

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
DE MINAS GERAIS – *CAMPUS BAMBUÍ-MG*
BACHARELADO EM ENGENHARIA DE COMPUTAÇÃO

Jorge Luís Vieira Murilo

**O USO DA PLATAFORMA ROBLOX PARA A CRIAÇÃO DE OBJETOS
VIRTUAIS DE APRENDIZAGEM: um estudo por meio do desenvolvimento de
experiências e elaboração de tutorial**

**BAMBUÍ-MG
2024**

Jorge Luís Vieira Murilo

**O USO DA PLATAFORMA ROBLOX PARA A CRIAÇÃO DE OBJETOS
VIRTUAIS DE APRENDIZAGEM: um estudo por meio do desenvolvimento de
experiências e elaboração de tutorial**

Trabalho de conclusão de curso apresentado
ao Curso Bacharelado em Engenharia de
Computação do Instituto Federal de Minas
Gerais - *Campus* Bambuí como requisito
parcial para obtenção do grau de Bacharel em
Engenharia de Computação.
Orientador: Gabriel da Silva

**BAMBUÍ-MG
2024**

Catálogo na Fonte Biblioteca IFMG – Campus Bambuí

M977U Murilo, Jorge Luís Vieira.
O uso da plataforma Roblox para a criação de objetos virtuais de aprendizagem: um estudo por meio do desenvolvimento de experiências e elaboração de tutorial [manuscrito] / Jorge Luís Vieira Murilo. – 2024.
62 f. : il. : color.

Orientador: Gabriel da Silva
Trabalho de Conclusão de Curso (Bacharelado em Engenharia de Computação) – Instituto Federal de Minas Gerais. Campus Bambuí, 2024.

1. Objetos virtuais de aprendizagem. 2. Roblox. 3. Novas metodologias de ensino. 4. Tutorial. I. Silva, Gabriel da. II. Instituto Federal de Minas Gerais. Campus Bambuí. III. Título.

CDD-371.35

Jorge Luís Vieira Murilo

O USO DA PLATAFORMA ROBLOX PARA A CRIAÇÃO DE OBJETOS VIRTUAIS DE APRENDIZAGEM: um estudo por meio do desenvolvimento de experiências e elaboração de tutorial

Trabalho de conclusão de curso apresentado ao Curso Bacharelado em Engenharia de Computação do Instituto Federal de Minas Gerais - *Campus* Bambuí como requisito parcial para obtenção do grau de Bacharel em Engenharia de Computação.

Aprovado em 09/04/2024



Documento assinado eletronicamente por **Gabriel da Silva, Professor**, em 09/04/2024, às 10:42, conforme Decreto nº 10.543, de 13 de novembro de 2020.



Documento assinado eletronicamente por **Claudio Ribeiro de Sousa, Professor EBTT**, em 09/04/2024, às 10:43, conforme Decreto nº 10.543, de 13 de novembro de 2020.



Documento assinado eletronicamente por **Felipe Lopes de Melo Faria, Professor**, em 09/04/2024, às 10:44, conforme Decreto nº 10.543, de 13 de novembro de 2020.



A autenticidade do documento pode ser conferida no site <https://sei.ifmg.edu.br/consultadocs> informando o código verificador **1896776** e o código CRC **620009DB**.

BAMBUÍ-MG
2024

*A todos aqueles que me deram apoio durante minha
caminhada estudantil.*

DEDICAMOS

AGRADECIMENTOS

A todos os professores que me acompanharam durante minha jornada estudantil, em especial ao Orientador desse trabalho de conclusão de curso. A todos os servidores e técnicos administrativos que participaram de alguma forma da minha formação acadêmica e a todos os colegas que concluíram essa jornada comigo. Em especial à minha família que sempre me apoiou em todos os momentos da minha vida e todos que ajudaram indiretamente na minha formação.

“A experiência é uma vela que ilumina apenas a quem a conduz”.
(Oscar Wilde)

RESUMO

No âmbito desta pesquisa, realizou-se um estudo sobre a viabilidade de se utilizar a plataforma Roblox na criação de Objetos Virtuais de Aprendizagem (OVA), cujo o intuito foi investigar a possibilidade de aplicação dessa ferramenta por professores sem experiência em programação, visando assim, expandir o uso de OVAs no ambiente educacional. Para alcançar este objetivo, optou-se por empregar a metodologia Design Science Research (DSR), que se mostrou adequada para definir o escopo de construção de dois artefatos: um tutorial destinado a professores leigos em programação e os próprios OVAs gerados a partir deste tutorial. Após a elaboração destes artefatos e aprofundada a exploração da plataforma, chegou-se à conclusão de que, embora o Roblox apresente um ambiente intuitivo para criação, ainda assim, é necessário possuir conhecimento prévio em programação para enfrentar os desafios inerentes à construção das experiências propostas. Consequentemente, a plataforma Roblox pode ser considerada uma opção viável para a construção de OVAs, desde que o usuário tenha um entendimento básico de programação para lidar com a criação de scripts. Este estudo evidencia a importância do contínuo desenvolvimento de ferramentas educacionais inovadoras, bem como da necessidade de preparo técnico dos docentes para a utilização eficaz de tais recursos no processo de ensino-aprendizagem.

Palavras Chave: Objetos Virtuais de Aprendizagem. Roblox. Novas Metodologias de Ensino. Tutorial.

ABSTRACT

As part of this research, was conducted a study on the feasibility of using the Roblox platform to create Learning Virtual Objects (LVOs). The aim was to investigate the potential application of this tool by teachers without programming experience, with the goal of expanding the use of LVOs in the educational environment. To achieve this objective, was used the Design Science Research (DSR) methodology, which proved suitable for defining the scope of construction of two artifacts: a tutorial for non-programming teachers and the LVOs themselves generated from this tutorial. Following the creation of these artifacts and an in-depth exploration of the platform, it is concluded that while Roblox provides an intuitive environment for creation, it is still necessary to have prior programming knowledge to address the challenges inherent in constructing the proposed experiences. Consequently, the Roblox platform can be considered a viable option for creating LVOs, provided that the user has a basic understanding of programming to manage the creation of scripts. This study highlights the importance of continuous development of innovative educational tools, as well as the need for technical preparation of teachers to effectively use such resources in the teaching-learning process.

Key words: Virtual Learning Objects. Roblox. New Teaching Methodologies. Tutorial.

LISTA DE FIGURAS

Figura 1 – Experiência envolvendo matemática disponível no Roblox.....	17
Figura 2 – Funcionalidade do H5P.....	20
Figura 3 – Mapa de elementos DSR.....	25
Figura 4 – OVA Mudanças de Estados Físicos utilizando H5P.....	30
Figura 5 – Versão inicial do OVA 2D construído no Roblox.....	30
Figura 6 – Versão Final do OVA 2D construído no Roblox.....	31
Figura 7 – Bloco de interação do OVA 3D.....	32
Figura 8 – Bloco de interação do OVA 3D.....	33
Figura 9 – Quadro de Transformações do OVA 3D.....	33
Figura 10 – Verificação de respostas do OVA 3D.....	34
Figura 11 – Tela do Roblox Studio.....	35

LISTA DE ABREVIATURAS E SIGLAS

OVA	Objetos Virtuais de Aprendizagem
UGC	Conteúdo Gerado pelo Usuário
DSR	<i>Design Science Research</i>
DSRM	<i>Design Science Research Methodology</i>

SUMÁRIO

1	APRESENTAÇÃO.....	12
1.1	Objetivos.....	12
1.1.1	Objetivo Geral.....	13
1.1.2	Objetivos Específicos.....	13
1.1.3	Resultados Esperados.....	13
1.2	Justificativa.....	13
2	REFERENCIAL TEÓRICO.....	15
2.1	Roblox.....	15
2.2	A evolução da Sala de aula.....	16
2.3	Design Science Research Methodology.....	17
2.4	H5P.....	18
2.5	Estado da Arte.....	19
3	METODOLOGIA.....	22
3.1	Classificação do Trabalho.....	22
3.2	Solução Proposta.....	22
3.3	Materiais e Método.....	23
4	RESULTADOS E DISCUSSÃO.....	28
5	CONCLUSÃO.....	36
	REFERENCIAS.....	38
	APÊNDICE A.....	42

1 APRESENTAÇÃO

Objetos Virtuais de Aprendizagem (OVAs) constituem recursos didáticos que apoiam a prática pedagógica dentro e fora de sala de aula, fazendo uso da tecnologia como um complemento ao ensino. Dentre eles, estão: jogos, animações, simuladores e videoaulas.

Segundo Salvador *et al.* (2017), o processo de ensino personalizado é autônomo promovido por OVA, é um potencializador de uma aprendizagem segura e apoiada pela tecnologia, a qual impacta diretamente na eficácia educacional. Sendo assim sua utilização é capaz de tornar o processo de ensino-aprendizagem mais interativo, estimulando a imaginação, a criatividade e o interesse dos alunos no ensino, trazendo inúmeros benefícios ao processo e beneficiando uma grande variedade de disciplinas.

As rápidas mudanças na tecnologia tendem a influenciar o modo como a sociedade educa as pessoas. Compreender como essa tecnologia pode ser distribuída é necessário, para definir como a comunicação é estabelecida. Dessa forma, a ascensão das diversas formas de comunicação, como redes sociais, plataformas de streaming, jogos e músicas, oferece uma gama de oportunidades para a educação da nova geração, conforme Long (2020).

Long (2020) afirma ainda que diferentes estratégias precisam ser consideradas para se utilizar a tecnologia na educação. Dentre elas, a motivação é um fator de decisão para que o estudante participe de forma adequada do processo de ensino-aprendizagem.

O Roblox, uma plataforma de Conteúdo Gerado pelo Usuário (CGU), mostra-se como uma ferramenta interessante para apoiar os profissionais da educação na construção de um modelo de ensino lúdico e efetivo. Essa plataforma possui duas modalidades de utilização: uma que possibilita a construção de experiências personalizadas, nomeada Roblox Studio, e outra que permite a utilização de experiências construídas por outros desenvolvedores, o Roblox Player. Diariamente, dezenas de milhões de alunos acessam o Roblox para aprender, explorar e compartilhar experiências em mundos 3D imersivos (ROBLOX, 2023).

1.1 Objetivos

A definição de objetivos constitui uma parte fundamental na delimitação do escopo do trabalho. Sendo assim, esses objetivos podem ser observados nas subseções 1.1.1 e 1.1.2, que correspondem, respectivamente, aos Objetivos Geral e Específicos do presente estudo.

1.1.1 Objetivo Geral

Realizar um estudo acerca da viabilidade da utilização da plataforma Roblox na construção e utilização de Objetos Virtuais de Aprendizagem.

1.1.2 Objetivos Específicos

- Caracterização do estado da arte da utilização do Roblox na educação;
- Construção de um tutorial para construção de OVAs genéricos;
- Realização de uma prova de conceito, construindo-se um OVA baseado no tutorial a ser gerado por este trabalho.

1.1.3 Resultados Esperados

Acredita-se que, ao fazer uso das oportunidades geradas pela plataforma Roblox, é possível explorar uma nova metodologia de ensino para diversas áreas do conhecimento. Essas oportunidades proporcionaram um ônus no modelo de ensino-aprendizagem adotado atualmente, resultando em benefícios substanciais para diversas áreas do conhecimento.

Espera-se, com a conclusão deste trabalho, que o tutorial desenvolvido auxilie na criação de Objetos Virtuais de Aprendizagem (OVAs) utilizando a plataforma Roblox, acessível e útil para professores sem conhecimentos avançados em programação, a fim de apoiar sua metodologia de ensino. Além disso, espera-se que um OVA seja construído seguindo-se o tutorial desenvolvido, servindo como validação da sua aplicabilidade e constituindo evidências da eficácia do artefato produzido.

1.2 Justificativa

As tecnologias emergentes têm impacto direto no cotidiano da população, especialmente nas novas gerações, que estão cada vez mais envolvidas no mundo da tecnologia. De acordo com Pizzol et al. (2022), a população conhecida como nativos digitais nasceu imersa no contexto tecnológico, o que afeta sua maneira de pensar e realizar ações, uma vez que a tecnologia se torna parte integrante de suas atividades diárias.

Essa nova forma de pensar e agir deve ser estimulada pelos professores, não apenas ao utilizarem jogos para implementar uma metodologia de ensino diferente do modelo convencional, mas também ao fazerem uso de uma ampla variedade de recursos tecnológicos disponíveis. Galafassi (2013) afirma que o uso das Tecnologias de Informação e Comunicação (TIC) oferece oportunidades para mudanças nos paradigmas relacionados à

aprendizagem, o que influencia diretamente o processo de ensino-aprendizagem.

A documentação disponível na plataforma Roblox para educadores auxilia na utilização de Objetos Virtuais de Aprendizagem (OVAs) previamente construídos. No entanto, é importante ressaltar que a documentação atual não apresenta uma distinção clara entre desenvolvedores e professores, apesar de existir uma seção exclusiva para educadores no website da plataforma.

A falta de distinção da documentação pode gerar confusão para pessoas que não possuem conhecimento em programação, dificultando a compreensão de certos termos e sua aplicação prática, podendo até mesmo desmotivar o profissional na criação de suas próprias experiências educacionais na plataforma.

Com base nessas constatações, propõe-se a utilização da plataforma Roblox como meio para a construção de Objetos Virtuais de Aprendizagem (OVAs), a fim de apoiar os professores no contexto educacional. Isso tornará o processo de ensino-aprendizagem mais descontraído e interativo, permitindo a inclusão de novas experiências no modelo convencional de ensino e aproveitando as tecnologias com as quais os estudantes estão familiarizados.

Abrahamsen (2022) menciona que a familiaridade dos estudantes com a plataforma Roblox pode ajudar a reduzir a carga cognitiva necessária em suas atividades educacionais, facilitando sua rotina na sala de aula.

É importante destacar que a área de ajuda da ferramenta Roblox Studio possui apenas redirecionamentos para a documentação presente no website da plataforma. A ausência de um menu de ajuda específico dentro do Roblox Studio dificulta o processo de construção de experiências, o que pode inviabilizar a construção de OVA por aqueles que não possuem conhecimento prévio acerca da utilização dessa ferramenta.

2 REFERENCIAL TEÓRICO

A plataforma Roblox se popularizou nos últimos anos, apresentando grande crescimento e diversificação de seus usuários, fato que possibilita a expansão de suas formas de utilização.

2.1 Roblox

A plataforma Roblox, fundada por David Baszucki, visa permitir o compartilhamento de experiências entre usuários. Ela fornece oportunidades para a criação de conteúdos que vão desde um acessório simples para um personagem ou um avatar até experiências imersivas 3D (ROBLOX, 2023).

