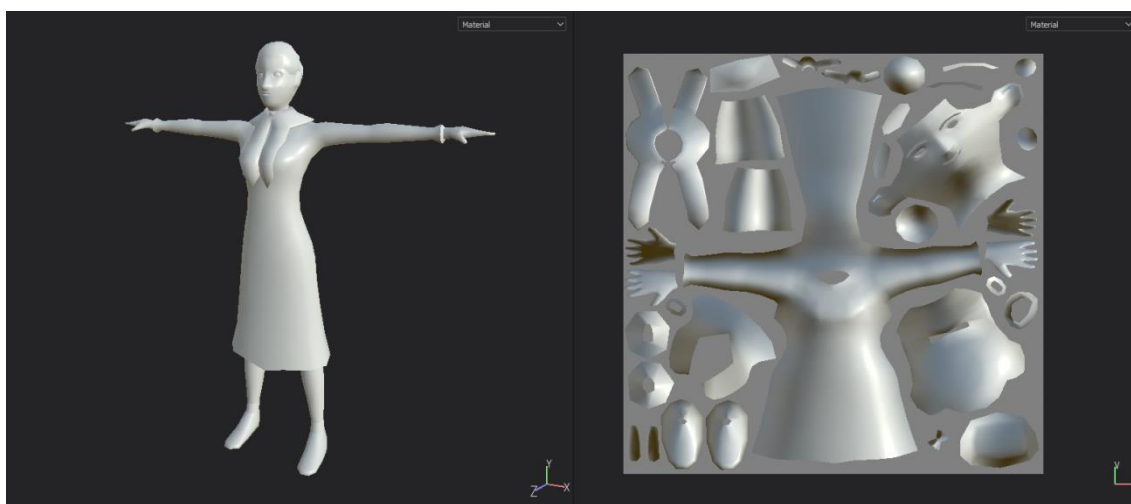


4.7.1.1 Modelagem 3D

Com o propósito de iniciar a modelagem 3D, primeiramente deve ser realizado um levantamento da arte conceitual ou a criação da mesma (etapa 7.1.). Nessa etapa efetuam-se buscas de imagens, são construídos esboços de objetos (*sketch*) e obtidas informações a respeito das dimensões do objeto que será realizada a modelagem.

Tendo finalizado a etapa 7.1., inicia a etapa 7.2. com a modelagem no software 3DS Max, utilizando recursos e modificadores disponíveis pelo software de acordo com a necessidade de cada objeto. Finalizando a modelagem, é necessária a construção do mapa UV (uma imagem 2D do modelo 3D elaborado). Tal mapa é utilizado para a pintura de objetos em 3D. Um exemplo de mapa UV pode ser observado na Figura 9.

Figura 9 - Exemplo de mapa UV

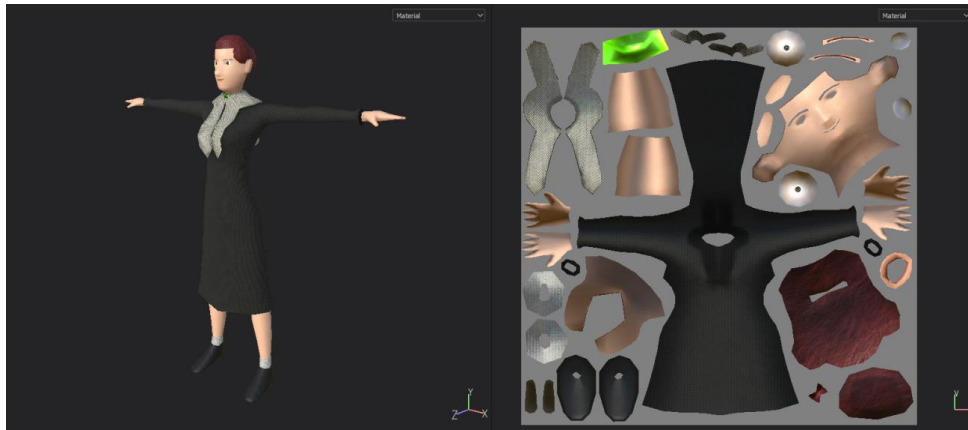


FONTE: O AUTOR, 2018

Após a finalização da modelagem e do mapa UV, o objeto é exportado para o SubstancePainter, onde pode realizar a etapa 7.4. construindo as

texturas. Como exemplo, a Figura 10 apresenta o modelo de Marie Currie com a texturização completa.

Figura 10 - Exemplo de mapa UV colorido



FONTE: O AUTOR, 2018

4.7.1.2 Construção das Animações

Para a animação de modelos orgânicos, como pessoas e animais, foi utilizada a técnica de animação por *bones*. Para isso, inicialmente são colocadas algumas estruturas lógicas dentro das malhas dos modelos a fim de que, futuramente, tais estruturas manipulem seus vértices. Esse processo é conhecido pelo nome inglês *rigging*. A Figura 11 mostra um exemplo de *bones*.

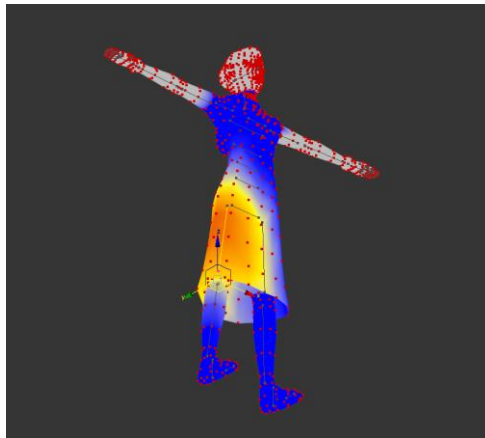
Figura 11 - Estruturação dos *bones* em um personagem humanóide



FONTE: O AUTOR, 2018

Depois de estruturar os *bones* necessários para criar as animações de um modelo utilizando tal técnica, é necessário manipular quais vértices da malha 3D serão afetados por cada osso. Esse processo é chamado pelo termo em inglês de *skinning*. Na Figura 11, é possível observar o conjunto de vértices afetado pelo osso do fêmur direito do modelo humanoide.

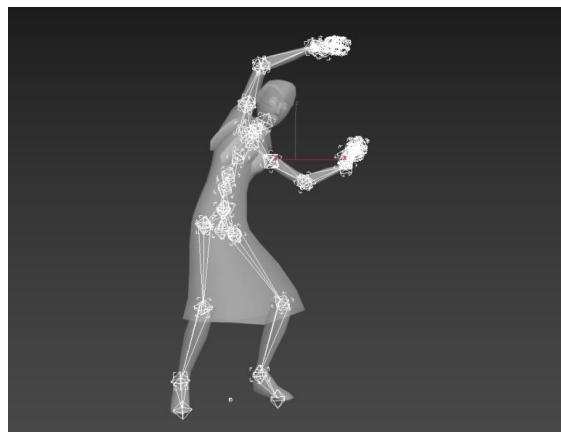
Figura 11 - Exemplo do processo de skinning em um modelo humanoide



FONTE: O AUTOR, 2018

Após terminar essa etapa, quando qualquer osso for movimentado, a malha do modelo seguirá o osso baseado no peso (*weight*) que o osso exerce em determinado vértice. Depois disso, é feita a criação das animações através de poses (*key frames*) ou utilizando dados de *motion capture*, como exemplificado na Figura 12.

Figura 12 - Exemplo de um modelo animado por bones

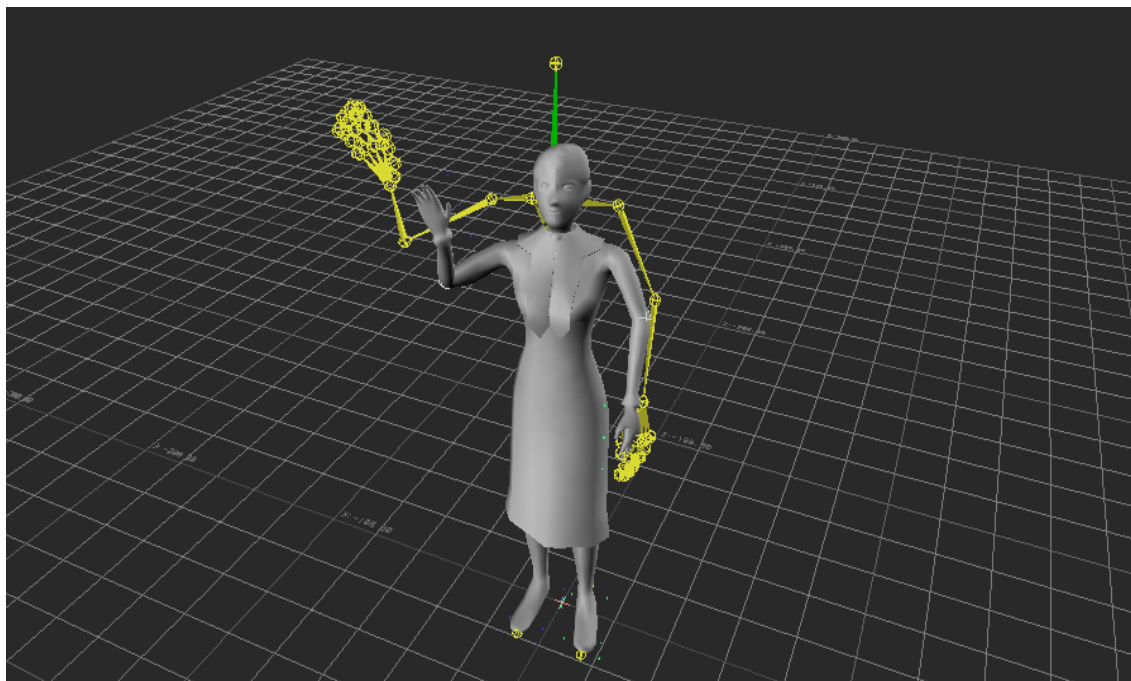


FONTE: O AUTOR, 2018

Em modelos simples (não orgânicos), não é utilizado dados de *motion capture* nem animação por *bones*. As animações mais simples são realizadas apenas com o 3DS Max por meio de *keyframes* de animação.