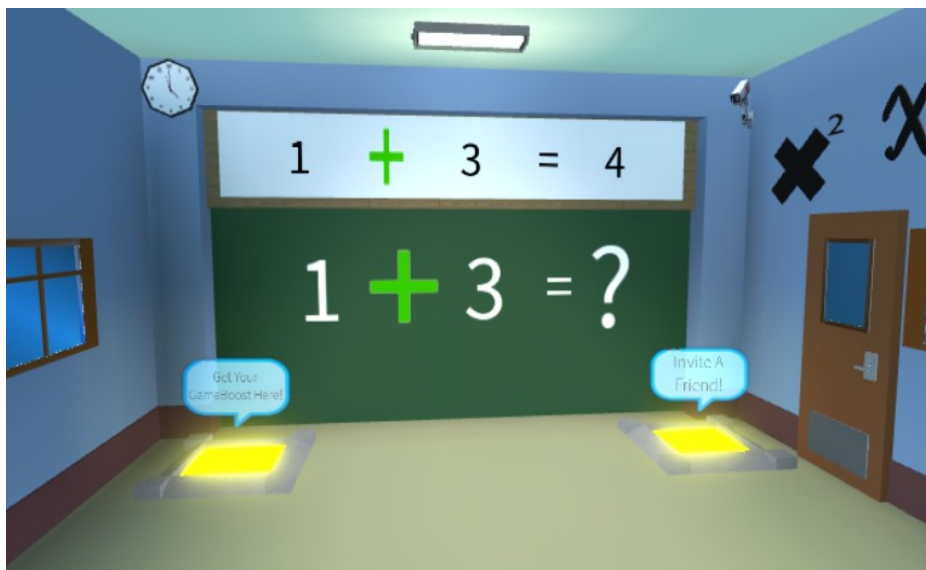
Há duas possibilidades de utilização do Roblox:

1. O Roblox Player, que possibilita a interação com as experiências disponíveis no website da plataforma.
2. O Roblox Studio, utilizado para desenvolvimento e personalização de experiências, as quais podem ser disponibilizadas na plataforma posteriormente.

Ambas as maneiras de utilização podem ser aplicadas como novas metodologias de ensino e necessitam da criação prévia de uma conta no Roblox. Este trabalho tem foco na segunda forma de utilização do Roblox, visto que ela é capaz de proporcionar liberdade para a construção de experiências personalizadas, fornecendo ao professor ferramentas para criação de suas próprias metodologias.

A plataforma Roblox é um ambiente que oferece oportunidades para a criação de conteúdo com disponibilização simplificada, sendo possível, através dela, obter monetização pelas experiências desenvolvidas, em que será cobrado um percentual quando ela fornece lucro ao desenvolvedor. Além disso, a plataforma disponibiliza ferramentas, serviços e suporte de forma gratuita, podendo-se oportunizar e utilizar experiências gratuitamente (ROBLOX, 2023).

Figura 1 – Experiência envolvendo matemática disponível no Roblox



Fonte: Roblox, 2023.

O Roblox dispensa taxas de hospedagem, armazenamento, suporte ao cliente, localização, requisitos regulatórios, processamento de pagamentos e taxas de plataforma. Além disso, possibilita a publicação em consoles, desktops e dispositivos, permitindo o acesso em diversos locais do globo e possibilitando a interação multiplayer (ROBLOX, 2023).

A plataforma atingiu 1,7 bilhão de usuários no mês de outubro de 2022, demonstrando um aumento de 16% se comparado com o mesmo período do ano anterior. Ela possuía uma média de 57,8 milhões de usuários ativos todos os dias (ADRENALINE, 2023). Atualmente, no ano de 2023, a plataforma conta com 66,1 milhões de usuários ativos todos os dias e 66,4 milhões de experiências, totalizando 14,5 bilhões de horas de engajamento com os seus usuários (ROBLOX, 2023).

2.2 A evolução da Sala de aula

Segundo a Fundação Lemann (2023), durante o período de março de 2020 a agosto de 2021, cerca de 180 mil escolas tiveram suas aulas presenciais interrompidas, fato ocasionado pela pandemia da COVID-19. Deste modo, o ensino remoto se tornou a única opção para os 47 milhões de estudantes brasileiros. Esse acontecimento mudou a realidade dos estudantes que passaram por esse período, e, mesmo com o esforço dos professores, a educação brasileira pode retroceder até quatro anos.

Para suprir a defasagem no ensino, causada pela pandemia, e implementar

efetivamente novas metodologias de ensino, é necessário aprimorar o modelo convencional, incorporando práticas de interação com os alunos e abordagens diferenciadas na apresentação dos conteúdos.

A evolução está sempre presente e o metaverso também. Avatares, realidade virtual e aumentada, criptomoedas, gamers, Roblox, além de Tiktokers e Youtubers são partes dessa geração. Os estudantes vivem e respiram tecnologia diariamente. Se os meios de comunicação evoluíram, a sala de aula não pode ficar para trás, estática, copiando modelo do início do século passado (PEREIRA, 2022)

A plataforma Roblox surge como uma alternativa para atenuar a defasagem no ensino, pois proporciona aos alunos a oportunidade de interagir diretamente com as experiências e assimilar o conteúdo de forma interativa, tornando-os protagonistas de seu próprio aprendizado.

2.3 Design Science Research Methodology

Segundo Peffers et al.(2007), o objetivo desta metodologia é fornecer um modelo mental para as características dos resultados da pesquisa, propiciando embasamento para a realização de pesquisas para Ciências do Artificial. Peffers et al.(2007) afirmam que existem seis atividades que norteiam o modelo de desenvolvimento:

- Atividade 1: Identificação do problema e motivação: definir o problema específico de pesquisa e justificar o valor de uma solução.
- Atividade 2: Definir os objetivos para uma solução: inferir os objetivos de uma solução a partir da definição do problema e do conhecimento do que é possível e viável.
- Atividade 3: *Design* e desenvolvimento: criar o artefato.
- Atividade 4: Demonstração: demonstrar o uso do artefato para resolver uma ou mais instâncias do problema.
- Atividade 5: Avaliação: observar e medir o quão bem o artefato suporta uma solução para o problema.
- Atividade 6: Comunicação: comunicar o problema e sua importância, o artefato, sua

utilidade e novidade, o rigor de seu *design* e sua efetividade para pesquisadores e outros públicos relevantes, como profissionais em exercício, quando apropriado.

Esse método se mostra como uma proposta interessante para o presente trabalho, vistas as atividades propostas por Peffers et al.(2007), as quais auxiliarão na delimitação dos passos a serem efetuados, bem como o referencial teórico necessário para isso.

2.4 H5P

O H5P trata-se de ferramenta que facilita a criação, o compartilhamento e a reutilização de conteúdos e aplicativos baseados em HTML5. Sua função primordial é capacitar indivíduos, independentemente do nível de proficiência técnica, a desenvolver experiências web interativas de maneira eficiente. Essa plataforma requer apenas um navegador e a implementação do plugin H5P no website para utilizar seus recursos (H5P, 2023).

Essa ferramenta se distingue por sua capacidade de expandir as fronteiras da interatividade online, oferecendo uma gama de recursos que enriquecem o panorama digital, contribuindo para a experiência do usuário e a qualidade do conteúdo web. Essa ferramenta dispõe de diversas funcionalidades, destacando-se, para esta proposta, a drag and drop, traduzido literalmente como “arrastar e soltar”, onde é possível arrastar rótulos textuais ou imagens para locais previamente definidos. A Figura 2 exibe algumas das funcionalidades disponíveis no H5P.

Figura 2 – Funcionalidade do H5P



Palavras cruzadas

Crie palavras cruzadas



Cartões de diálogo

Criar cartões giratórios baseados em texto



Ferramenta de Documentação

Criar um assistente de formulário com exportação de texto



Arrastar e soltar

Criar tarefas de arrastar e soltar com imagens

Fonte: O autor, 2023.

2.5 Estado da Arte

O Roblox é uma plataforma gratuita, atualmente dominada por crianças de 5 a 12 anos que criam personagens virtuais para interagirem entre si (SHARON MISTRETTA, 2022). Essas experiências podem ser utilizadas de diversas formas na sala de aula. Segundo Pizzol et al. (2022, p. 82):

Entre sua principal descrição, evita-se nomear a palavra jogo em seu léxico de produtos, mas sim experiências, pois reconhece que seus produtos não tem como foco apenas em jogos em si, mas diversos tipos de atividades, mesmo que em sua maioria sejam lúdica, existem experiências científicas, de programação, de controle de Inteligência Artificial, de salas de ensino etc. Logo, compreender as atividades que o Roblox promove como uma experiência e não apenas como um mero jogo, corresponde às potencialidades enquanto ferramenta criativa e inovadora.

Essa plataforma também pode ser utilizada como fonte de consulta para a proposta de atividades que envolvam experiências de gamificação, assim como no caso do estudo realizado por Brugnago et al. (2021), em que foi efetuado um levantamento de quais plataformas ou jogos poderiam fornecer insumos suficientes para realizar um estudo de Física. Neste, o Driving Simulator, recurso do Roblox, foi empregado como fonte de aquisição de dados para o experimento, revelando uma das aplicações que as experiências da plataforma podem proporcionar.

Outro exemplo de aplicação do Roblox na sala de aula é apresentado por Siqueira

(2019) no ensino de Geografia Física, evidenciando que há possibilidades de se alterar as funcionalidades do território da experiência, bem como a maneira como os indivíduos se organizam no espaço.

Siqueira (2019) utilizou o mundo fictício intitulado *Natural Disaster Survival*, que simula a atuação de fenômenos naturais, como furacões, tempestades severas, tornados, terremotos, tsunamis, entre outros. Essa experiência estimula a sobrevivência do jogador aos impactos apresentados, recompensando-o com pontos, para que se mantenha resiliente às situações futuras.

O Roblox Studio também pode ser utilizado para construir experiências a fim de promover inovação e ludicidade no ensino. Esse é o caso do projeto Música e Tecnologia, proposto por Da Cruz, Hofstätter e Pires (2022), em que foi desenvolvida uma proposta de construção de materiais audiovisuais utilizando-se o Roblox, como alternativa de lecionar conceitos relevantes do conteúdo de Artes.

Pires et al.(2022), em um trabalho correlato, afirmam que foi desenvolvida uma experiência que apresenta desafios de perguntas e respostas sobre os conceitos de música em que estudantes do Curso Técnico em Informática Integrado ao Ensino Médio interagiram com a experiência construída através de minijogos de perguntas e respostas, baseadas em um conceito musical disponível em forma de textos, imagens e sons.

O uso da plataforma tem se popularizado nos últimos anos, e novas oportunidades de sua aplicação vem se tornando concretas. Porém, ainda há pouca pesquisa científica envolvendo o assunto, em comparação com assuntos similares.

Hernández et al. (2022, p. 11) afirmam que, em uma pesquisa realizada com o Google Acadêmico no ano de 2021, ao se utilizar as palavras “Minecraft/Education”, era possível encontrar 67 artigos acadêmicos com alguma delas em seu título; porém, utilizando-se as palavras “Roblox/Education”, foi encontrado apenas um trabalho com essas palavras.

A diferença apontada por Hernández et al. (2022, p. 11) ainda se mantém no ano de 2023, em que, ao utilizar a mesma forma de busca, foi possível encontrar 1590 resultados para trabalhos que utilizam a forma “Minecraft/Education” em seu título e apenas 23 que fazem uso de “Roblox/Education”.

Houve um avanço entre os anos de 2021 e 2023 ao se comparar os resultados apresentados. No entanto, essa diferença demonstra que a plataforma Roblox ainda possui um potencial de exploração no contexto educacional, sendo que, em comparação com o outro assunto, houve uma produção menor de trabalhos.

É visível o impacto que essa ferramenta pode proporcionar ao ensino, proporcionando oportunidades para a inovação na sala de aula, promovendo maior interesse e interação dos alunos para com as atividades propostas, agregando conhecimento e tornando-os mais receptivos a novos modelos de ensino.

A presente proposta diferencia-se dos trabalhos apresentados pelo fato de ser planejada a elaboração de um tutorial que auxilie professores leigos em programação na construção de suas próprias experiências. Os trabalhos apresentados demonstram, em sua maioria, resultados de experimentos utilizando experiências prontas do Roblox, o que vai na contramão do que é proposto por este trabalho.

3 METODOLOGIA

Nesta seção, é apresentado como se deu o desenvolvimento do presente trabalho, evidenciando os passos metodológicos necessários para se alcançar o objetivo geral, bem como as ferramentas, objetos e tecnologias utilizadas.

3.1 Classificação do Trabalho

A presente proposta trata-se de uma pesquisa aplicada, a fim de desenvolver um tutorial para criação de objetos virtuais de aprendizagem utilizando-se a plataforma Roblox. Para isso, efetuou-se uma pesquisa exploratória na qual um tutorial foi construído.

Quanto aos procedimentos, empregou-se a técnica Design Science Research (DSR), com o intuito de construir dois artefatos, um tutorial para professores leigos em programação e um OVA gerado a partir do tutorial.

A pesquisa enquadra-se em uma abordagem qualitativa, visto que não foi considerada sua representatividade numérica, mas sim sua proposta em analisar a funcionalidade do Roblox, por meio de um conjunto de testes de caixa-preta, bem como a completude do tutorial, o qual deve ser capaz de gerar um OVA funcional, seguindo-se todos os passos nele evidenciados. Segundo Wazlawick (2009), esta pesquisa se caracteriza como uma apresentação de Algo Diferente, visto que busca sustentar a hipótese levantada e propor a criação de um passo a passo para a construção de OVA por meio da plataforma Roblox.

3.2 Solução Proposta

Como visto na sessão 1.2, a falta de uma documentação que direcione o professor na construção de suas próprias experiências mostra-se como um dificultador do processo de inovação no ensino-aprendizagem, uma vez que educadores podem não possuir familiaridade com o teor técnico exigido pela documentação do Roblox.

Educadores sem conhecimento prévio podem se sentir perdidos e desencorajados a utilizar a plataforma. Sendo assim, a construção de um tutorial para construção de OVA, utilizando a plataforma Roblox, apresenta-se como uma alternativa para promover a liberdade do docente na inovação de sua metodologia de ensino.