Para os modelos humanoides, depois do processo de *skinning*, foi utilizado o programa MotionBuilder e os dados de *motion capture* disponibilizados pelo site <https://www.mixamo.com/>. Assim, não é necessário criar animações complexas como andar, pular, ou pegar itens no chão, uma vez que esses dados gerados por equipamentos específicos estão disponibilizados pelo site. A Figura 13 apresenta o modelo sendo animado no MotionBuilder.

Figura 13 - Exemplo de um modelo animado por *bones*



FONTE: O AUTOR, 2018

4.7.2 Processo de Desenvolvimento na *Game Engine*

O primeiro passo é o levantamento dos *game assets* que serão utilizados no jogo. Sendo que objeto em um jogo é formado por um conjunto de *assets*.

Esse processo é realizado durante todo o processo de concepção e desenvolvimento do jogo, isto é, desde o levantamento dos requisitos até sua finalização.

Muitos *assets* são desenvolvidos por membros da comunidade ou por desenvolvedores do Unity 3D, podendo eles serem compatíveis com inúmeros diferentes propósitos, por exemplo, um *script* de posicionamento de câmera pode ser incluso em vários projetos diferentes que utilizam a mesma configuração.

Tendo definido os *assets* que farão a base do jogo, o próximo passo é a importação dos que foram criados de forma específica para esse trabalho. Assim, pode ser iniciado o primeiro passo na programação dos *scripts* que irão controlar os objetos. Essa programação é feita de forma isolada de início, ou seja, sem considerar a interação de tal objeto com outros em jogo eletrônico. Tendo concluído essa etapa, é finalizada a programação inserindo os trechos no código para lidar com a interação dos objetos.

Para ilustrar o que foi dito no parágrafo anterior, considere o caso do modelo do personagem principal: inicialmente ele é importado na *game engine* e assim inicia a programação de seus controladores de forma separada. É possível, por exemplo, criar um *script* para fazer o personagem andar e pular antes de implementar os *scripts* que lidarão com funções de interação como pegar itens no chão ou usar determinado objeto em um cenário.

4.8 Passo 8 – Análise e realização de testes

Esse passo consiste em verificar novos requisitos que foram levantados durante o processo de desenvolvimento a fim de refinar futuramente o GDD e a arquitetura do software.

Também nessa etapa é realizada uma bateria de teste para ter certeza de que o que foi implementado funciona conforme o especificado.

4.9 Passo 9 – Revisão do GDD

Etapa que consiste em atualizar o GDD com novas informações observadas durante o processo de análise e realização de testes.

4.10 Passo 10 – Refinamento da arquitetura

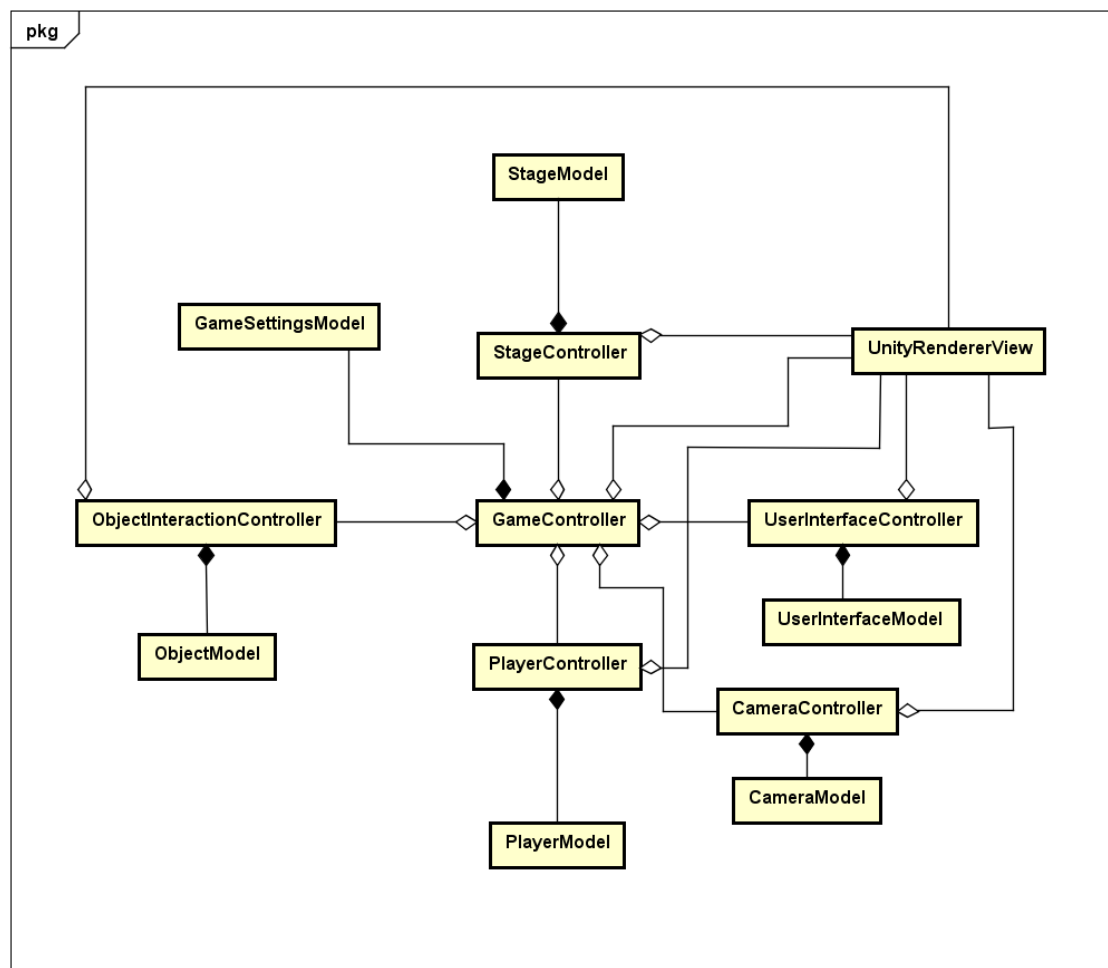
Etapa que consiste em atualizar a arquitetura de software com novas informações observadas durante o processo de análise e realização de testes.

5 DESENVOLVIMENTO

5.1 Definição inicial da arquitetura

Tendo concluído o processo de formulações das ideias e ter concluído a primeira versão do GDD, o desenvolvimento foi iniciado com a criação da arquitetura de software inicial utilizando o software Astah. Essa arquitetura é baseada no MVC e pode ser visualizada na Figura 14.

Figura 14 - Definição inicial da arquitetura



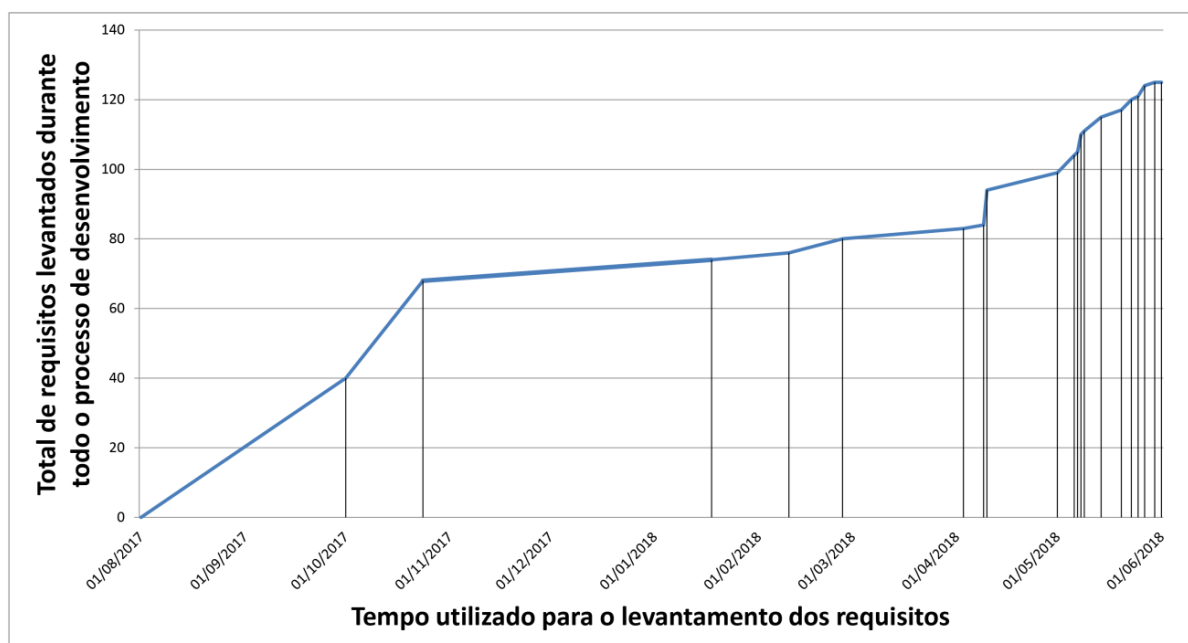
FONTE: O AUTOR, 2018

Esse modelo é bastante genérico e pode ser empregado em uma variedade bem grande de jogos.

5.2 Levantamento de requisitos

A Figura 15 apresenta a quantidade de requisitos levantados desde o início do projeto até esse momento. Um total de 125 requisitos.