Para que o tutorial pudesse ser construído, propôs-se a utilização da DSR, a qual foi empregada como um meio de delimitar o escopo dos artefatos. Espera-se, com a sua

utilização, elaborar um tutorial que torne mais prática a construção de OVA pelos professores, aproximando o teor teórico da documentação oferecida pela plataforma da construção prática desses objetos. Segundo a DSR, o OVA gerado pelo tutorial classifica-se como uma instância, e este último, por sua vez, como um método.

Dentre as ações que podem ser apoiadas pela DSR, está a simplificação de conteúdos, a qual foi utilizada na presente proposta, na sintetização e adaptação do conteúdo disponível na documentação da plataforma Roblox, para geração de um tutorial focado na construção de objetos virtuais de aprendizagem.

3.3 Materiais e Método

Segundo Pimentel, Filippo e Santoro (2020, p.22), “A Design Science Research é uma abordagem em que, além de produzir ciência sobre a realidade, objetiva-se projetar uma realidade diferente, modificada por artefatos projetados para resolver problemas em determinados contextos”. Para que isso possa ser efetivamente realizado nesta proposta, as seguintes etapas, com suas respectivas tecnologias, foram realizadas:

- Revisão bibliográfica dos temas: Viabilidade de utilização do Roblox como ferramenta de ensino-aprendizagem e material disponível para esse fim, utilização do Roblox Studio na construção de OVA, OVA disponíveis para aplicação em sala de aula;
- Caracterização do estado da arte para utilização do Roblox como ferramenta aliada ao ensino-aprendizagem;
- Replicação de experiências para compreender o funcionamento do Roblox Studio, suas principais funcionalidades, a linguagem que ele utiliza e maneiras de construção, disponíveis no *website* do Roblox;
- Levantamento dos passos essenciais para construção de OVA utilizando-se o Roblox Studio;
- Elaboração de Tutorial para construção de OVA utilizando o Roblox Studio;
- Modelagem de OVAs para prova de conceito;
- Disponibilização de OVAs construído.

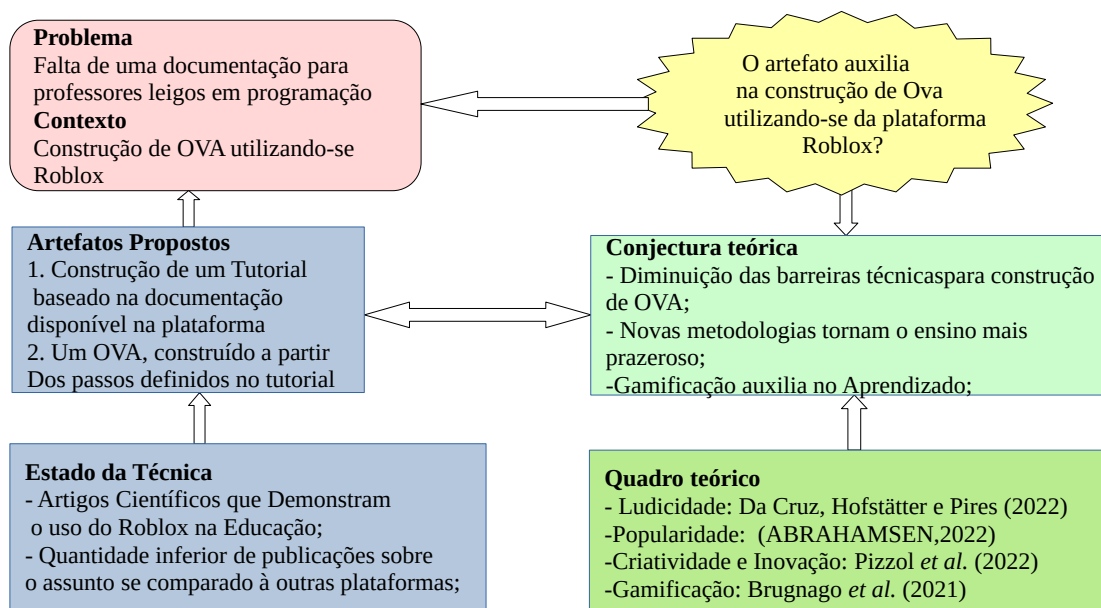
Para que os passos previamente apresentados fossem realizados de maneira

adequada, o mapa evidenciado na Figura 3 foi construído visando orientar o ciclo de investigação do problema, o qual contempla elementos fundamentais para aplicação da DSR neste estudo. Esses elementos são a identificação do problema, o contexto no qual ele está incluso, os artefatos propostos para a resolução do problema, o estado da técnica que fundamenta a construção desses artefatos, o quadro teórico para apoiar a conjectura teórica e a validação da conjectura.

Em DSR, é preciso atentar para dois ciclos que se inter-relacionam: um sobre projeto do artefato(ciclo de *Design*, ou ciclo de Engenharia) com o objetivo de resolver o problema real em um determinado contexto; e outro sobre teorias científicas relacionadas ao comportamento humano ou organizacional(Ciclo de Conhecimento, ou ciclo do Rigor). As teorias científicas subsidiam o projeto do artefato; e o uso do artefato, por sua vez, possibilita investigar as conjecturas teóricas que subjazem o artefato (DA SILVA, 2016, p. 23)

Os ciclos apresentados por Da Silva (2016) demonstram uma estrita correlação, pois, para construir um artefato, é necessário fundamentar o conhecimento disponível acerca dessa proposta, e, de maneira semelhante, utilizar-se do artefato construído para verificar se as conjecturas propostas foram satisfeitas. Nesta proposta, a fundamentação do conhecimento é apresentada na seção 2 , intitulada Referencial Teórico, e a verificação da satisfação das conjecturas propostas, na seção 4 , intitulada Resultados e Discussão.

Figura 3 – Mapa de elementos DSR



Fonte: O autor, 2023.

A instanciação possui um caráter facilitador, promovendo liberdade na construção de

novas experiências e direcionamento para que os passos necessários para construção de experiências personalizadas sejam construídos.

O método DSRM e suas atividades foram utilizadas como meio de se alcançar o que é proposto no mapa dos elementos (Figura 3). Para validar os artefatos, foram desenvolvidos dois OVAs, utilizados como critérios de classificação da adequação do artefato à solução proposta.

Ao se satisfazer a validação proposta, é possível afirmar que o tutorial auxilia na construção de OVA utilizando-se a plataforma Roblox e apresenta uma solução válida para o problema proposto, demonstrando a efetividade do tutorial construído e a possibilidade de substituição do help do Roblox Studio por ele.

Deste modo, no contexto do presente trabalho, têm-se:

- Identificação do problema e motivação:
 - Falta de distinção entre a documentação para professores e desenvolvedores;
 - Teor técnico da documentação disponível;
 - Falta de um help que auxilie o desenvolvedor no Roblox Studio.
- Definição dos objetivos para uma solução:
 - Explorar o Roblox Studio;
 - Construir os projetos e realizar tutoriais disponíveis na documentação da plataforma;
 - Analisar e elencar passos essenciais para construção de experiências no Roblox Studio;
 - Definir um passo a passo para criação de OVA utilizando o Roblox.
- *Design* e desenvolvimento: Demonstrar e descrever os seguintes tópicos:
 - Evidenciar como deve ser realizada a instalação do Roblox Studio;
 - Como criar uma conta no Roblox;

- Como criar projetos no Roblox Studio;
- Comandos básicos do Roblox Studio;
- Onde encontrar modelos prontos para se incluir no projeto;
- Como incluir esses modelos no projeto;
- Explicitar o que são *scripts* e onde encontrá-los;
- Como realizar testes nas experiências construídas;
- Demonstração: Utilizar o tutorial criado para construir OVA;
- Avaliação:
 - Utilizar um conjunto de testes de caixa-preta para avaliar a funcionalidade da plataforma Roblox na construção de OVA;
 - Avaliar a completude do Tutorial na geração de OVA;
- Comunicação: Realizar a publicação do trabalho e disponibilizar os OVA construídos na plataforma Roblox.

Os tópicos para construção do tutorial foram elencados de acordo com uma análise dos passos essenciais para a construção de experiências, os quais foram obtidos a partir da exploração da documentação da plataforma e replicação dos projetos disponíveis para aprendizado.

Segundo Bastos et al. (2007), os testes funcionais do sistema, também chamados de testes de caixa-preta, são construídos para garantir que os requisitos e especificações do sistema sejam atendidos, possuindo diversas técnicas de realização. Nesta proposta, foi utilizada a técnica de testes de requisitos.

Bastos et al. (2007) afirmam ainda que os testes de requisitos visam verificar se o sistema executa corretamente as funcionalidades, averiguando se elas estão implementadas

em conformidade com o que foi solicitado, e se correções realizadas se mantêm em funcionamento. Esses testes podem ser executados por meio da criação de um checklist de funcionalidades; sendo assim, eles foram listados a seguir, com seus respectivos resultados:

- *Movimentação das Labels*: O movimento das *labels* pode ser executado nos dois cenários, no Roblox e no H5P.
- *Verificação da quantidade de acertos do usuário*: É possível contabilizar essa quantidade em ambos os cenários.
- *Labels não podem ser alocadas no mesmo local*: Não há possibilidade de alocação de *labels* no mesmo local, o Roblox evita o posicionamento retornando a *label* para a origem e o H5P realiza a substituição.
- *Reinicialização da experiência*: No Roblox pode ser realizado a qualquer momento, no H5P apenas ao finalizar todas as etapas, e apresenta travamentos nessa reinicialização.
- *Destaque de locais de posicionamento de Labels*: As duas ferramentas deixam o plano de fundo dos locais de cores diferentes.

Ao final da construção do objeto com a plataforma Roblox, pôde-se verificar que o OVA produzido condiz com as funcionalidades existentes no H5P e cumpre com sua proposta metodológica, sendo possível replicar os comportamentos do H5P, adaptando-se alguns pontos quando necessário. Além disso, a completude do tutorial construído pôde ser verificada, uma vez que OVAs funcionais foram produzidos ao final de seus passos.

4 RESULTADOS E DISCUSSÃO

Este capítulo apresenta as experiências construídas no Roblox em duas abordagens, uma bidimensional e outra tridimensional. O detalhamento do desenvolvimento destas instâncias é descrito no tutorial disponível no Apêndice A e no repositório online disponível em Murilo (2024).

Foram disponibilizadas no repositório do Roblox as duas experiências (OVA) criadas, permitindo aos usuários da plataforma tanto jogá-las quanto baixar o pacote do objeto e editá-lo no Roblox Studio. O detalhamento destes resultados, bem como as instruções para utilizá-los, estão presentes no tutorial. Os scripts criados também estão disponíveis no mesmo repositório e podem ser utilizados e reutilizados/modificados, desde que preservada e mencionada a autoria original.

O treinamento *Creating an Adventure Game*¹, disponível na plataforma Roblox, foi utilizado como ponto de partida para a definição dos primeiros passos na criação de experiências com o Roblox. Através do treinamento, foi possível compreender a construção de mapas, estruturas básicas, scripts e controles do Roblox Studio, além de compreender a sintaxe e a semântica da linguagem empregada na plataforma.

Após a compreensão dos conceitos básicos envolvendo o Roblox, foram construídos dois OVAs utilizando o Roblox Studio, os quais exploram o conceito de mudanças de estado físico. Estes objetos se basearam em outros desenvolvidos anteriormente por Murilo, Santos e Silva (2022) utilizando uma outra tecnologia, a H5P².

O primeiro OVA desenvolvido, em 2D, consiste em replicar um objeto desenvolvido em um projeto de pesquisa anteriormente executado, o qual possui a funcionalidade de arrastar rótulos para as posições corretas de transformação. A Figura 4 apresenta a solução proposta para o OVA, construída com o H5P, enquanto, nas Figuras 5 e 6, são demonstradas, respectivamente, a primeira e a última versão construída no ambiente do Roblox.

1 Criação de um jogo de aventura onde os jogadores exploram um mundo em busca de itens para colher e vender. É indicado para pessoas que têm alguma familiaridade com o Roblox Studio e conceitos básicos de programação até declarações condicionais (if/then statements). Disponível em: [Creating an Adventure Game | Documentação - Central de Criadores Roblox](#).

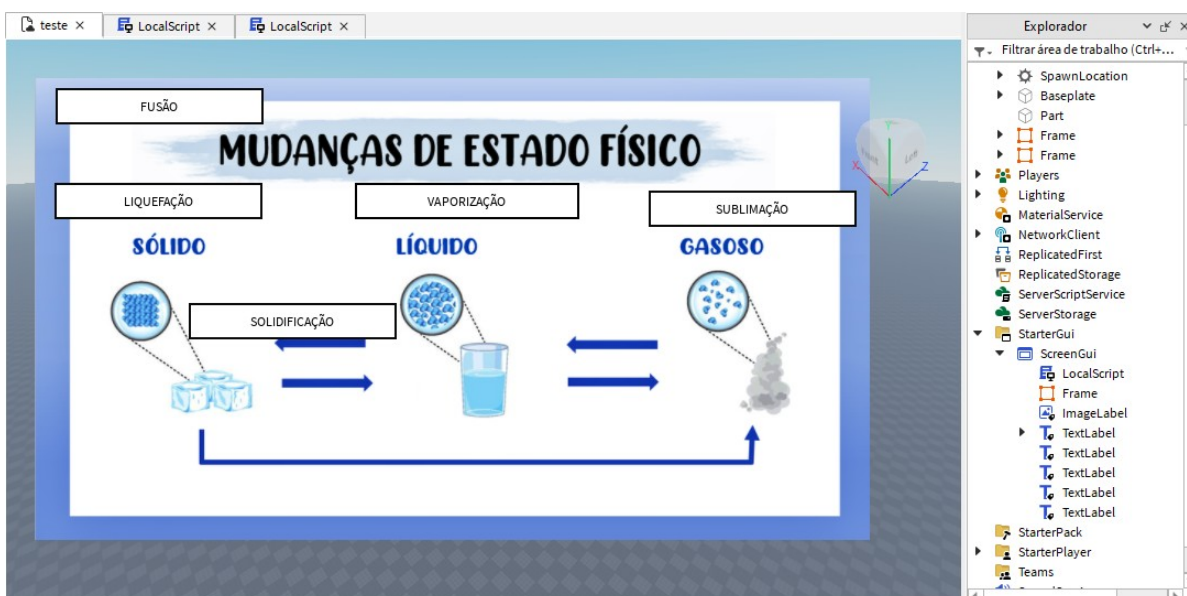
2 O H5P é um sistema de criação de conteúdo interativo baseado em HTML5 que permite criar e compartilhar facilmente conteúdo online interativo, como questionários, apresentações, simulações, jogos, etc. Ele é usado por educadores, desenvolvedores de treinamento e profissionais de e-learning para criar e distribuir facilmente conteúdo interativo na web. Disponível em: [H5P – Create and Share Rich HTML5 Content and Applications](#).

Figura 4 – OVA Mudanças de Estados Físicos utilizando H5P



Fonte: O autor, 2023.

Figura 5 – Versão inicial do OVA 2D construído no Roblox



Fonte: O autor, 2023.

Figura 6 – Versão Final do OVA 2D construído no Roblox



Fonte: O autor (2023).

No desenvolvimento do OVA 2D, surgiu um desafio que dificultava sua finalização. Os objetos do Roblox Studio usados na criação de interfaces 2D, conhecidos como GUI Objects, não possuem métodos nativos para a colisão de elementos. Portanto, para determinar se os rótulos foram arrastados para a posição adequada, foi necessário implementar um método de colisão.

O método implementado avalia as dimensões do rótulo e realiza um somatório com sua posição na tela, o qual possibilita verificar se o rótulo entrou na área adequada de soltura, viabilizando, assim, a verificação da corretude da resposta.

Além da verificação da posição dos rótulos na versão final do OVA, demonstrada na Figura 6, foi implementado um placar que adiciona um ponto para cada rótulo posicionado corretamente.

Com o intuito de aprimorar a experiência do usuário, foram introduzidos dois botões distintos. O primeiro tem a finalidade de acionar o método de avaliação das dimensões, conforme previamente descrito. Caso os rótulos estejam posicionados corretamente, é modificada a cor do rótulo, proporcionando um reforço visual positivo ou negativo. A cor verde foi selecionada para indicar a disposição correta, e a vermelha, para sinalizar uma posição incorreta.

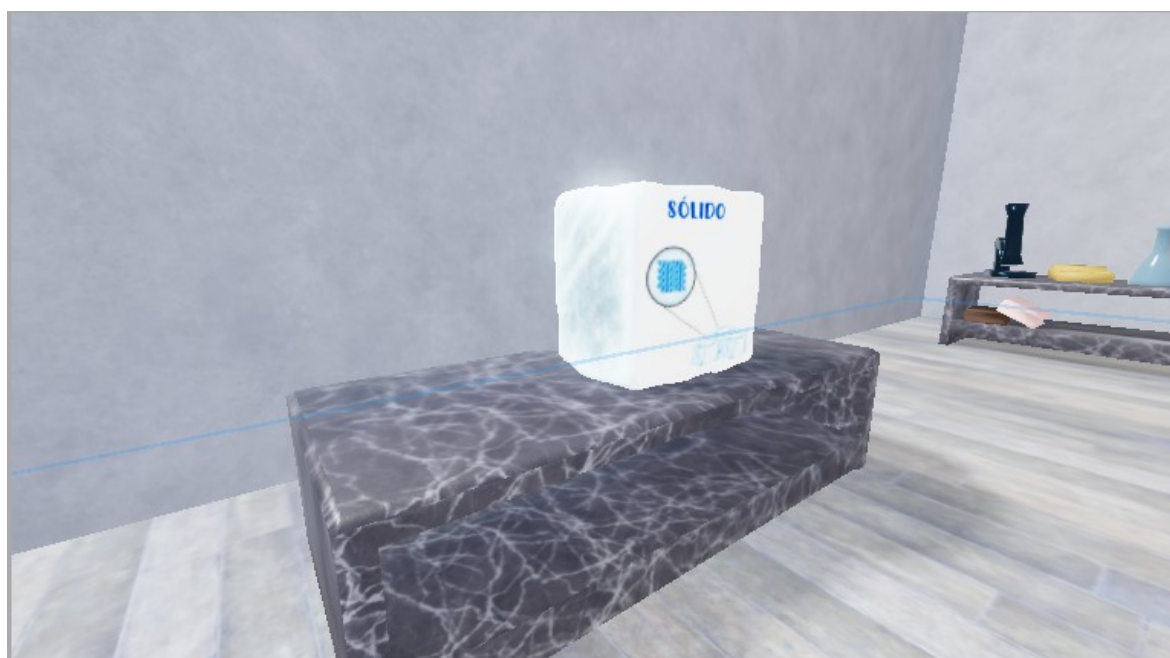
O segundo botão tem a função de reiniciar a experiência, revertendo-a ao ponto inicial,

com os rótulos reposicionados no topo da interface. Nesse estágio, os rótulos assumem cores acinzentadas, e o placar, o valor de zero pontos.

O segundo OVA desenvolvido, este em 3D, representa uma adaptação da versão em duas dimensões, entretanto com interações distintas. As interações desse modelo estão centradas na coleta de objetos encontrados em blocos que simbolizam os estados da matéria: sólido, líquido e gasoso.

A coleta dos objetos é ativada ao pressionar a tecla **e**, permitindo a incorporação de um desses estados ao inventário do jogador, resultando na sua visualização nas mãos do personagem, conforme ilustrado nas Figuras 7 e 8.

Figura 7 – Bloco de interação do OVA 3D



Fonte: O autor (2023).

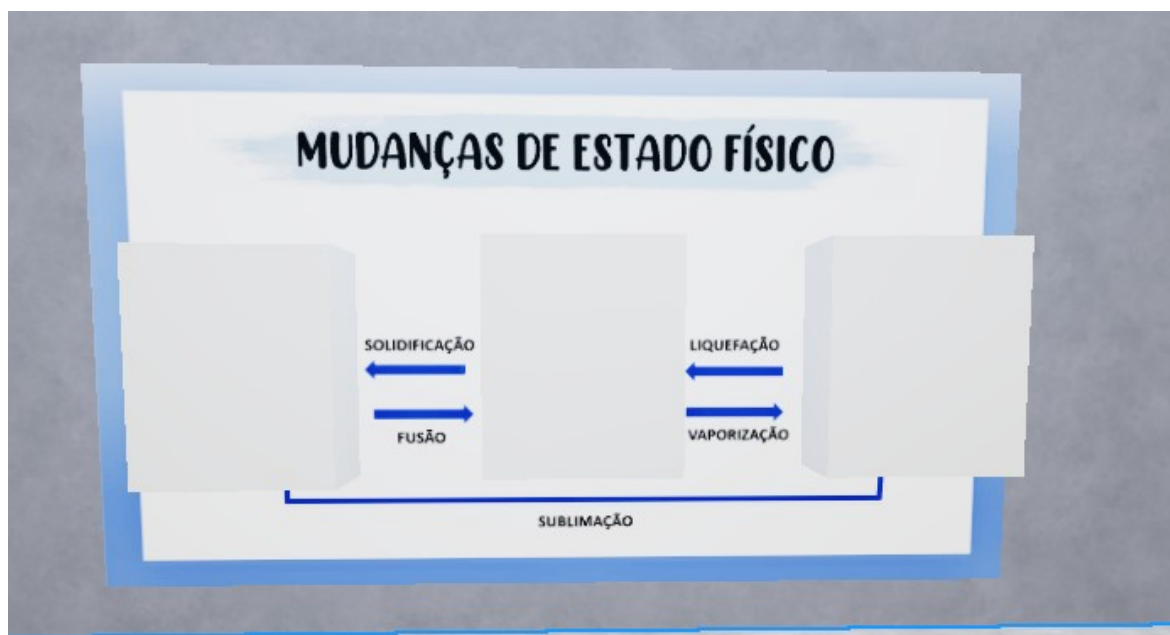
Figura 8 – Bloco de interação do OVA 3D



Fonte: O autor (2023).

Ao adquirir um destes blocos, o jogador precisa transportá-lo até o quadro de transformações, o qual partilha o mesmo plano de fundo do OVA 2D. No entanto, nesse contexto, a estratégia utilizada envolve o preenchimento de espaços vazios que representam os estados da matéria, em vez de as transformações, como exemplificado na Figura 9.

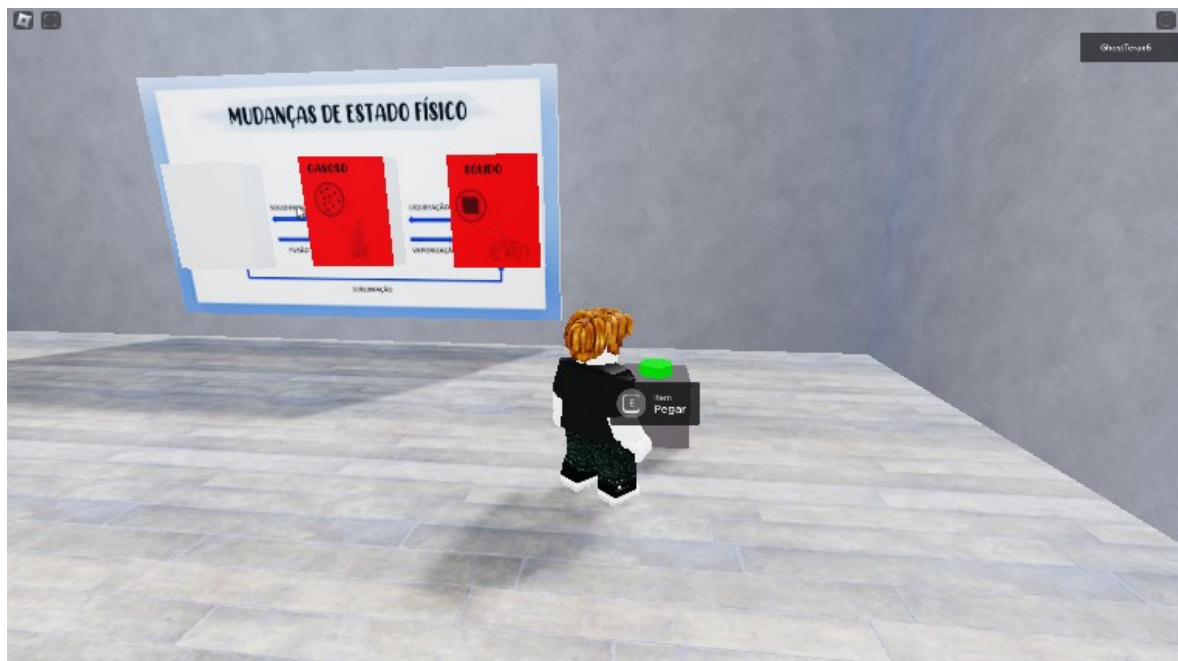
Figura 9 – Quadro de Transformações do OVA 3D



Fonte: O autor (2023).

Após realizar a interação com o quadro de transformações, o jogador tem a possibilidade de interagir com um botão localizado ao lado do quadro e verificar suas respostas.

Figura 10 – Verificação de respostas do OVA 3D



Fonte: O autor (2023).

Após a verificação das respostas, a experiência aguarda três segundos e retorna ao seu estado original, em que o jogador pode capturar os blocos novamente e colocá-los no quadro de transformação.

Nessa etapa de construção do objeto 3D, as principais dificuldades foram em como representar e equipar os itens no instante em que o jogador interage com os blocos de estado da matéria, uma vez que é necessário realizar uma construção que envolve uma malha gráfica e funcionalidades que identificam a proximidade do jogador com o objeto em questão.

Apesar da proposta inicial, de desenvolver um tutorial destinado a professores leigos, com base nas dificuldades encontradas, acredita-se que essa abordagem não é viável, pois a criação de experiências no Roblox demanda conhecimentos de programação, os quais, aplicados na elaboração de *scripts*, desempenham a função de controlar integralmente as reações decorrentes de uma ação na experiência.

Apesar de se acreditar que o tutorial não pode ser utilizado por leigos em programação, a sua criação também foi efetuada nesta proposta, porém, com foco na descrição de como replicar os dois tipos de objetos apresentados por ela, bem como direcionar

adequadamente aqueles que possuem os conhecimentos mínimos para tal. Este tutorial é uma ferramenta digital em formato PDF, seguindo a estrutura delineada na Seção 2.3. Os pontos previamente estabelecidos foram organizados em formato de orientação passo a passo, em que os leitores têm acesso a descrições textuais e imagens que guiarão suas ações, como evidenciado na Figura 11.

Figura 11 – Tela do Roblox Studio

1 Como instalar o Roblox Studio

O Roblox Studio é uma ferramenta para construção de experiências personalizadas da plataforma Roblox, o qual está disponível no próprio site da empresa, acessível através do link: <https://create.roblox.com>. Para realizar a instalação dessa ferramenta, você deve acessar o link apresentado, e clicar no botão destacado pelo retângulo vermelho, **COMECE A CRIAR**, assim como demonstra a Figura 1.

Figura 1 - Botão de instalação da Ferramenta Roblox Studio



Fonte: Os autores (2023).

Após realizar o clique no botão **COMECE A CRIAR**, será exibida uma mensagem informando que o site está tentando abrir o Roblox, nesse momento, é necessário clicar no botão abrir, que será exibido em formato de janela de mensagem, conforme a Figura 2. Em seguida irá se iniciar a instalação do Roblox Studio, a qual se dará de forma automática, não sendo necessário realizar nenhuma configuração. As Figuras 2 e 3, demonstram esta etapa.

Fonte: O autor (2023).

A fim de facilitar a construção das experiências, o tutorial aborda repositórios de modelos 3D gratuitos que disponibilizam estruturas, vegetação e vários outros itens para compor o cenário da experiência a ser criada. Esses repositórios agilizam a construção das experiências, pois não é necessário investir tempo na criação dos elementos do cenário.

Também se destaca que existem alguns scripts prontos, ou que servem de modelo, para que os usuários possam utilizá-los para construir os seus próprios scripts. O acesso a

esses scripts também pode ser encontrado no tutorial.

A presente pesquisa evidenciou a vasta gama de aplicações do Roblox Studio no contexto educacional, corroborando as observações de Pizzol et al. (2022). A ferramenta não apenas se revelou eficaz na criação de ambientes virtuais de aprendizagem, destacando-se por sua potencialidade como instrumento criativo e inovador, mas também demonstrou uma adaptabilidade notável, conforme discutido por Siqueira (2019). A capacidade de construir experiências educacionais personalizadas alinha-se com a flexibilidade destacada por diversos pesquisadores.

As contribuições de Da Cruz, Hofstätter e Pires (2022) no desenvolvimento de material audiovisual para o ensino de Artes acrescentam uma perspectiva valiosa ao panorama, evidenciando que a versatilidade do Roblox Studio transcende as fronteiras de disciplinas específicas. Nesse contexto, os Objetos Virtuais de Aprendizagem (OVAs) delineados neste trabalho emergem como uma aplicação prática e significativa da ferramenta no ensino de Química.