Figura 15 - Levantamento de requisitos

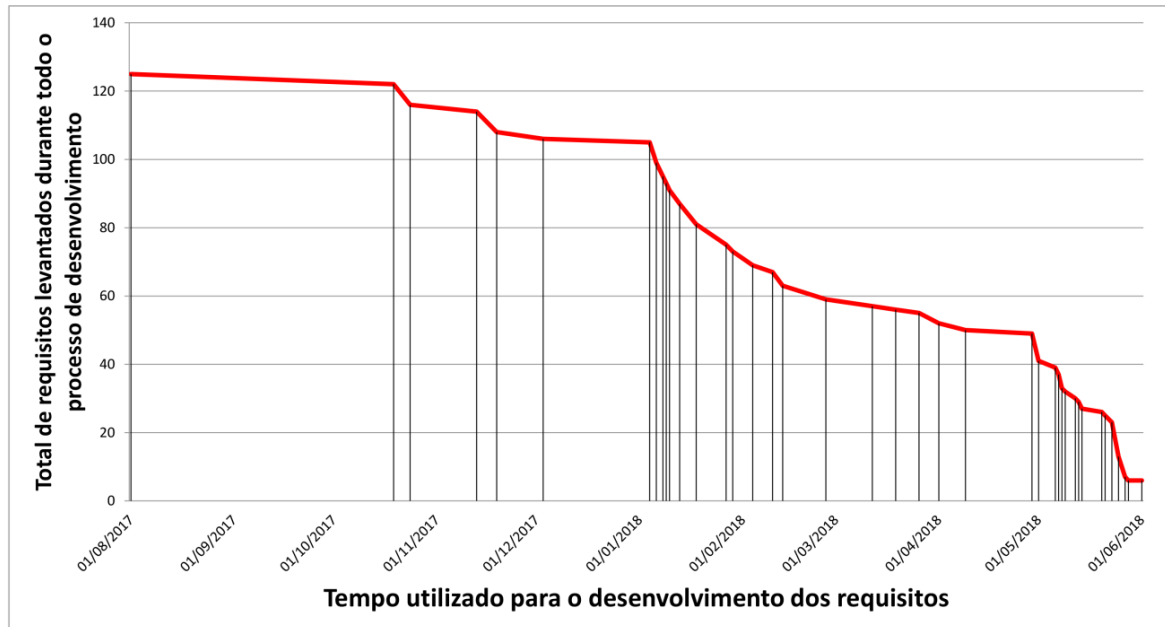


FONTE: O AUTOR, 2018

5.3 Desenvolvimento de requisitos

A Figura 16 apresenta a o gráfico de requisitos concluídos conforme o tempo do projeto foi passando.

Figura 16 - Desenvolvimento de requisitos



FONTE: O AUTOR, 2018

5.4 Requisitos concluídos

A Tabela 2, lista de todos os requisitos concluídos até o momento.

Tabela 2 - Requisitos concluídos

Product Backlog				
Tipo	Descrição funcionalidade	Data proposto	Sprint	Data conclusão
Programação	Organização do projeto inicial no Unity 3D.	01/10/2017	Sprint 1	19/10/2017
Programação	Tela de login do Jogo.	01/10/2017		19/10/2017
Programação	Entrando no jogo através de conta do Google (Firebase).	01/10/2017		19/10/2017
Modelo 3D	Personagem Marie Curie.	01/10/2017	Sprint 2	24/10/2017
Pintura 3D	Personagem Marie Curie.	01/10/2017		24/10/2017
Animação	Personagem Marie Curie.	01/10/2017		24/10/2017
Programação	Movimentação da câmera com angulação isométrica do cenário pela ação de toque simples na tela.	01/10/2017		24/10/2017
Programação	Zoom da câmera com a ação de belisco.	01/10/2017		24/10/2017
Programação	Movimentação do personagem pelo cenário através do toque duplo na tela.	01/10/2017		24/10/2017
Programação	Inventário para armazenar os itens do jogo.	01/10/2017	Sprint 3	13/11/2017
Programação	Interface para acionar o inventário pessoal da personagem.	01/10/2017		13/11/2017
Modelo 3D	Vidro Erlenmeyer.	01/10/2017	Sprint 4	19/11/2017
Pintura 3D	Vidro Erlenmeyer.	01/10/2017		19/11/2017
Modelo 3D	Vidro Round Bottom.	01/10/2017		19/11/2017
Pintura 3D	Vidro Round Bottom.	01/10/2017		19/11/2017
Modelo 3D	Tubo de ensaio.	01/10/2017		19/11/2017
Pintura 3D	Tubo de ensaio.	01/10/2017		19/11/2017
Modelo 3D	Melhoramento do modelo 3D da Marie Curie.	24/10/2017	Sprint 5	03/12/2017
Pintura 3D	Melhoramento do modelo 3D da Marie Curie.	24/10/2017		03/12/2017

Programação	Criação de novos itens a partir da junção de outros pré-existentes (sistema de craft).	01/10/2017	Sprint 6	04/01/2018
Modelo 3D	Banquinho de laboratório.	24/10/2017	Sprint 7	06/01/2018
Pintura 3D	Banquinho de laboratório.	24/10/2017		06/01/2018
Modelo 3D	Microscópio.	24/10/2017		06/01/2018
Pintura 3D	Microscópio.	24/10/2017		06/01/2018
Modelo 3D	Agitador de frascos.	01/10/2017		06/01/2018
Pintura 3D	Agitador de frascos.	01/10/2017		06/01/2018
Modelo 3D	Balança Analítica.	01/10/2017	Sprint 8	07/01/2018
Pintura 3D	Balança Analítica.	01/10/2017		07/01/2018
Modelo 3D	Mesa de escritório para o laboratório.	24/10/2017		07/01/2018
Pintura 3D	Mesa de escritório para o laboratório.	24/10/2017		07/01/2018
Modelo 3D	Mesa central do laboratório.	24/10/2017	Sprint 9	09/01/2018
Pintura 3D	Mesa central do laboratório.	24/10/2017		09/01/2018
Modelo 3D	Suporte de itens da mesa central do laboratório.	24/10/2017	Sprint 10	10/01/2018
Pintura 3D	Suporte de itens da mesa central do laboratório.	24/10/2017		10/01/2018
Modelo 3D	Prateleiras suspensas.	24/10/2017	Sprint 11	13/01/2018
Pintura 3D	Prateleiras suspensas.	24/10/2017		13/01/2018
Modelo 3D	Mesa de canto de laboratório.	24/10/2017		13/01/2018
Pintura 3D	Mesa de canto de laboratório.	24/10/2017		13/01/2018
Modelo 3D	Janela grande externa.	24/10/2017	Sprint 12	18/01/2018
Pintura 3D	Janela grande externa.	24/10/2017		18/01/2018
Modelo 3D	Geladeira.	01/10/2017		18/01/2018
Pintura 3D	Geladeira.	01/10/2017		18/01/2018
Modelo 3D	Capela.	01/10/2017		18/01/2018
Pintura 3D	Capela.	01/10/2017		18/01/2018
Modelo 3D	Paredes internas do laboratório.	24/10/2017	Sprint 13	27/01/2018

Pintura 3D	Paredes internas do laboratório.	24/10/2017		27/01/2018
Modelo 3D	Freezer horizontal.	01/10/2017		27/01/2018
Pintura 3D	Freezer horizontal.	01/10/2017		27/01/2018
Modelo 3D	Mesinha central da sala de recepção.	18/01/2018		27/01/2018
Pintura 3D	Mesinha central da sala de recepção.	18/01/2018		27/01/2018
Modelo 3D	Notebook.	18/01/2018	Sprint 14	29/01/2018
Pintura 3D	Notebook.	18/01/2018		29/01/2018
Modelo 3D	Bebedouro.	18/01/2018		04/02/2018
Pintura 3D	Bebedouro.	18/01/2018	Sprint 15	04/02/2018
Modelo 3D	Mesa da recepção.	24/10/2017		04/02/2018
Pintura 3D	Mesa da recepção.	24/10/2017		04/02/2018
Modelo 3D	Cadeira de escritório.	24/10/2017	Sprint 16	10/02/2018
Pintura 3D	Cadeira de escritório.	24/10/2017		10/02/2018
Modelo 3D	Porta principal e a porta do banheiro.	24/10/2017		13/02/2018
Pintura 3D	Porta principal e a porta do banheiro.	24/10/2017	Sprint 17	13/02/2018
Modelo 3D	Gramma alta.	10/02/2018		13/02/2018
Pintura 3D	Gramma alta.	10/02/2018		13/02/2018
Modelo 3D	Estrutura interna da recepção do laboratório.	24/10/2017		26/02/2018
Pintura 3D	Estrutura interna da recepção do laboratório.	24/10/2017	Sprint 18	26/02/2018
Modelo 3D	Janela do banheiro.	01/10/2017		26/02/2018
Pintura 3D	Janela do banheiro.	01/10/2017		26/02/2018
Modelo 3D	Quadro laboratório Marie Curie.	26/02/2018	Sprint 19	12/03/2018
Pintura 3D	Quadro laboratório Marie Curie.	26/02/2018		12/03/2018
Modelo 3D	Parte externa de laboratório (gramma e muros).	26/02/2018	Sprint 20	19/03/2018
Pintura 3D	Parte externa de laboratório (gramma e muros).	26/02/2018	Sprint 21	26/03/2018
Modelo 3D	Personagem Marie Curie com jaleco.	01/10/2017	Sprint 22	01/04/2018

Animação	Personagem Marie Curie com jaleco.	01/10/2017		01/04/2018
Pintura 3D	Personagem Marie Curie com jaleco.	01/10/2017		01/04/2018
Modelo 3D	Porta interna do laboratório.	01/10/2017	Sprint 23	09/04/2018
Pintura 3D	Porta interna do laboratório.	01/10/2017		09/04/2018
Programação	Abrir as portas quando a personagem chegar perto.	10/04/2018	Sprint 24	29/04/2018
Programação	Ajuste: Sombras e Iluminação interna do laboratório.	10/04/2018	Sprint 25	01/05/2018
Modelo 3D	Árvore1 exterior.	10/04/2018		01/05/2018
Pintura 3D	Árvore1 exterior.	10/04/2018		01/05/2018
Modelo 3D	Árvore2 exterior.	10/04/2018		01/05/2018
Pintura 3D	Árvore2 exterior.	10/04/2018		01/05/2018
Modelo 3D	Árvore3 exterior.	10/04/2018		01/05/2018
Pintura 3D	Árvore3 exterior.	10/04/2018		01/05/2018
Modelo 3D	Melhorando modelo do laboratório (aumentando o banheiro).	30/05/2018		01/05/2018
Programação	Interação entre com os itens.	03/04/2018	Sprint 26	06/05/2018
Programação	Melhoramento do game controller.	01/05/2018		06/05/2018
Programação	Ajuste: Terminar de fazer a interação de todos os itens do cenário.	06/05/2018	Sprint 27	07/05/2018
Programação	Troca entre o modelo de jaleco e o modelo normal.	06/05/2018		07/05/2018
Programação	Configurando o MouseOver e o TouchOver nos itens do cenário.	08/05/2018	Sprint 28	08/05/2018
Programação	Melhorando a movimentação pelos toques na tela	08/05/2018		08/05/2018
Programação	Correção: Arrumando um defeito na tela de abertura que fez a personagem desaparecer.	08/05/2018		08/05/2018
Programação	Correção: Quando o jogador alternava entre os modelos da personagem o controlador do jogo perdia a referência do script do jogador	08/05/2018		08/05/2018
Programação	Impedir a personagem de entrar no laboratório sem as vestimentas adequadas.	07/05/2018	Sprint 29	09/05/2018
Programação	Construção do seletor de fases.	01/10/2017	Sprint 30	12/05/2018
Programação	Tela de load da fase.	08/05/2018		12/05/2018
Programação	Balão de conversa.	03/04/2018	Sprint 31	13/05/2018

Programação	Correção: Arrumando a rendering order dos balões de conversa.	14/05/2018	Sprint 32	14/05/2018
Programação	Adicionando os ícones nos equipamentos existentes.	14/05/2018	Sprint 33	14/05/2018
Programação	Script para fazer a personagem dizer coisas através dos balões de conversa.	20/05/2018	Sprint 34	20/05/2018
Programação	Menu para sair da fase.	01/10/2017	Sprint 35	21/05/2018
Programação	Craft de itens diferentes, baseado no equipamento utilizado.	06/05/2018	Sprint 36	23/05/2018
Programação	Ajuste: Terminando de organizar a interface dos menus.	23/05/2018		23/05/2018
Programação	Centralizar na personagem.	20/05/2018	Sprint 37	25/05/2018
Programação	Script para passar da primeira fase.	09/05/2018		25/05/2018
Programação	Painel para parabenizar o jogador por ter passado de estágio.	23/05/2018		25/05/2018
Modelo 3D	Banco do vestiário.	01/05/2018		25/05/2018
Pintura 3D	Banco do Vestiário.	01/05/2018		25/05/2018
Modelo 3D	Gabinete do Vestiário.	01/05/2018		25/05/2018
Pintura 3D	Gabinete do Vestiário.	01/05/2018		25/05/2018
Modelo 3D	Becker.	06/05/2018		25/05/2018
Pintura 3D	Becker.	06/05/2018		25/05/2018
Programação	Correção: Atualizando o Inventory System para quando a personagem transicionar entre os estágios não cause erros.	25/05/2018	Sprint 38	25/05/2018
Programação	Jogar itens do inventário sempre no pé da personagem.	27/05/2018	Sprint 39	27/05/2018
Modelo 3D	Modelo separado da pia da mesa central.	14/05/2018		27/05/2018
Pintura 3D	Modelo separado da pia da mesa central.	14/05/2018		27/05/2018
Programação	Menu de lista de tarefas.	03/04/2018		27/05/2018
Modelo 3D	Portas internas do banheiro.	10/04/2018		27/05/2018
Pintura 3D	Portas internas do banheiro.	10/04/2018		27/05/2018
Programação	Depósito de itens. (Geladeira, freezer, armário, etc.).	27/05/2018	Sprint 40	28/05/2018

FONTE: O AUTOR, 2018

5.5 Requisitos pendentes

A Tabela 3, lista de todos os requisitos pendentes até o momento.

Tabela 3 - Requisitos pendentes

Tipo	Descrição funcionalidade	Data proposto	Sprint	Data conclusão
Programação	Terminar construção da primeira fase.	01/10/2017		
Programação	Terminar construção da segunda fase.	01/10/2017		
Programação	Terminar construção da terceira fase.	01/10/2017		
Programação	Script para abrir a porta do banheiro.	27/05/2018		
Programação	Ajuste: Gramas exteriores.	09/04/2018		
Programação	Ajuste: Remover itens de interação temporários da memória.	23/05/2018		

FONTE: O AUTOR, 2018

5.6 Implementação da Autenticação

O processo de autenticação dos jogadores foi realizado utilizando a plataforma Firebase do Google. Assim, é necessária uma conta do Google para que o usuário possa entrar no jogo. O *login* (processo de autenticação de um usuário e senha) é realizado através da utilização Firebase Authentication Module (módulo de autenticação da Firebase).

De início foi estudada a interação do Unity 3D com a plataforma Firebase e estruturado como os scripts iriam se comunicar com os módulos do Google e obter as informações das credenciais baseadas no usuário e senha que o jogador inserir.

É interessante ressaltar que salvar as informações do usuário e senha nos arquivos do jogo não é uma boa prática, portanto, foi implementado um modo em que, quando o usuário efetua *login*, a comunicação entre o jogo e a plataforma Firebase ocorre de maneira direta, a qual retorna o *token* de credenciais informando se o usuário existe e se ele informou os dados corretamente. O *token* que o Google devolve reconhece apenas essa aplicação com determinado *token*, assim os problemas de segurança são resolvidos via *server side*.

Dessa forma, toda vez que essa aplicação utilizar determinado *token* com a plataforma Firebase para um usuário, o mesmo poderá ser validado. As aplicações então mantêm armazenados apenas o nome de usuário e o *token* validado.

Uma vez que um jogador realize um *login* e senha e obtenha um *token*, não é necessário que ele realize o processo de *login* novamente. Como o jogo desenvolvido armazena o usuário e o *token* obtido pelo processo de autenticação, quando a aplicação é aberta já é iniciado o processo de autenticação com o último usuário válido. Esse processo é chamado de *silent login* (autenticação silenciosa).

6 RESULTADO FINAL

6.1 Game Design Document – GDD

6.1.1 Personagem

Nascida em 1867, Marie Sklodowska foi uma cientista de grande nome. Após seu casamento passou a ser a Marie Sklodowska Curie, conhecida apenas como Marie Curie.

Em estudos junto com seu marido, Pierre Curie e Becquerel, ganha o prêmio Nobel de Física em 1903. Oito anos depois, em 1911 ela ganha o prêmio Nobel de Química, dentre outros diversos títulos e cargos recebidos de grande mérito. A Figura 17 apresenta um retrato de Marie Curie colorizado por Dana Keller.

Figura 17- Foto de Marie Curie colorizada por Dana Keller.