Ao finalizar, a pesquisa ressalta não apenas a eficiência do Roblox Studio como uma ferramenta educacional multifacetada, mas também destaca a importância de explorar continuamente suas potencialidades em diferentes domínios de conhecimento. Este estudo contribui para uma compreensão mais aprofundada das aplicações práticas do Roblox Studio no contexto educacional, abrindo portas para futuras investigações e inovações no campo do design instrucional e da tecnologia educacional.

5 CONCLUSÃO

O objetivo principal do presente trabalho foi compreender como são construídas experiências para o ambiente do Roblox, sob a perspectiva de verificar a viabilidade da construção de objetos virtuais de aprendizagem por professores leigos em programação.

As dificuldades e problemas encontrados ao longo do trabalho mostraram que, apesar de ser possível a utilização do Roblox para o desenvolvimento de OVA, acredita-se ser necessário um conhecimento prévio de programação por parte dos professores. Esta constatação pode revelar uma refutação da hipótese de que professores leigos conseguiriam construir OVA no Roblox.

Entretanto, ressalta-se que é viável a utilização deste ambiente para o desenvolvimento de OVAs, respeitando-se suas características e capacidade conceituais, como a possibilidade de replicação e modificação de objetos.

O primeiro objetivo específico alcançado nesta pesquisa está relacionado à caracterização do estado da arte na utilização do Roblox na educação, conforme evidenciado na Seção 2.5. Nela, destacam-se trabalhos que empregaram, de diversas maneiras, o Roblox Studio como ferramenta de apoio à educação, resultando na construção de experiências classificadas como Objetos Virtuais de Aprendizagem (OVAs).

Atendendo ao segundo objetivo específico, foi elaborado um tutorial que delinea os principais pontos para a criação de experiências tanto bidimensionais quanto tridimensionais. Este tutorial, embora tenha definido a construção de experiências relacionadas ao conteúdo de química, oferece a flexibilidade para aplicar os conceitos abordados a uma variedade de conteúdos, exigindo conhecimentos básicos de lógica de programação.

Apesar de o segundo objetivo específico propor a construção de um tutorial para professores leigos em programação, devido às dificuldades encontradas, há indícios que levam a crer que é necessário um conhecimento básico em lógica de programação, uma vez que é indispensável o uso de scripts para definir o comportamento e as ações executadas na experiência.

Por fim, o terceiro e último objetivo específico também foi cumprido. Este consistiu na realização de uma prova de conceito na qual se propunha a construção de um OVA com base no tutorial. Destaca-se que não foi desenvolvido apenas um, mas dois OVAs, empregando abordagens e dimensionalidades distintas.

Ambos os OVAs estão relacionados à transformação dos estados da matéria, evidenciando a aplicabilidade prática dos conceitos abordados. Esses resultados consolidam a

contribuição deste trabalho para a compreensão e aplicação do Roblox Studio como uma ferramenta educacional versátil e eficaz.

Destaca-se a relevância dos resultados obtidos ao se preencher uma lacuna na literatura existente sobre a utilização do Roblox como ferramenta na construção de OVAs. As descobertas não apenas fornecem uma visão clara da versatilidade do Roblox Studio, mas também oferecem insights valiosos para estudantes, educadores e pais.

Compreender a versatilidade de uma ferramenta como o Roblox Studio na construção de ferramentas de apoio à educação, em um cenário onde a conectividade digital desempenha um papel cada vez mais central na vida dos adolescentes, é de grande valia no contexto do ensino contemporâneo.

Ao encerrar esta pesquisa, é imperativo reconhecer que é necessário conhecimento prévio para construir experiências de qualidade. No entanto, a relevância contínua deste estudo dependerá da capacidade de adaptar suas conclusões às transformações rápidas e contínuas no mundo digital. A aplicação destas descobertas a diversas áreas da educação oferece oportunidades para aprimorar métodos de ensino e promover a inovação educacional.

Sugere-se que futuras pesquisas explorem a integração do Roblox Studio em contextos específicos de aprendizagem, bem como a criação de recursos adicionais para apoiar professores e estudantes no desenvolvimento de habilidades em programação. Investigar estratégias pedagógicas eficazes para a implementação do Roblox na sala de aula poderia ampliar ainda mais o impacto dessa ferramenta no processo educacional.

Sugere-se, também, que seja realizado um estudo envolvendo professores leigos, a fim de validar a efetividade do tutorial e a possibilidade de refutação da hipótese de que esses profissionais não poderiam utilizar o modelo construído.

Em suma, este estudo destaca não apenas as potencialidades do Roblox na construção de OVAs, mas também ressalta a importância de considerar as habilidades necessárias para maximizar seu potencial como ferramenta de apoio à educação.

REFERENCIAS

ABRAHAMSEN, Jeanette. **Tools to Create Interactive Digital Communities as our World Embraces Virtual Learning**. 2022. Disponível em:

https://members.aect.org/pdf/Proceedings/proceedings22/2022/22_01.pdf. Acesso em: 23 mai. 2023.

BASTOS, Anderson *et al.* **Base de conhecimento em teste de software**. 2. ed. rev. São Paulo: Martins, 2007. ISBN 978-85-99102-89-3.

BRUGNAGO, Ana Paula *et al.* **Jogos Digitais No Ensino De Física: Um Olhar Sob O Prisma De Pibidianos Relato De Experiência**. In: VIII ENALIC, 2018, Fortaleza. **Anais eletrônicos** [...] Fortaleza: UTFPR, 2018. Disponível em:

https://editorarealize.com.br/editora/anais/enalic/2021/TRABALHO_EV163_MD3_SA101_I_D847_16112021144151.pdf. Acesso em: 19 abr. 2023.

CARBONE, Filipe. Roblox atinge 1,7 bilhão de usuários em setembro. **Adrenaline**. 19 de out. 2022. Disponível em: <https://www.adrenaline.com.br/games/roblox-atinge-17-bilhao-de-usuarios-em-setembro/>. Acesso em: 19 abr. 2023.

DA CRUZ, Raíssa Eduarda Alves; HOFSTÄTTER, Yuri; PIRES, Débora. **MÚSICA E TECNOLOGIA: UMA PROPOSTA DE DESENVOLVIMENTO DE MATÉRIAS AUDIOVISUAIS**. In: XV MICTI, 2022, Rio do Sul. Disponível em:

<https://publicacoes.ifc.edu.br/index.php/micti/article/download/3731/3070>. Acesso em: 19 abr. 2023.

DA SILVA, Ricardo. **Gamificação e Inteligência Coletiva para Promover a Participação em Sistema de Bate-Papo para Educação**. Apresentada como dissertação de mestrado, UNIRIO, 2016. Disponível em:

<http://www.repositorio-bc.unirio.br:8080/xmlui/bitstream/handle/unirio/11515/MI%2019-2016.pdf?sequence=1>. Acesso em: 24 mai. 2023.

FUNDAÇÃO LEMANN. 2002-2023 .Uma organização de filantropia familiar. Disponível em: <https://fundacaolemann.org.br/noticias/aprendizagem-centrada-no-estudante>. Acesso em: 21 mai. 2023.

GALAFASSI, Fabiane Penteadó; GLUZ, João Carlos; GALAFASSI, Cristiano. Análise Crítica das Pesquisas Recentes sobre as Tecnologias de Objetos de Aprendizagem e

Ambientes Virtuais de Aprendizagem. **Revista Brasileira De Informática Na Educação**, Porto Alegre, v. 21, n. 3, p. 41-52, dez. 2013. Disponível em:

<http://ojs.sector3.com.br/index.php/rbie/article/view/2351/2457>. Acesso em: 19 abr. 2023.

HERNÁNDEZ, Lorena *et al.* The use of massive online games in game-based learning activities. **Revista Innova Educación**, v. 4. n. 03, p. 7-30, mar. 2022. Disponível em:

<https://www.revistainnovaeducacion.com/index.php/rie/article/view/573/585>. Acesso em: 21 mai. 2023.

HEVNER, Alan R. Design Science In Information Systems Research. **Revista Mis Quarterly**, v. 28, n. 1, p. 75-105, mar. 2004. Disponível em:

https://www.researchgate.net/publication/201168946_Design_Science_in_Information_Systems_Research. Acesso em: 06 mai. 2023.

H5P. Site oficial do H5P. Disponível em: <https://h5p.org>. Acesso em: 25 dez. 2023.

LONG, Roma. **ROBLOX AND EFFECT ON EDUCATION**. 2020. Foundation University-Philippines. Disponível em:

https://www.researchgate.net/profile/Roma-Long/publication/340102957_ROBLOX_AND_EFFECT_ON_EDUCATION/links/5e78eaa392851c309137c7ea/ROBLOX-AND-EFFECT-ON-EDUCATION.pdf. Acesso em: 19 abr. 2023.

MISTRETTA, Sharon. The Metaverse – An Alternative Education Space. **IntechOpen Journals**, p. 1-23, mar. 2022. Disponível em:

<https://cdn.intechopen.com/journals/static/87/acrt2022005.pdf>. Acesso em: 19 abr. 2023.

MURILO, Jorge Luís Vieira; SANTOS, Alda; SILVA, Gabriel da. (2022). **Inovando o processo ensino-aprendizagem pela adoção de objetos virtuais de aprendizagem para o ensino de Química**. Programa Institucional de Bolsas de Iniciação em Desenvolvimento Tecnológico e Inovação, Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais – Campus Bambuí, Bambuí, Brasil. (Não Publicado).

MURILO, Jorge Luís Vieira. TUTORIAL PARA CONSTRUÇÃO DE OVA UTILIZANDO A FERRAMENTA ROBLOX STUDIO. Disponível em: no endereço:

<https://github.com/Jorge-Comp/ESTUDO-DA-VIABILIDADE-DE-USO-DA-PLATAFORMA-ROBLOX-PARA-A-CRIA-O-DE-OBJETOS-VIRTUAIS-DE-APRENDIZAGEM/blob/e1d6367ed60a594302e01c56c4d3577eac0f99cc/Tutorial.pdf>.

Acesso em 19 mar. 2024.

PEFFERS, Ken *et al.* **A Design Science Research Methodology for Information Systems Research**. Disponível em: <https://www.tandfonline.com/doi/epdf/10.2753/MIS0742-1222240302?needAccess=true&role=button>. Acesso em: 24 mai. 2023.

PEREIRA, Soraya Rachel. METAVERSONA EDUCAÇÃO: CURRÍCULO, DESAFIOS E POSSIBILIDADES. **Revista Painel Metaverso**. Florianópolis, v. 1, n. 4, 2022. Disponível em: <https://painelmetaverso.com.br/ojs/index.php/home/article/view/6/5>. Acesso em: 21 mai. 2023.

PIMENTEL, Mariano; FILIPPO, Denise; SANTORO, Flávia Maria. **Design Science research: fazendo pesquisas científicas rigorosas atreladas ao desenvolvimento de artefatos computacionais projetados para a educação**. Disponível em: https://metodologia.ceie-br.org/wp-content/uploads/2018/10/cap1_5.pdf. Acesso em: 24 mai. 2023.

PIRES, Débora Costa *et al.* **MUSTEC: UMA PROPOSTA DE DESENVOLVIMENTO DE MATERIAIS AUDIOVISUAIS**. In: FEPEX: 7ª Feira de ensino, Pesquisa e Extensão do IFC campus Fraiburgo, 2022, Fraiburgo.. Disponível em: <https://publicacoes.ifc.edu.br/index.php/fepexfraiburgo/article/view/3264>. Acesso em: 21 mai. 2023.

PIZZOL, Andrieli DAL; BUSSOLOTTO, Luis Eduardo; LIRA, Aliandra Cristina Mesomo. O Processo Educativo Para Além Do Jogo: Roblox E A Revolução na Experiência Virtual Dos Nativos Digitais. **Revista Aproximação**, Guarapuava, v. 4, n. 09, p. 76-85, dez. 2022. Disponível em: <https://revistas.unicentro.br/index.php/aproximacao/article/view/7377/5308>. Acesso em: 19 abr. 2023.

ROBLOX. Desenvolvido por Roblox Corporation. Site oficial da plataforma Roblox. Disponível em: <https://create.roblox.com/docs/getting-started>. Acesso em: 19 abr. 2023.

SALVADOR, Pétala Tuani Candido de Oliveira *et al.* Objeto e ambiente virtual de aprendizagem: análise de conceito. **Revista Brasileira de Enfermagem**, v. 70, p. 572-579, jun. 2017. Disponível em: <https://www.scielo.br/j/reben/a/DnCS4GNJYW3vVq93bDxQDZx/abstract/?lang=pt#>. Acesso em: 29 jun. 2023.

SIQUEIRA, Beatriz. O ensino de Geografia Física e os jogos digitais: trabalhando susceptibilidade, vulnerabilidade e resiliência frente aos desastres naturais. **Revista Terrae Didat**, Campinas, v. 15, n. 19022, p. 1-12, abri. 2019. Disponível em:

<https://periodicos.sbu.unicamp.br/ojs/index.php/td/article/view/8653224/21320>. Acesso em: 23 mai. 2023.

WAZLAVICK, Raul Sidnei. **Metodologia de Pesquisa para Ciência da Computação**. Rio de Janeiro: Elsevier, 2009. Disponível em:

<https://tsxvpsbr.dyndns.org/arquivos/UFFS/Metodologia%20De%20Pesquisa%20CienciaDaComputacao%20%20TCC1.pdf>. Acesso em: 06. mai 2023.

APÊNDICE A

INSTITUTO FEDERAL DE EDUCAÇÃO CIÊNCIA E TECNOLOGIA
DE MINAS GERAIS – *CAMPUS* BAMBUÍ-MG
BACHARELADO EM ENGENHARIA DE COMPUTAÇÃO

JORGE LUÍS VIEIRA MURILO

**TUTORIAL PARA CONSTRUÇÃO DE OVA UTILIZANDO A FERRAMENTA
ROBLOX STUDIO**

**BAMBUÍ-MG
2023**

JORGE LUÍS VIEIRA MURILO

**TUTORIAL PARA CONSTRUÇÃO DE OVA UTILIZANDO A FERRAMENTA
ROBLOX STUDIO**

Tutorial elaborado para auxiliar professores que na construção de OVA personalizados utilizando o Roblox Studio.