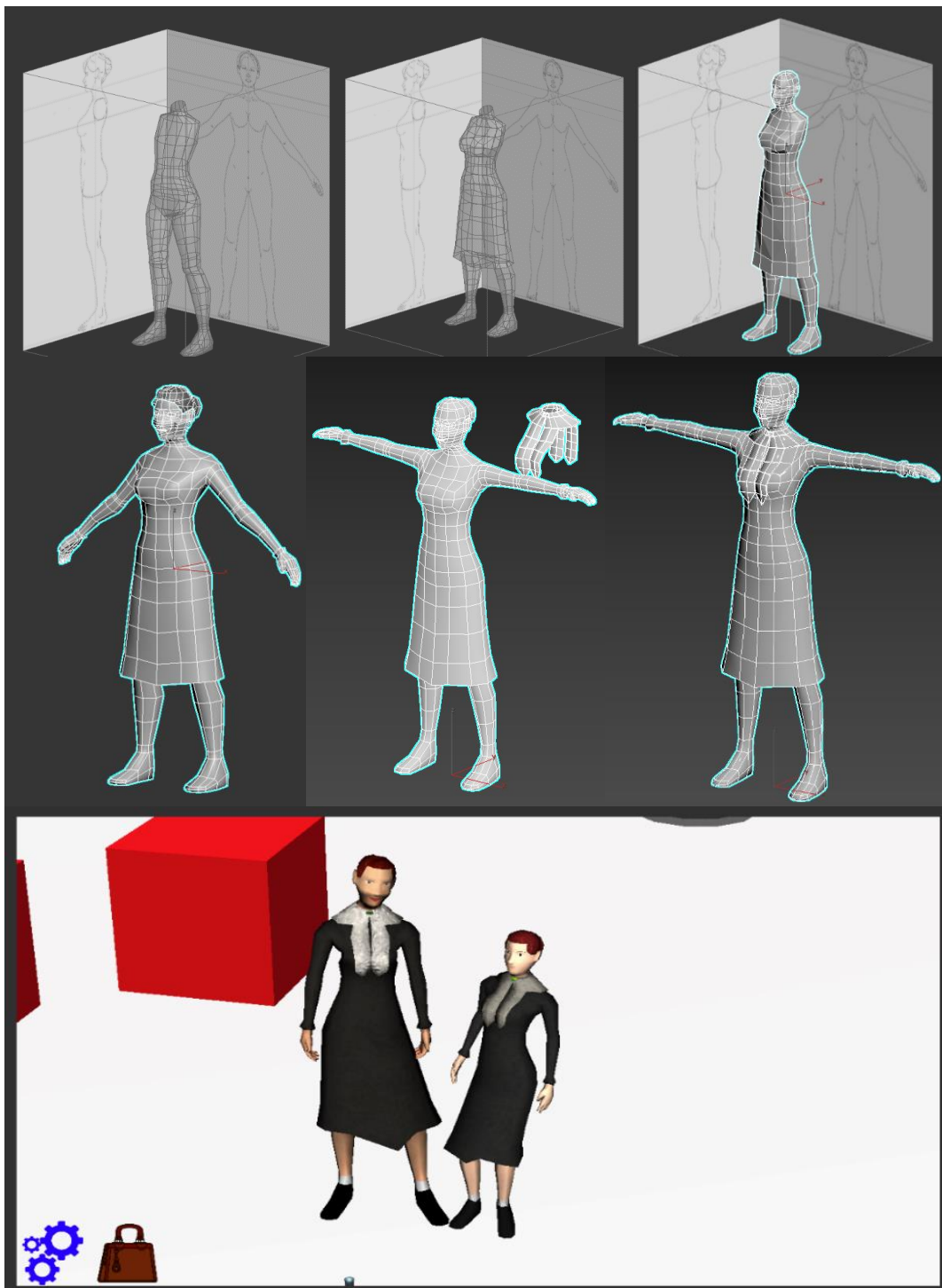


FONTE: HISTORY IN COLOR ³

³ Disponível em: <<http://www.danarkeller.com/>>

O processo de modelagem e os resultados obtidos estão apresentados na Figura 18. A personagem passou por uma evolução para o ajuste da altura e correção da face.

Figura 18 - Evolução do modelo 3D da personagem principal.



FONTE: O AUTOR, 2017

6.1.2 Gameplay

A mecânica principal do jogo consiste em recolher materiais básicos espalhados pelo laboratório e, com o uso de vários equipamentos típicos de laboratório o jogador deverá realizar desafios onde será necessário o conhecimento prévio de química inorgânica para realizar reações com ácidos, bases, sais e óxidos.

Os desafios encontrados pelo jogador são as etapas necessárias para criação de compostos químicos. Em cada estágio o jogador terá acesso a uma lista de objetivos para completar tal nível; tal lista iniciará fazendo o passo-a-passo de ações simplistas e, em níveis mais avançados, omitirá várias etapas para a conclusão da composição de tais compostos. Assim, esses objetivos se tornarão mais difíceis conforme o progresso do jogador, o que o levará a raciocinar como cada processo deverá ser realizado.

A condição de vitória consiste em avançar cada fase dentro de um tempo mínimo. Existem várias condições de derrota, por exemplo, estourar o tempo limite para concluir uma fase, esgotar os reagentes disponíveis, realizar reações que não estão como objetivo da fase, etc.

A fase 1 terá como objetivo a entrada no laboratório com os devidos equipamentos de segurança. Para isso, o jogador deverá entrar no prédio, ir até o banheiro, selecionar o vestiário. Assim que for efetuada a troca, a personagem poderá entrar no laboratório. Quando ela entrar, será obtida uma mensagem de sucesso.

6.1.3 Controles

O jogo foi desenvolvido para dispositivos móveis, logo, toda interação ocorrerá por meio de toques na tela ou informações a respeito do acelerômetro do celular.

Para o personagem principal:

- Toque simples e contínuo na tela corresponde a ação de iniciar o movimento da tela;
- A ação de beliscar a tela corresponde à mudança no fator de ampliação da tela;
- Toque duplo na tela corresponde a ação de movimentar o personagem até o lugar;
- Toque simples corresponde a ação de utilizar ou pegar algum item na tela.

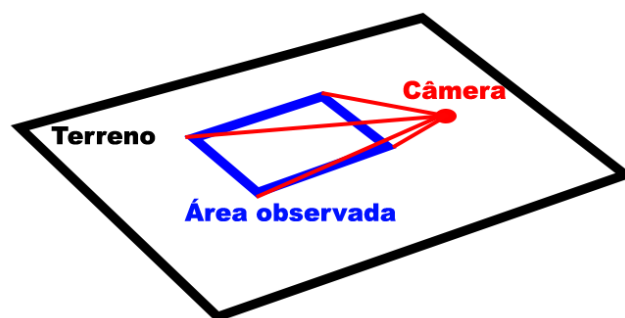
Para ações no inventário:

- Toque simples corresponde a ação de utilizar o item;
- Toque simples e contínuo corresponde a ação de desempilhar itens;
- Toque simples com ação de arrastar corresponde a ação de mover os itens pela bolsa.

6.1.4 Câmera

O jogo apresenta uma visão isométrica, ou seja, a perspectiva oferecida possui uma projeção de forma ortogonal, como apresentado na Figura 19. Desta maneira, a visão se aproxima da visão real, por isso, essa posição de câmera está entre as mais utilizadas.

Figura 19 - Representação da câmera com visão isométrica

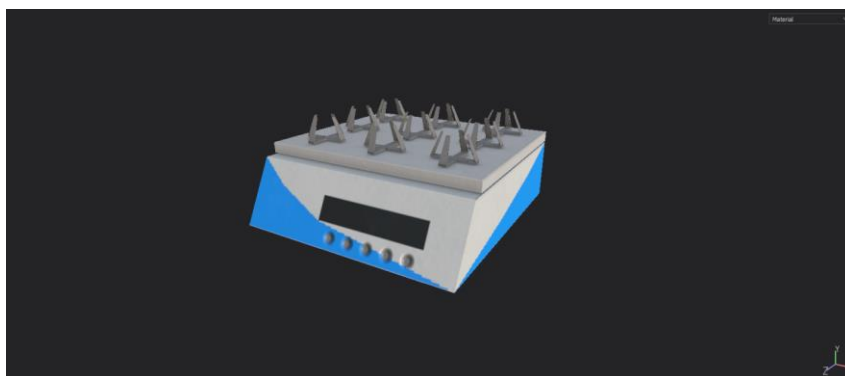


FONTE: O AUTOR, 2018

6.1.5 Universo do jogo

O jogo se passa em um laboratório de química. Assim, no cenário estão presentes vidrarias típicas, geladeiras, estantes, microscópios, e muitos outros equipamentos presentes nestes ambientes. Os itens que contemplam o cenário estão apresentados a seguir, da Figura 20 a Figura 52.

Figura 20- Agitador magnético



FONTE: O AUTOR, 2018

- **Modelo:** Lucas de Oliveira Queiroz e Leonardo Bravo Estácio;
- **Pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 670;
- **Material:** 4 imagens de 512x512 pixels.

Figura 21 - Armário metálico



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 14;
- **Material:** 4 imagens de 512x512 pixels.

Figura 22 - Armário suspenso



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 170;
- **Material:** 8 imagens de 512x512 pixels.

Figura 23 - Balança analítica



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 143;
- **Material:** 8 imagens de 512x512 pixels.

Figura 24 – Banco para o vestuário do banheiro

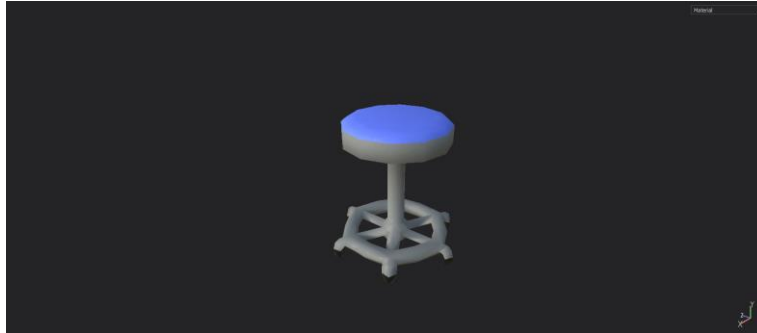


FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 48;

- **Material:** 4 imagens de 512x512 pixels.

Figura 25 - Banco pequeno



FONTE: O AUTOR, 2018

- **Modelo:** Alison Rocha da Silva e Leonardo Bravo Estácio;
- **Pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 306;
- **Material:** 4 imagens de 512x512 pixels.

Figura 26 - Bebedouro



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 437;
- **Material:** 8 imagens de 512x512 pixels.

Figura 27 - Cadeira de escritório



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio
- **Quantidade de polígonos (Quads):** 222;
- **Material:** 4 imagens de 512x512 pixels.

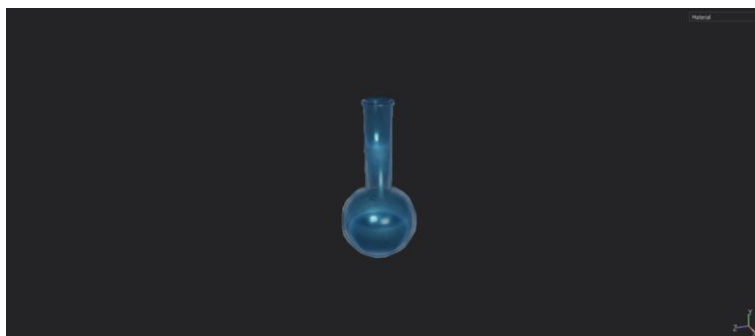
Figura 28 - Capela de química



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 904;
- **Material:** 8 imagens de 512x512 pixels.

Figura 29 - Frasco balão volumétrico



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 264;
- **Material:** 4 imagens de 512x512 pixels.

Figura 30 - Frasco Becker



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 46;
- **Material:** 4 imagens de 512x512 pixels.

Figura 31 - Frasco Erlenmeyer



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 143;
- **Material:** 4 imagens de 512x512 pixels.

Figura 32 - Frasco Tubo de Ensaio



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 132;
- **Material:** 4 imagens de 512x512 pixels.

Figura 33 - Freezer horizontal



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 264;
- **Material:** 8 imagens de 512x512 pixels.

Figura 33 - Geladeira



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 655;
- **Material:** 8 imagens de 512x512 pixels.

Figura 34 - Janela banheiro



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 53;
- **Material:** 8 imagens de 512x512 pixels.

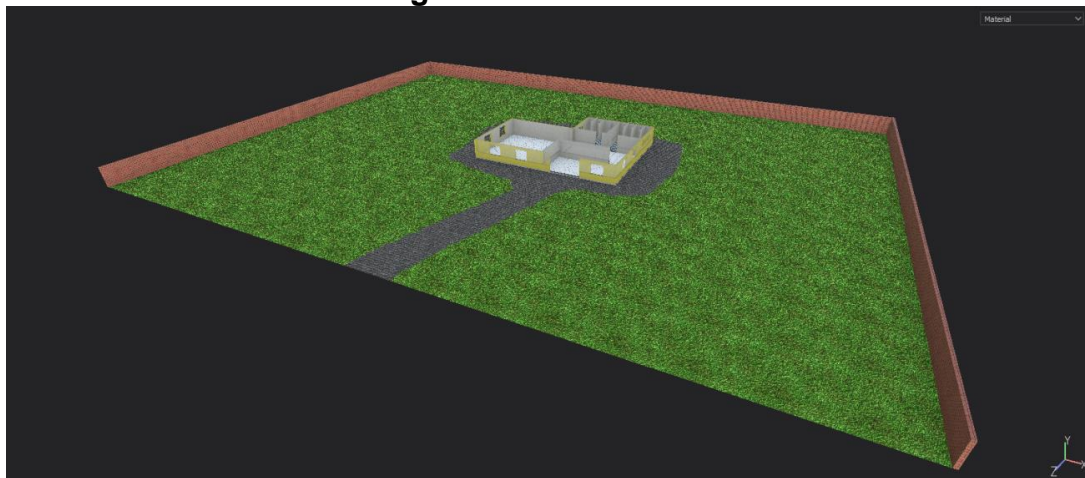
Figura 35 - Janela externa



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 406;
- **Material:** 8 imagens de 512x512 pixels.

Figura 36 - Laboratório



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 857;
- **Material:** 20 imagens de 2048x2048 pixels.

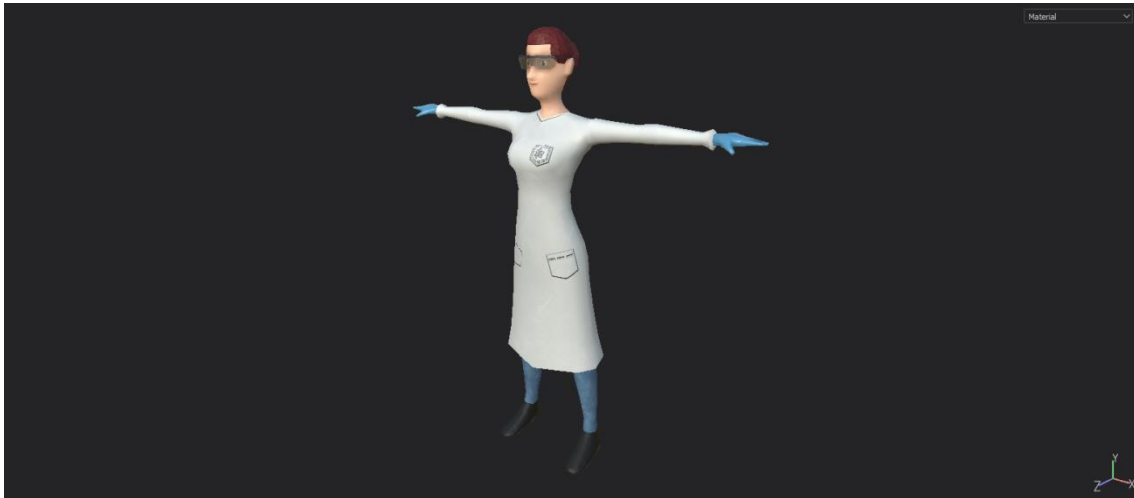
Figura 37 - Modelo Marie Curie



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio
- **Quantidade de polígonos (Quads):** 1618;
- **Material:** 4 imagens de 2048x2048 pixels.

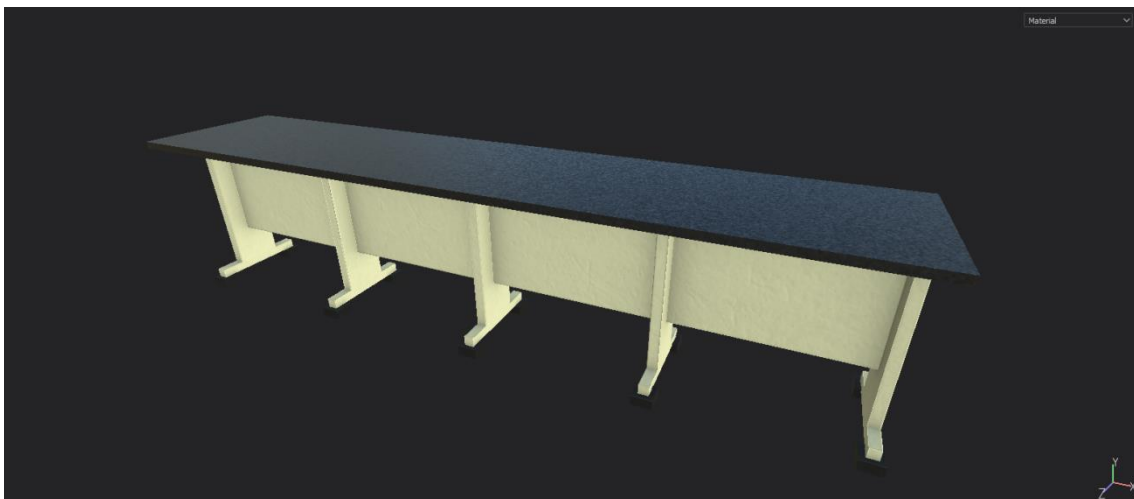
Figura 38 - Modelo Marie Curie com equipamentos de segurança



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio
- **Quantidade de polígonos (Quads):** 1464;
- **Material:** 4 imagens de 2048x2048 pixels.

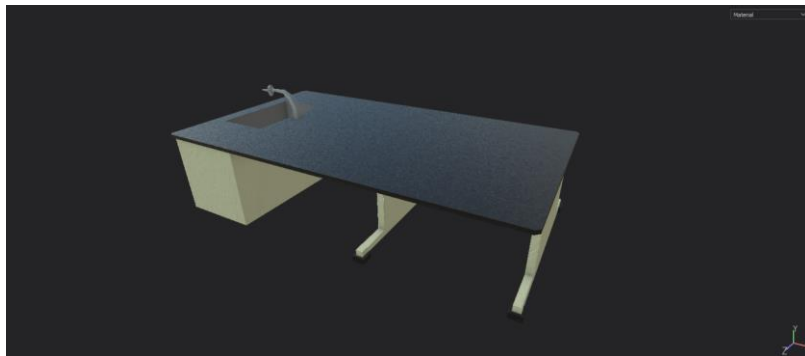
Figura 39 - Mesa de canto para laboratórios



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 393;
- **Material:** 4 imagens de 512x512 pixels.

Figura 40 - Mesa central para laboratórios



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 319;
- **Material:** 4 imagens de 512x512 pixels.

Figura 41 - Mesa de escritório



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 266;
- **Material:** 4 imagens de 512x512 pixels.

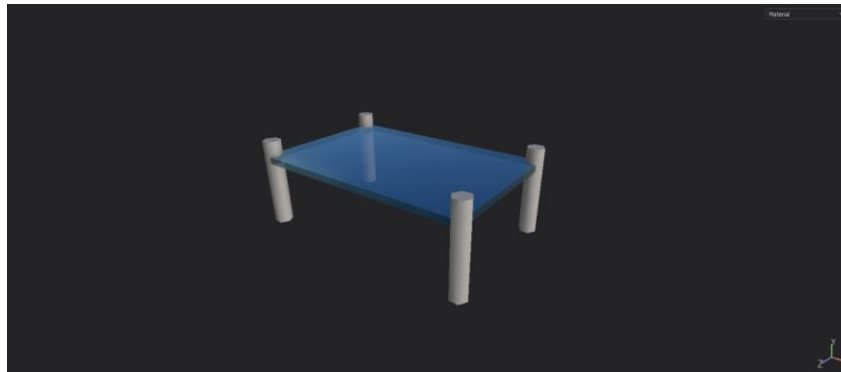
Figura 42 - Mesa da recepção do laboratório



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 73;
- **Material:** 4 imagens de 512x512 pixels.

Figura 43 - Mesinha de centro



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 100;
- **Material:** 8 imagens de 512x512 pixels.

Figura 44 - Microscópio



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 456;
- **Material:** 4 imagens de 512x512 pixels.

Figura 45 - Notebook



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 61;
- **Material:** 4 imagens de 512x512 pixels.

Figura 46 - Pia da mesa central



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 112;
- **Material:** 4 imagens de 512x512 pixels.

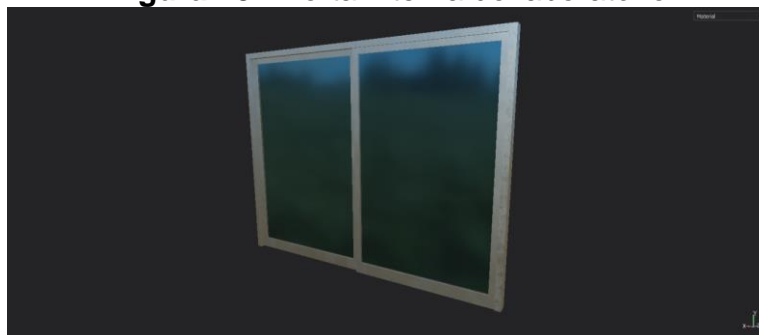
Figura 47 - Porta interna de banheiro



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 88;
- **Material:** 4 imagens de 512x512 pixels.