**BAMBUÍ-MG
2023**

LISTA DE FIGURAS

Figura 1 – Botão de instalação da Ferramenta Roblox Studio.....	7
Figura 2 – Iniciar Instalação do Roblox Studio.....	8
Figura 3 – Etapa de Instalação.....	8
Figura 4 – Criar uma conta no Roblox.....	11
Figura 5 - Prenchimento de informações pessoais.....	12
Figura 6 – Confirmar o Login.....	13
Figura 7 – Tela do Roblox Studio.....	13
Figura 8 – Templates Básicos.....	14
Figura 9 – Criação de Experiência.....	15
Figura 10 – Login no Roblox Studio.....	16
Figura 11 – Salvar na Roblox.....	17
Figura 12-Criar novo Jogo.....	17
Figura 13-Informações Básicas.....	18
Figura 14 – Tela do Roblox Studio.....	18
Figura 15 – Tela do <i>website</i> Free3D.....	20
Figura 16 – Tela do <i>website</i> TurboSquid.....	21
Figura 17 – Importar modelo 3D no Roblox.....	22
Figura 18 – Importar modelo 3D no Roblox.....	23
Figura 19 – Importar Modelo.....	23
Figura 20 – Modelo Importado.....	24
Figura 21 – Exibição de janelas.....	26
Figura 22 – Seleção de um objeto “part”.....	27
Figura 23 – Adição de Script.....	27
Figura 24 - Testar a Experiência.....	28
Figura 25 – Exibição de janelas.....	29
Figura 26 – Repositório de Scripts, Haddock.....	30
Figura 27 – Repositório de Scripts, Haddock.....	30
Figura 28 – Local de Aparecimento do Jogador.....	32
Figura 29 – Inserção de objetos via Caixa de Ferramentas.....	32
Figura 30 – Inserção de objetos via Caixa de Ferramentas.....	33
Figura 31 – Cenário do OVA.....	34
Figura 32 – Inserção de cubo de gelo.....	35
Figura 33 – Inserção de retângulo.....	35
Figura 34 – Inserção de retângulo.....	36
Figura 35 – Mudança de Hierarquia de objetos.....	37
Figura 36 – Cubos.....	38
Figura 37 – Imagens na Caixa de Ferramentas.....	38
Figura 38 – Cubos com Texturas.....	39
Figura 39 – Inserir Objetos Script e ProximityPrompty.....	40
Figura 40 – Localizar ReplicatedStorage.....	42
Figura 41 – Quadro de Transformações.....	43
Figura 42 – Incluir ScreenGui.....	47
Figura 43 – Estrutura da Interface.....	48
Figura 44 – Estrutura da Interface.....	49
Figura 45 – Plano de Fundo ImageLabel.....	50
Figura 46 – Layout da experiência 2D.....	51
Figura 47 – Acessar a Experiência.....	58
Figura 48 – Abrir Roblox Player.....	59
Figura 49 – Interagir com o OVA 3D.....	59

Figura 50 – Editar OVA 3D.....	60
Figura 51 – Editar OVA 3D no Roblox Studio.....	61

LISTA DE SCRIPTS

Script 1 – Script para Coletar o Item Líquido.....	42
Script 2 – Script para posicionar o Item Líquido.....	44
Script 3 – Script Reiniciar a experiência.....	45
Script 4 – Script para movimentação das Label's.....	52
Script 5 – Script de verificação de Label Correta.....	54
Script 6 – Script para o botão de verificação.....	56
Script 7 – Script para o botão de recarregamento.....	57

SUMÁRIO

1	COMO INSTALAR O ROBLOX STUDIO.....	7
1.1	Requisitos Mínimos para instalação do Roblox Studio.....	9
1.1.1	Requisitos do sistema operacional.....	9
1.1.2	Requisitos de hardware do sistema.....	9
2	RECURSOS DO ROBLOX STUDIO.....	11
2.1	Criar uma conta.....	11
2.2	Criar Projetos.....	14
2.2.1	Salvar na Roblox.....	14
2.3	Comandos Básicos.....	19
3	REPOSITÓRIOS DE MODELOS PRONTOS.....	20
3.1	Free3d.....	20
3.2	TurboSquid.....	21
3.3	Como Incluir Modelos no Projeto.....	21
4	SCRIPTS.....	25
4.1	Como e Onde Incluir Scripts.....	26
4.2	Repositórios de Scripts.....	29
5	COMO CRIAR UMA EXPERIÊNCIA 3D NO ROBLOX.....	31
6	CONSTRUÇÃO DE UMA EXPERIÊNCIA 2D.....	47
7	EXPERIÊNCIAS DISPONIBILIZADAS.....	58

1 Como instalar o Roblox Studio

O Roblox Studio é uma ferramenta para construção de experiências personalizadas da plataforma Roblox, disponível no próprio site da empresa, acessível através do link: <https://create.roblox.com>. Para realizar a instalação dessa ferramenta, você deve acessar o link apresentado e clicar no botão COMECE A CRIAR, destacado pelo retângulo vermelho na Figura 1.

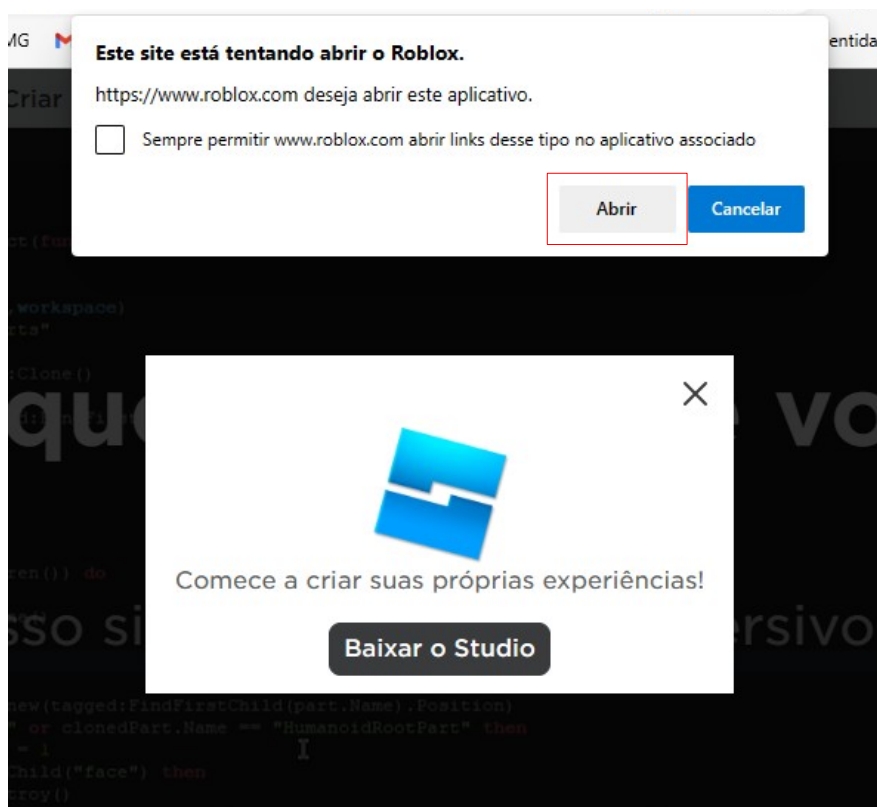
Figura 1 – Botão de instalação da Ferramenta Roblox Studio



Fonte: O autor, 2023.

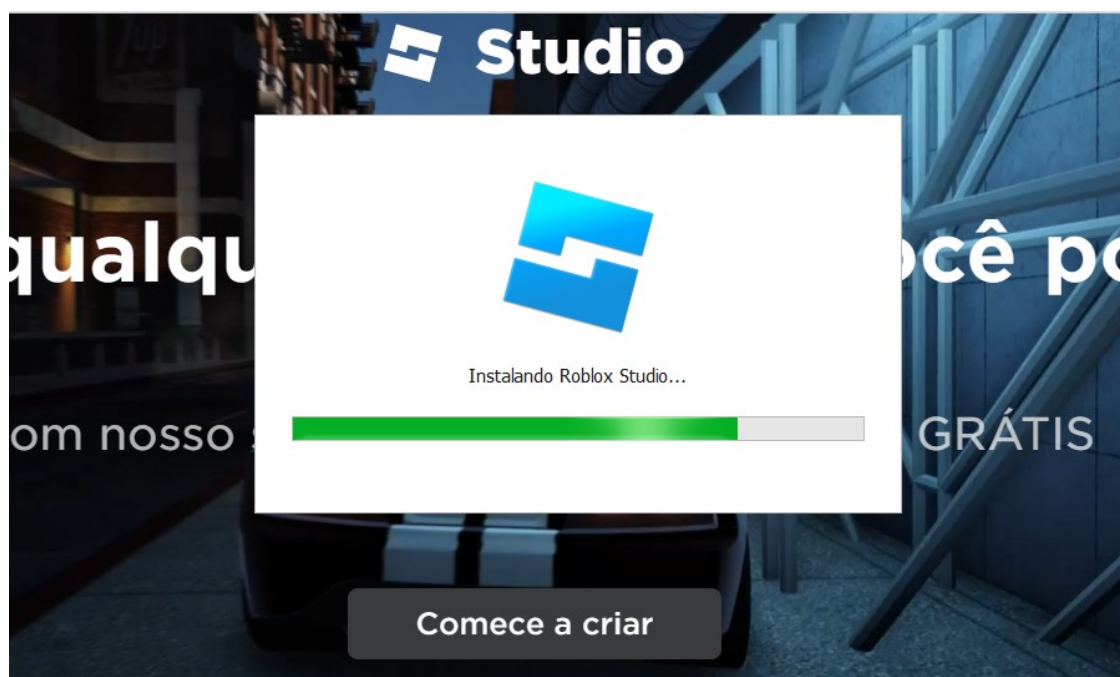
Após clicar no botão **COMECE A CRIAR**, será exibida uma mensagem informando que o site está tentando abrir o Roblox. Nesse momento, é necessário clicar no botão “abrir” que será exibido em formato de janela de mensagem, conforme a Figura 2. Em seguida, será iniciada a instalação do Roblox Studio, a qual se dará de forma automática, não sendo necessário realizar nenhuma configuração. As Figuras 2 e 3, demonstram esta etapa.

Figura 2 – Iniciar Instalação do Roblox Studio



Fonte: O autor, 2023.

Figura 3 – Etapa de Instalação



Fonte: O autor, 2023.

Após a realização dessas etapas, a ferramenta Roblox Studio estará instalada em seu computador. Para que você utilize a ferramenta de forma adequada, recomenda-se que sua máquina possua os seguintes requisitos:

1.1 Requisitos Mínimos para instalação do Roblox Studio

Os requisitos mínimos do sistema para a Roblox estão listados nas seções a seguir.

1.1.1 Requisitos do sistema operacional

Para que seja possível utilizar as Ferramentas do Roblox, é necessário que o seu computador possua os seguintes requisitos do sistema operacional.

- **Windows:** Windows 7, Windows 8.1 ou Windows 10. Para o Windows 8.1, é necessário executar o Roblox no modo Desktop, pois o modo Metro (a tela inicial lado a lado) não é suportado atualmente.
- **Navegadores suportados pelo site do Roblox para PC:** Chrome, Firefox, e Microsoft Edge.
- **Mac:** O Roblox Client será instalado na versão 10.10 (Yosemite) e superior, enquanto o Roblox Studio será instalado no Mac OS 10.11 (El Capitan) e superior.
- **O site do Roblox suporta os navegadores para Mac:** Chrome, Firefox, e Safari.
- **Linux:** O Roblox não é suportada no Linux.

1.1.2 Requisitos de hardware do sistema

Para se obter uma boa performance, tanto no teste quanto na construção de experiências com o Roblox, é necessário que seu computador possua os seguintes requisitos mínimos de seus componentes:

- **Placa de vídeo:** No PC/Windows, o aplicativo Roblox requer suporte em nível de recurso DirectX 10 ou superior. Para obter o melhor desempenho, recomenda-se um computador com menos de 5 anos com placa de vídeo dedicada ou um laptop com menos de 3 anos com placa de vídeo integrada.
- **Processador:** Recomenda-se um processador recente (a partir do ano 2005) com uma velocidade de clock de 1,6 Ghz ou superior. Visto que houve alguns problemas com processadores AMD mais antigos
- **RAM ou Memória:** A Roblox recomenda que você tenha pelo menos 1 GB de memória no Windows 7, Windows 8.1 ou Windows 10.

- **Espaço de Armazenamento:** É recomendado que haja pelo menos 20 Mb de espaço de armazenamento do sistema para instalar o Roblox.

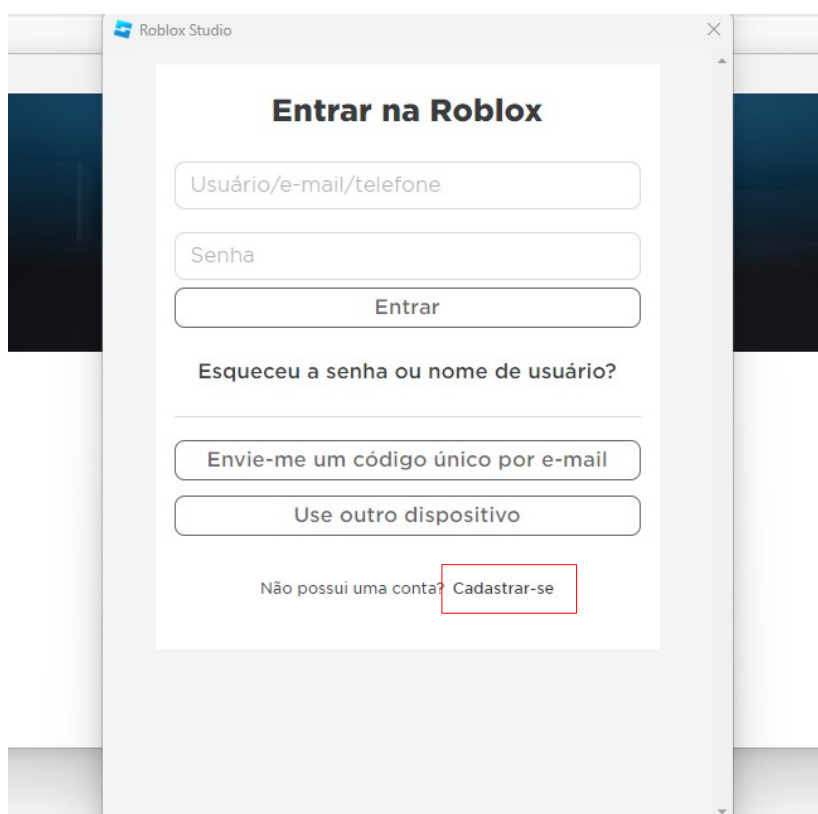
2 RECURSOS DO ROBLOX STUDIO

Para acessar os recursos do Roblox Studio, é necessário que seja criada uma conta no Roblox, caso ainda não a tenha criado. Do contrário, basta realizar login em sua conta e utilizar os recursos da ferramenta.