Figura 48 - Porta interna do laboratório



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 58;
- **Material:** 8 imagens de 512x512 pixels.

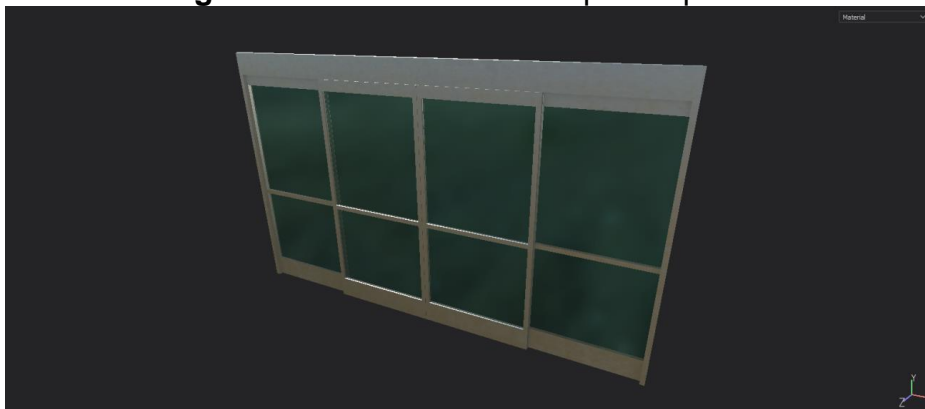
Figura 49 - Porta de acesso ao banheiro



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 92;
- **Material:** 4 imagens de 512x512 pixels.

Figura 50 - Porta de entrada para o prédio



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 182;
- **Material:** 8 imagens de 512x512 pixels.

Figura 51 - Quadro Marie Curie



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 18;

- **Material:** 4 imagens de 512x512 pixels.

Figura 52 - Suporte central para mesa de laboratório



FONTE: O AUTOR, 2018

- **Modelo e pintura:** Leonardo Bravo Estácio;
- **Quantidade de polígonos (Quads):** 124;
- **Material:** 8 imagens de 512x512 pixels.

6.1.6 Interface

Nas interfaces seguintes, estão exibidas o estado atual do cenário. Na Figura 53, é apresentada a interface inicial. O ícone do Google é para a realização da autenticação; contudo, após a primeira vez, não é necessário repetir caso a aplicação seja encerrada, a não ser em caso de alteração de dados na conta.

Figura 53 - Interface Inicial.



FONTE: O AUTOR, 2017

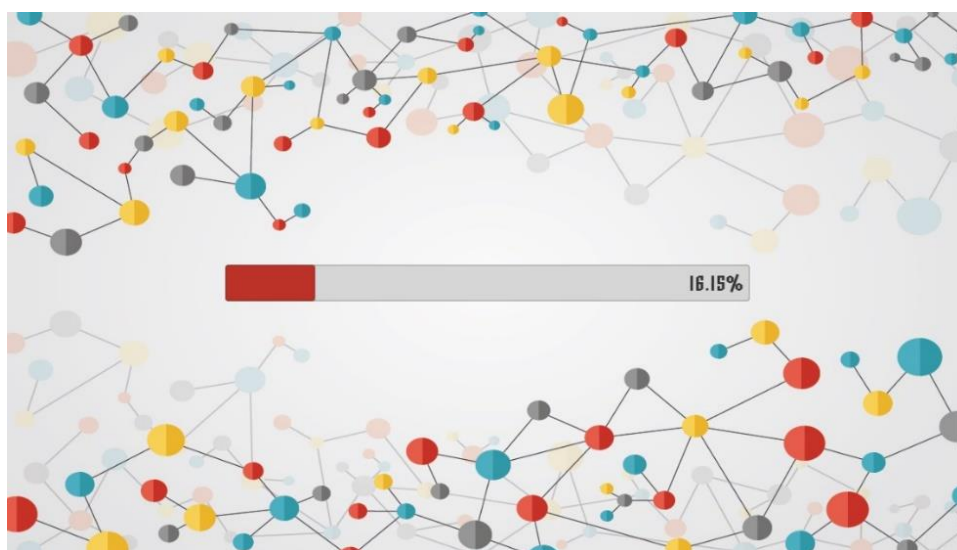
Após o login e a seleção da opção jogar, o jogador é encaminhado para a interface apresentada na Figura 54, onde nela está listada os estágios, em que para jogar o segundo estágio o jogador deverá ter concluído o primeiro. Após selecionar o estágio, aparecerá a barra de carregamento, conforme apresentado na Figura 55.

Figura 54 - Interface seleção de fase



FONTE: O AUTOR, 2018

Figura 55 - Interface tela de carregamento



FONTE: O AUTOR, 2018

A Figura 56 apresenta o cenário do jogo. No canto inferior esquerdo da Figura 56, estão 3 ícones: a engrenagem azul que serve para configurações do jogo, o inventário representado pela bolsa marrom que será para guardar os itens coletados durante o estágio e a caderneta que terá as missões do estágio, será o guia para o jogador.

Figura 56 - Interface do cenário



FONTE: O AUTOR, 2018

A Figura 57 mostra a aba de missões, a janela azul localizada ao lado esquerdo da tela é apresentada na tela quando o jogador seleciona a caderneta. O jogador deverá seguir esse passo a passo para concluir o estágio com sucesso. O passo a passo apresentado na Figura 57, corresponde ao estágio um.

Figura 57 - Interface aba de missão



FONTE: O AUTOR, 2018

Há momentos que será necessária uma comunicação com o jogador, e para isso, foram adotados balões representando a fala da personagem. Na Figura 58 há o exemplo de uma comunicação se caso o jogador tenta entrar com a personagem dentro do laboratório sem a vestimenta adequada.

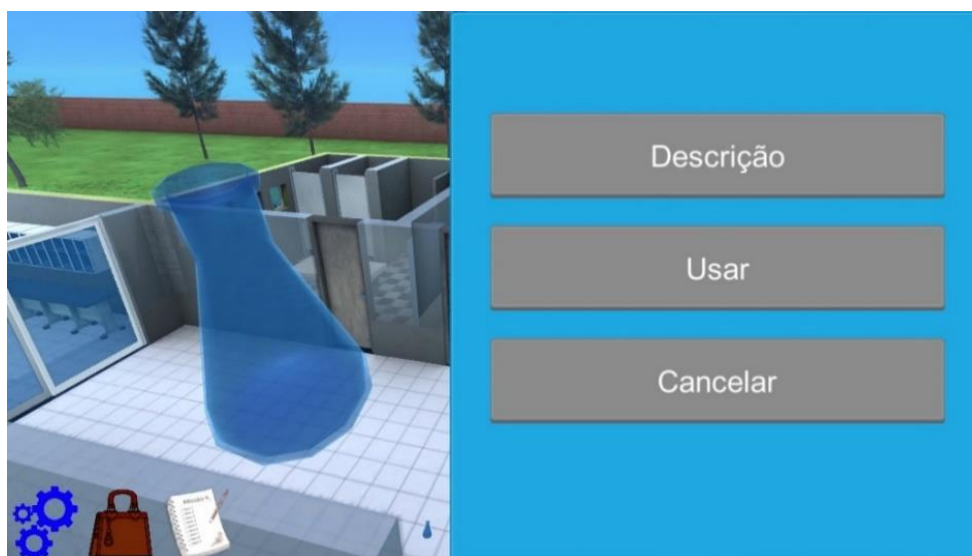
Figura 58 - Interface apresentando a comunicação personagem



FONTE: O AUTOR, 2018

Quando o jogador clica em um item do cenário, é aberta uma janela no lado direito da tela, conforma mostrado na Figura 59 com 3 possíveis ações: descrição, usar e cancelar. O objeto fica rotacionando no canto esquerdo.

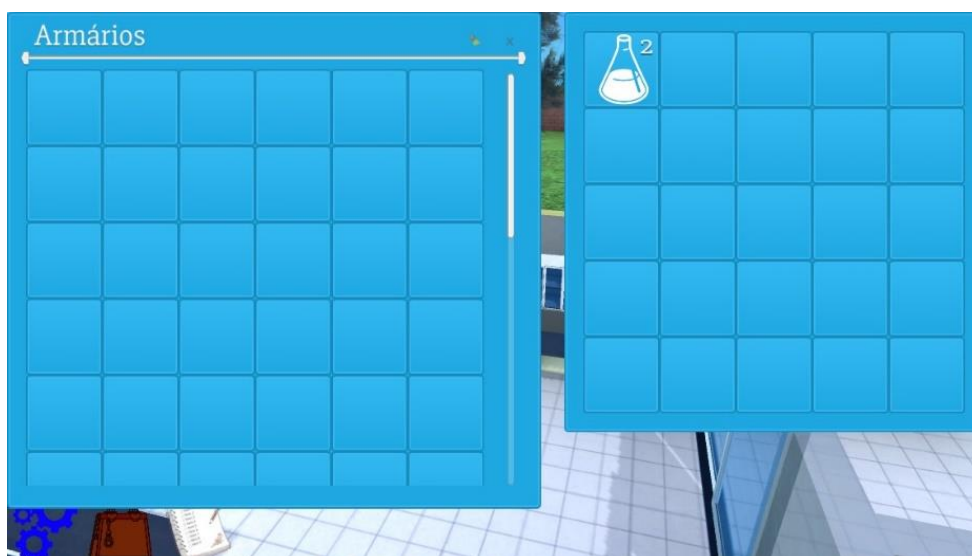
Figura 59 - Interface seleção de objeto



FONTE: O AUTOR, 2018

Quando o usuário selecionar o inventário, aparecerá um quadro localizado no canto esquerdo, com os itens e a respectiva quantidade. Na Figura 60 está a janela do inventário com dois *erlenmeyer*. O inventário funciona em conjunto com outros equipamentos, como o armário, apresentado na Figura 60, pois a pessoa realiza a transição de um local para o outro, podendo assim, retirar um item do armário e depois leva-lo ao agitador apresentado na Figura 61.