2.1 Criar uma conta

Caso ainda não tenha uma conta, basta criá-la clicando no link **Cadastre-se** do Roblox Studio, o qual estará disponível na tela durante o seu primeiro acesso à ferramenta, conforme exemplificado na Figura 4. Após clicar no link, você será redirecionado para uma tela semelhante à demonstrada na Figura 5.

Figura 4 – Criar uma conta no Roblox



Roblox Studio

Entrar na Roblox

Usuário/e-mail/telefone

Senha

Entrar

Esqueceu a senha ou nome de usuário?

Envie-me um código único por e-mail

Use outro dispositivo

Não possui uma conta? **Cadastrar-se**

Fonte: O autor, 2023.

Figura 5 - Preenchimento de informações pessoais

ROBLOX

CADASTRE-SE E DIVIRTA-SE!

Data de nascimento

Dia ▼ Mês ▼ Ano ▼

Usuário

Não use o seu nome real

Senha

Mínimo de 8 caracteres

Gênero (opcional)

Ao clicar em Cadastrar-se, você aceita os [Termos de Uso](#) (incluindo a cláusula de arbitragem) e a [Política de Privacidade](#).

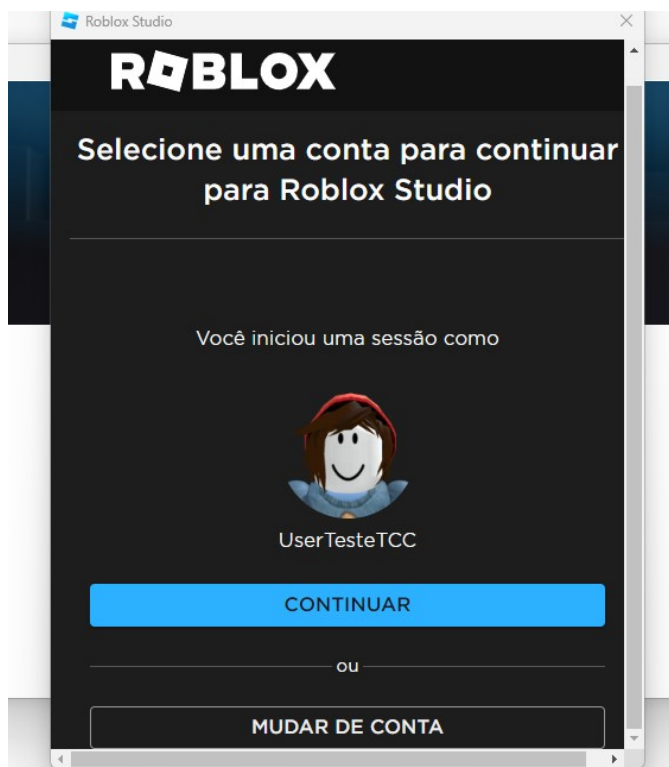
Cadastrar-se

Fonte: O autor, 2023.

Após a finalização do cadastro, você será redirecionado para a página inicial da plataforma Roblox. Nesse momento, você poderá fechar o seu navegador e realizar login no Roblox Studio, assim como evidenciado na Figura 5. Para efetuar o login, deverá preencher o campo **usuário** com o nome cadastrado na plataforma e inserir sua senha, para que seja possível acessar as funcionalidades da ferramenta. Depois disso, você deve clicar no botão **Entrar**.

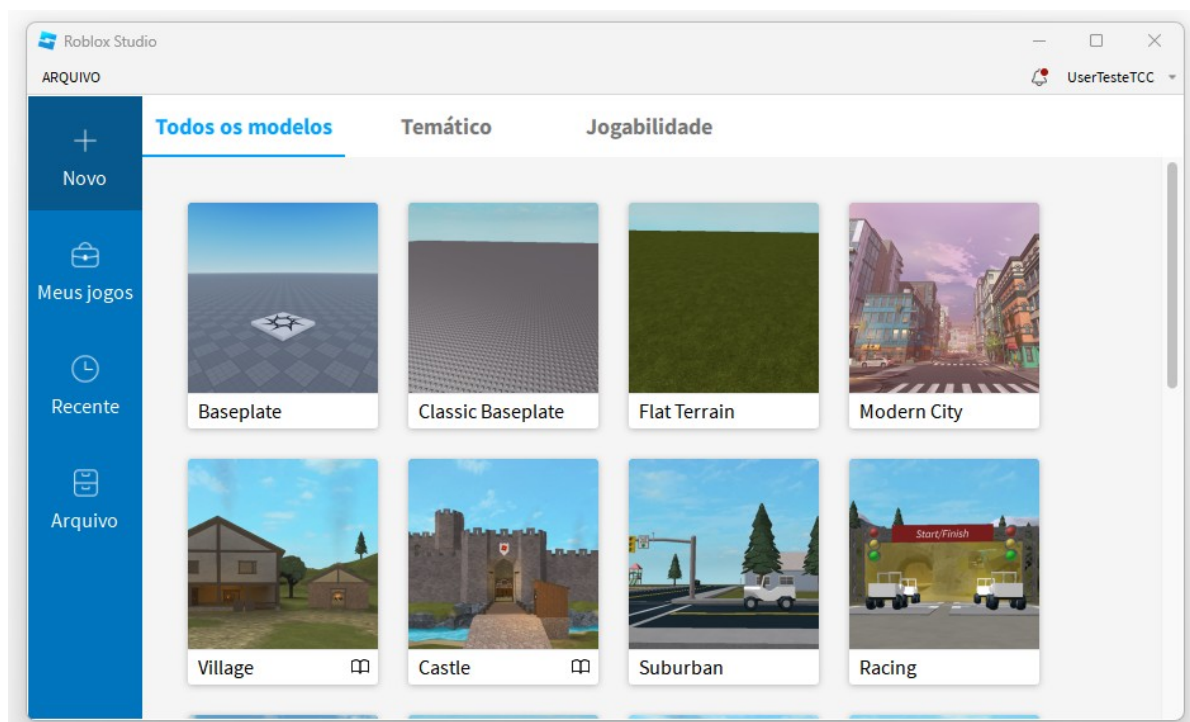
Na tela evidenciada na Figura 6, é apresentado o passo seguinte, em que você deverá clicar no botão **continuar**. Pronto! Você agora tem acesso aos recursos do Roblox Studio e verá uma tela semelhante à demonstrada na Figura 7.

Figura 6 – Confirmar o Login



Fonte: O autor, 2023.

Figura 7 – Tela do Roblox Studio

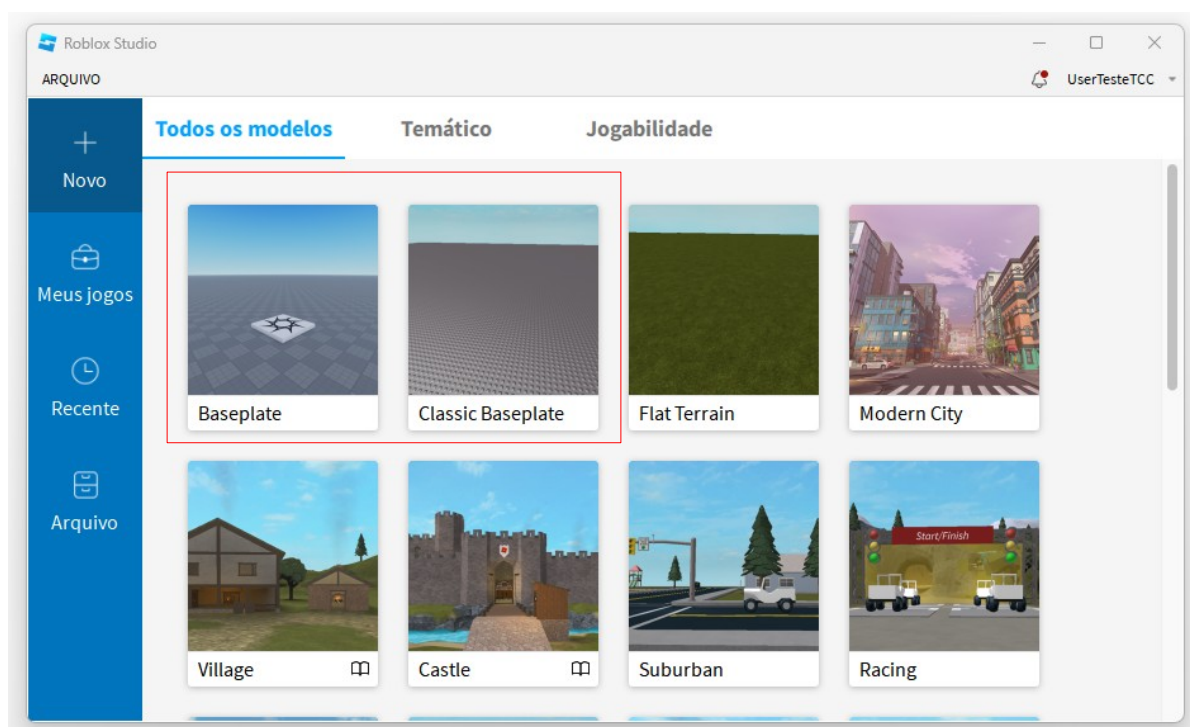


Fonte: O autor, 2023.

2.2 Criar Projetos

Para se criar um projeto, existem diversos modelos prontos, os quais incluem cenários diversificados. Dentre eles, há dois que constituem os modelos mais básicos para construção de experiências e que podem ser utilizados para construí-las **do zero**. Eles se encontram destacados pelo retângulo vermelho, na Figura 8.

Figura 8 – Templates Básicos



Fonte: O autor, 2023.

A diferença entre esses cenários é quanto à possibilidade de escolha do local onde o personagem surgirá. No primeiro modelo, chamado de **Baseplate**, é possível realocar a estrutura de surgimento de acordo com a vontade do desenvolvedor da experiência; já no segundo modelo, ele surgirá em um local padrão do cenário

2.2.1 Salvar na Roblox

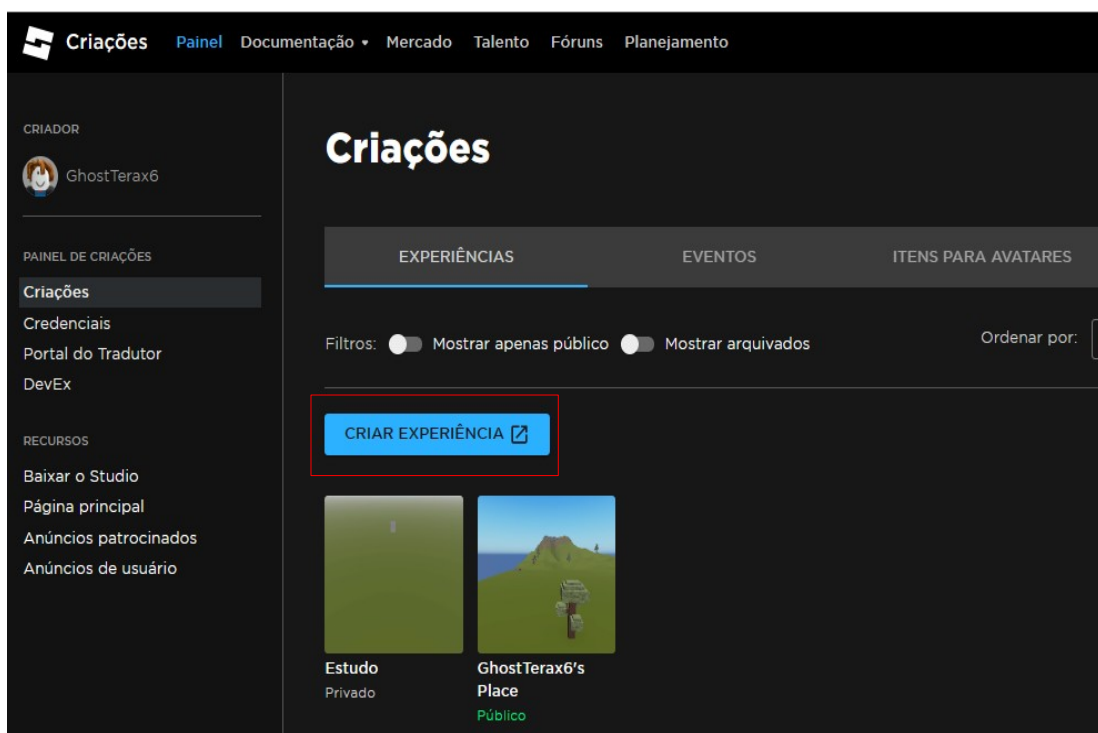
Para assegurar que seus modelos não sejam perdidos e para facilitar o acesso a todos os recursos da plataforma, é altamente recomendável salvar suas criações diretamente no site. Ao fazer isso, você poderá usufruir de diversas funcionalidades de maneira facilitada,

incluindo a adição de paisagens, imagens clicáveis e objetos de interface.

Antes de mais nada, é necessário criar suas experiências. Para isso, existem duas formas: uma delas foi demonstrada nas sessões anteriores, e a segunda pode ser realizada ao se acessar o link <https://create.roblox.com/dashboard/creations>. Caso não tenha feito login na plataforma, você deve clicar no botão **Entrar na Conta**, que se encontra na mesma posição do botão evidenciado no canto superior direito da Figura 1.

Após clicar no botão, você deve preencher seu usuário e senha, assim como demonstrado na Figura 4. Após efetuar *login* na plataforma, uma tela semelhante a demonstrada na Figura 9 será exibida. Nela, você deve clicar no botão **CRIAR EXPERIÊNCIA**, destacado na imagem pelo retângulo vermelho.