Figura 60 - Interface inventário



FONTE: O AUTOR, 2018

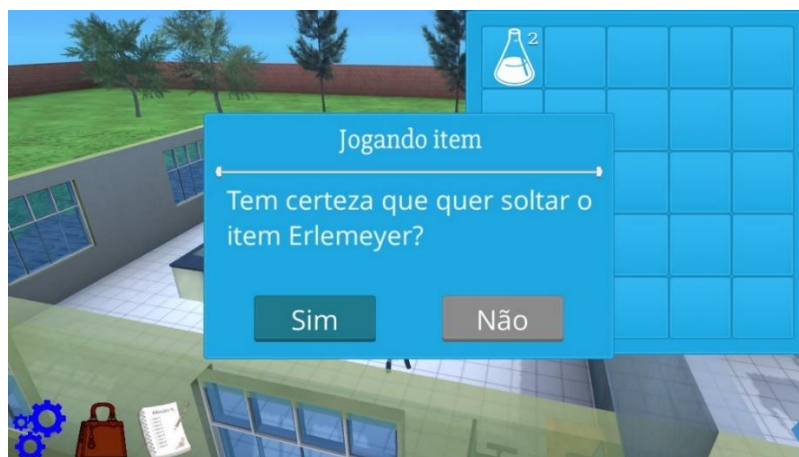
Figura 61 - Interface Agitador, local onde realiza as reações



FONTE: O AUTOR, 2018

O jogador tem a possibilidade de soltar o item para fora do inventário como apresentado na Figura 61.

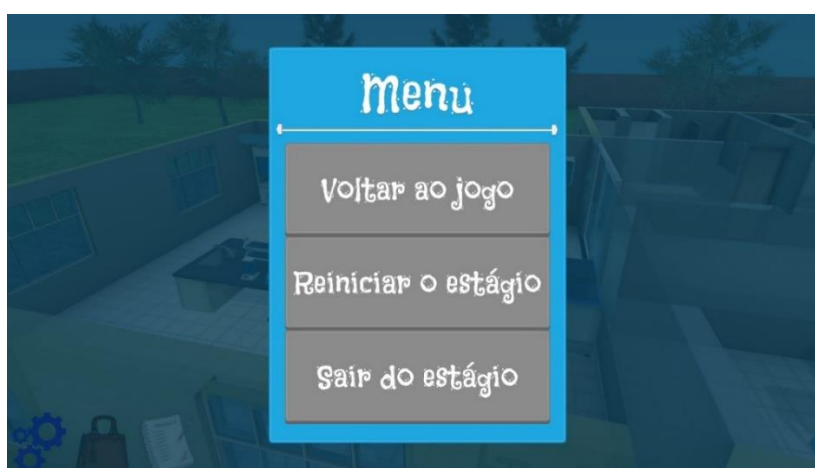
Figura 61: Interface modificações dentro do inventário



FONTE: O AUTOR, 2018

Caso o jogador pressione o ícone das engrenagens azul, ele será encaminhado para um menu, onde ele tem a opção de voltar ao jogo, reiniciar o estágio ou sair do estágio como apresentado na Figura 63.

Figura 63: Interface tela de *pause*



FONTE: O AUTOR, 2018

7 CONSIDERAÇÕES FINAIS

O presente trabalho de conclusão de curso propôs o desenvolvimento de um jogo que simule um laboratório de ciências, de modo a auxiliar o aprendizado da disciplina de química aos estudantes do ensino médio.

Com base nos estudos bibliográficos realizados e no decorrer do processo de desenvolvimento deste trabalho de conclusão de curso, foi encontrada uma alternativa de estudo facilitado na área de química, a fim de satisfazer a condição de que grande parte das escolas brasileiras não possuem um laboratório. Além de mostrar o método iterativo de desenvolvimento que se enquadrou na rotina de trabalho dos participantes deste trabalho.

No processo de desenvolvimento, foi concluído todo o cenário com a construção de todos os modelos 3D necessários, assim como as pinturas e animações e a conclusão da primeira fase.

Como trabalhos futuros pretende-se desenvolver um sistema de pontos de conquistas na qual o jogador poderá compartilhar seus feitos em redes sociais. Baseado na quantidade de pontos obtidos, o jogador poderá ter acesso a modelos de personagens, itens e fases diferentes.

Por fim, deseja-se que o jogo desenvolvido neste trabalho possa auxiliar na aprendizagem dos alunos do ensino médio de forma a representar um laboratório de ciências, visto que em trabalhos de futuros pretende implementar fases para a realização de reações de acordo com a ementa escolar.

8 REFERÊNCIAS

ALLEGORITHIMIC. **Substance Designer**. Disponível em: <<https://www.allegorithmic.com/products/substance-designer>> Acesso em: 27 nov. 2017.

BRASIL, Lei de Diretrizes e B. Lei nº 9.394/96, de 20 de dezembro de 1996.

FALKEMBACH, G. A. M. **O lúdico e os jogos educacionais**. Disponível em: <http://penta3.ufrgs.br/midiasedu/modulo13/etapa1/leituras/arquivos/Leitura_1.pdf> Acesso em: 10. Nov. 2017.

FIREBASE. **Firestore Authentication**. Disponível em: <<https://firebase.google.com/docs/auth/?hl=pt-br>>. Acesso em: 20 nov. 2017.

FULLERTON, Tracy; SWAIN, Chris; HOFFMAN, Steven. **Game design workshop: Designing, prototyping, & playtesting games**. CRC Press, 2004.

G1. **39% das escolas municipais de ensino fundamental têm bibliotecas; nas particulares, índice é de 82%, diz Censo. 2018**. Disponível em: <<https://g1.globo.com/educacao/noticia/39-das-escolas-municipais-de-ensino-fundamental-tem-bibliotecas-nas-particulares-indice-e-de-82-diz-censo.ghtml>>. Acesso em: 20 abr. 2018.

GUIMARÃES, E. S.; PARENTE, M. R. **Método para construção de modelos 3D em realidade virtual: um estudo de caso aplicado à mastologia**. 2008. 87 p. Monografia (Bacharel em Ciência da Computação) - Universidade Católica de Brasília, Brasília, 2008. Disponível em: <<https://repositorio.ucb.br/jspui/bitstream/123456789/9504/1/EduardoSchettiniGuimar%C3%A3esTCCGradua%C3%A7%C3%A3o2008.pdf>>. Acesso em: 15 nov. 2017.

LARSSON, B. F.; BERTOLI, L. **Desenvolvimento de concept art e modelagem 3D de um personagem de videogame**. 2014. 119 f. Trabalho de Conclusão de Curso (Graduação) - Universidade Tecnológica Federal do Paraná. Curitiba, 2014. Disponível em: <<http://repositorio.roca.utfpr.edu.br/jspui/handle/1/4964>>. Acesso em 20 out. 2017.

LECHETA, R. R. Introdução ao Android. In: LECHETA, R. R. **Android Essencial**. 1. ed. São Paulo-SP: Novatec Editora Ltda., 2016. cap. 1, p. 17-32. Disponível em: <<https://s3.novatec.com.br/capitulos/capitulo-9788575224793.pdf>>. Acesso em: 16 mar. 2018.

MARÇAL, E. ; RIBEIRO, J. W. ; LIMA, L. ; VIANA, W. ; ANDRADE, R.M. . **A Utilização de Dispositivos Móveis com Ambientes Tridimensionais como Ferramenta para Favorecer o Ensino de Hardware**. In: Simpósio Brasileiro de Informática na Educação, 2009, Florianópolis, SC. **Anais do XX SBIE: Simpósio Brasileiro de Informática na Educação (CD-ROM)**, 2009. Disponível em: <<http://www.br-ie.org/pub/index.php/sbie/article/view/1146/1049> > Acesso em: 25 nov. 2017.

POPOLIN NETO, M.; AGOSTINHO, I. A.; DIAS, D. R. C.; RODELLO, I. A.; BREGA, J. R. F. A realidade virtual e o motor de jogo unity. **Tendências e Técnicas em Realidade Virtual e Aumentada**[S.l: s.n.], 2015. Disponível em: <https://www.researchgate.net/publication/277598033_A_Realidade_Virtual_e_o_Motor_de_Jogo_Unity> Acesso em: 20 nov. 2017.

ROGERS, Scott. **Level Up! The guide to great video game design**. John Wiley & Sons, 2014.

SANTOS, Barbara Ferreira; RIBEIRO, Marcelo. **Brasil está entre os piores em ranking mundial de educação**. 2016. Disponível em: <<https://exame.abril.com.br/brasil/brasil-esta-entre-os-8-piores-em-ciencias-em-ranking-de-educacao/>>. Acesso em: 20 abr. 2018.

SCHELL, Jesse. **The Art of Game Design: A book of lenses**. AK Peters/CRC Press, 2014.

TAROUCO, L.M.R.; COSTA, V. M.; ÁVILA, B. G.; BEZ, M. R.; SANTOS, E. **Objetos de Aprendizagem: teoria e prática**. 1. ed. Porto Alegre: Evangraf, 2014. v. 1. 501p. Disponível em: <<http://penta3.ufrgs.br/ObjetosAprendizagem/LivroOA-total.pdf>>