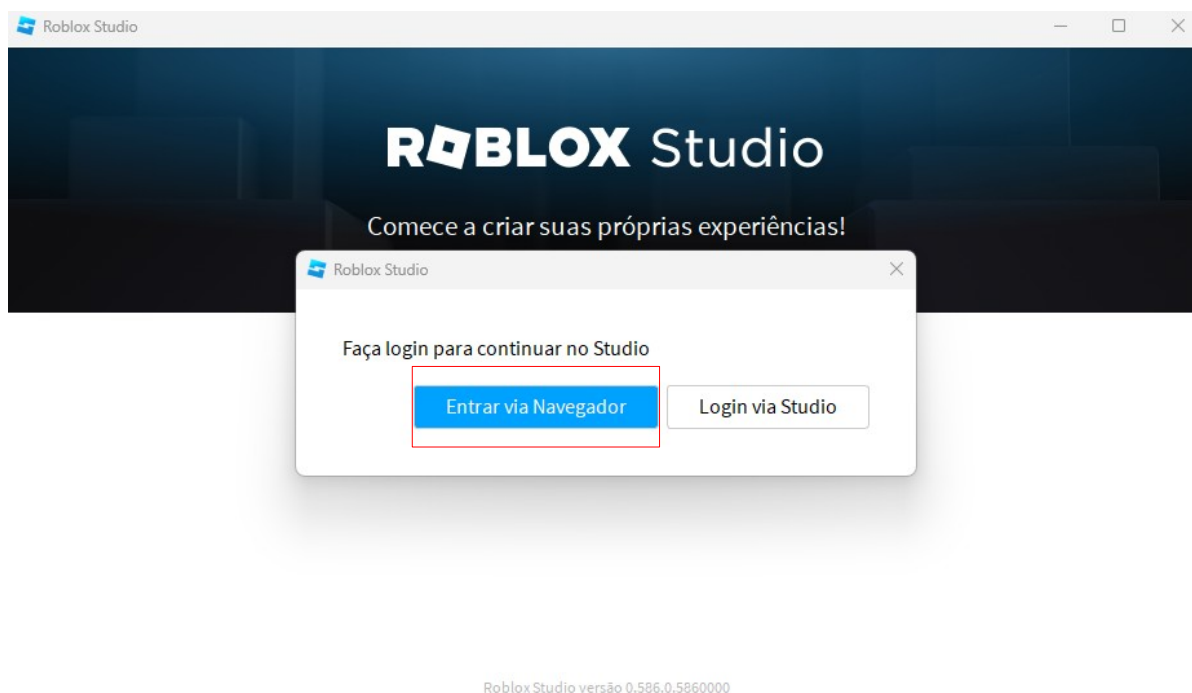
Figura 9 – Criação de Experiência



Fonte: O autor, 2023.

Clicando no botão destacado, uma mensagem semelhante à apresentada na Figura 2 será exibida. Dessa forma, você deve clicar em **abrir**. Caso não tenha instalado o Roblox Studio ainda, retorne à Seção 1 e realize a instalação. Após clicar no botão, o Roblox Studio abrirá, e caso você não tenha realizado login durante a instalação dele, uma tela semelhante à demonstrada na Figura 10 será apresentada.

Figura 10 – Login no Roblox Studio

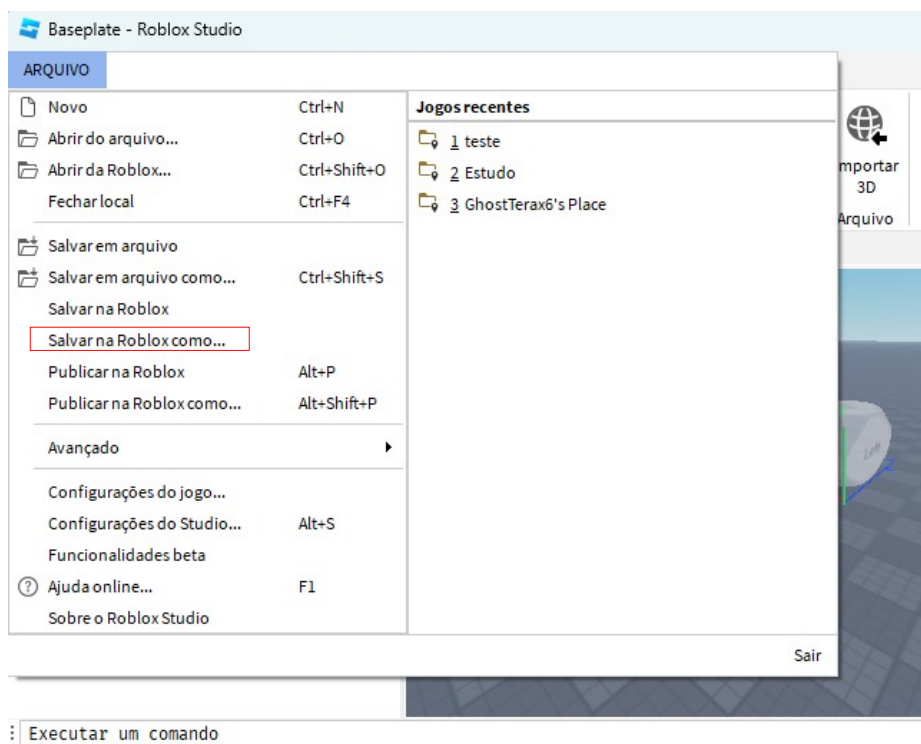


Fonte: O autor, 2023.

Para garantir um login com sucesso, selecione **Login via Studio**, destacado pelo retângulo vermelho. Você deverá seguir os passos evidenciados na Figura 4 e, em seguida, selecionar o *template* de sua escolha.

Ambas as formas de se criar experiências possuem a mesma eficiência, basta escolher com a qual melhor se adapta. Porém, o ponto mais importante desta seção é destacar a necessidade de se salvar a experiência criada na Roblox. Para isso, você deve selecionar, no canto superior esquerdo, o menu **Arquivo**, o qual exibirá uma lista de opções, assim como demonstrado na Figura 11. Nesta lista, você deve clicar na opção **Salvar na Roblox como....**, destacada na figura com o retângulo vermelho.

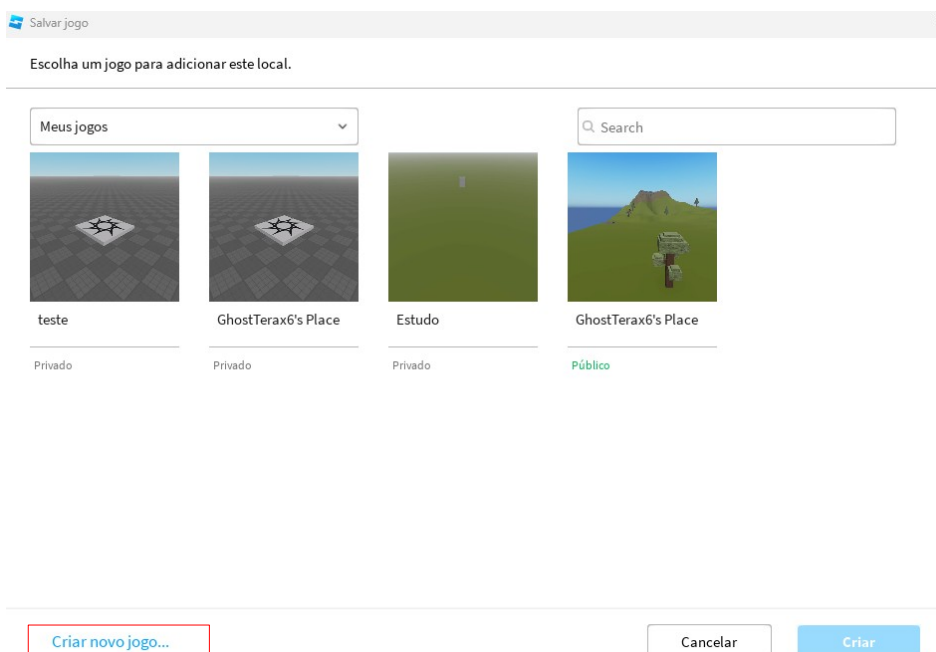
Figura 11 – Salvar na Roblox



Fonte: O autor, 2023.

Em seguida a tela evidenciada pela Figura 12 será exibida. Nesta tela, você deve clicar em **Criar novo jogo....**

Figura 12-Criar novo Jogo



Fonte: O autor, 2023.

2.3 Comandos Básicos

O Roblox Studio possui alguns atalhos para auxiliar na construção das experiências e facilitar a manipulação do ambiente. Dentre eles, podem-se destacar as teclas:

- **W, A, S, D:** movimenta a câmera para a frente, esquerda, para trás e para a direita, respectivamente;
- **Q, E:** movimenta a câmera para baixo e para cima, respectivamente;
- **Shiftb** diminui a velocidade de movimentação da câmera ao ser pressionado em conjunto com algum botão de movimento, por exemplo, shift + W;
- **F:** movimenta a câmera instantaneamente para o objeto selecionado, ou seja, centraliza o objeto no ambiente de visualização.

Além das teclas de atalho, o mouse também auxilia na manipulação dos objetos para construção da experiência. Seus botões possuem as seguintes funções:

- **Botão Direito:** quando pressionado, permite rotacionar a visualização do ambiente ao redor, ou seja, muda a perspectiva de visualização do ambiente de desenvolvimento;
- **Scroll:** também conhecida como “rodinha” do mouse, ao ser rotacionada, permite aumentar ou diminuir o zoom, possibilitando visualizar detalhes com maior nitidez;
- **Pressionar o Scroll:** pressionar o Scroll e movimentar o mouse permite arrastar a câmera, sem que haja necessidade de pressionar os botões de movimentação.

3 Repositórios de Modelos Prontos

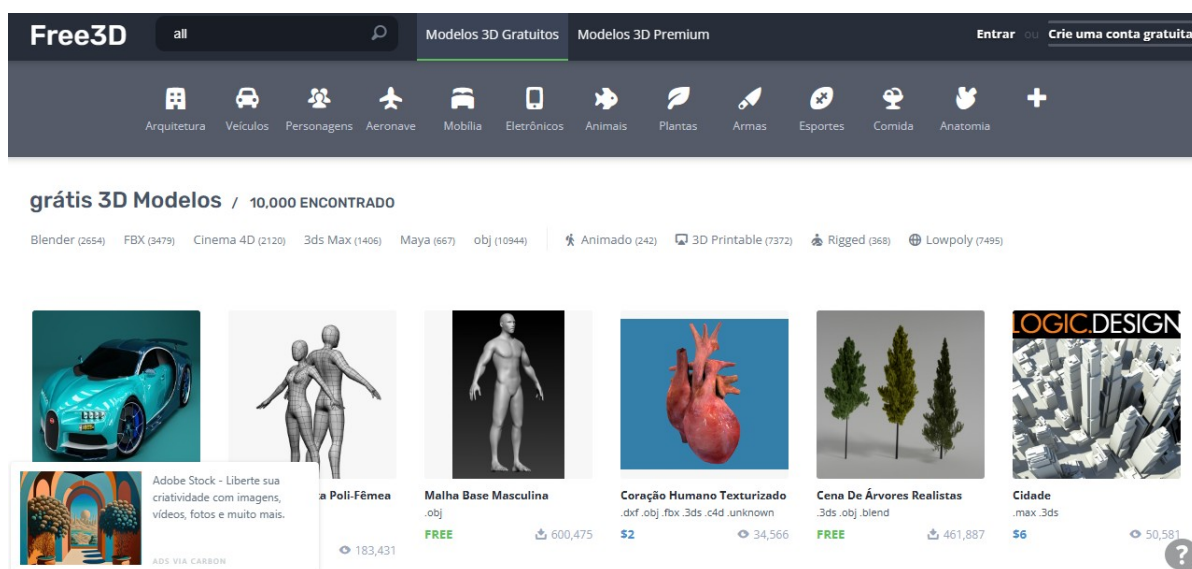
O Roblox studio suporta objetos 3D no formato **.fbx** e **.obj**, sendo que esses sufixos de arquivos constituem uma espécie de rótulo que descreve o tipo ou o formato de um arquivo. Ele é uma convenção utilizada em sistemas operacionais para facilitar a abertura desse arquivo pelo programa que o gerou.

Suponhamos que um arquivo de texto tenha o nome **Teste**. Ao observar a maneira como esse arquivo está disponível em seu computador, você verá algo como **Teste.txt**, o que facilitará para o sistema identificar qual programa é o adequado para abri-lo. Isso não é diferente para modelos 3D, que, após seu *download*, terão nomes como **modelo3D.fbx**. Esses modelos podem ser encontrados em diversas plataformas, e algumas delas encontram-se nas seções a seguir.

3.1 Free3d

O Free3D é um repositório com uma grande variedade de modelos 3D prontos para download. O site possui mais de 10.000 modelos gratuitos, que vão desde arquitetura à anatomia. Nessa plataforma, é possível encontrar modelos em diversos formatos; dentre eles, os suportados pelo Roblox Studio, assim como evidenciado na Figura 15.

Figura 15 – Tela do *website* Free3D



Fonte: O autor, 2023.

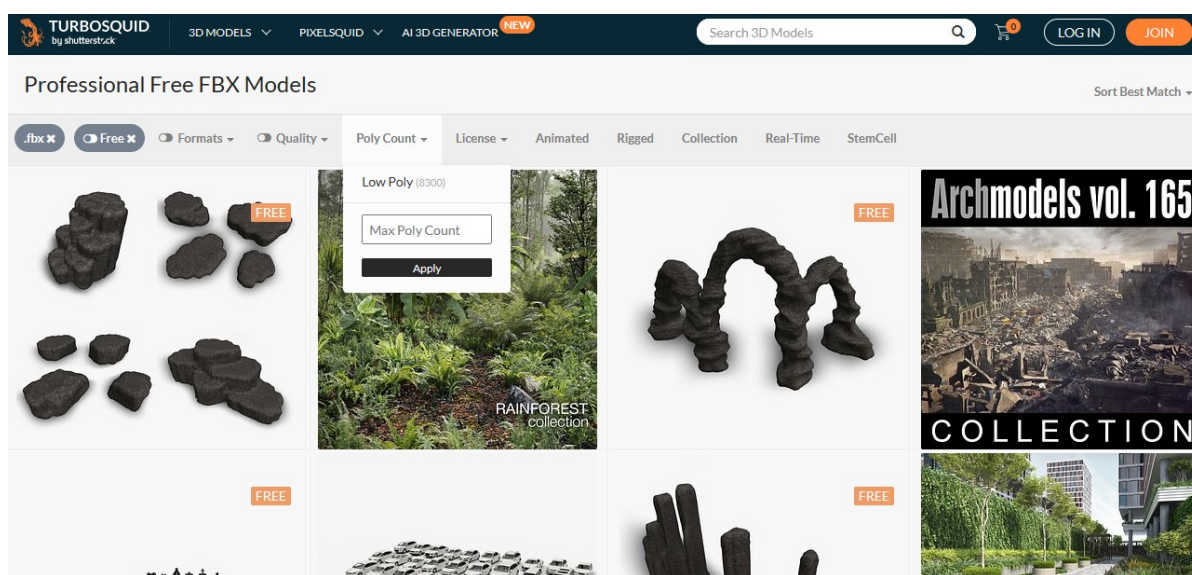
O Free3D pode ser acessado através do link: <https://free3d.com/pt/>. Caso você tenha

afinidade com a construção de modelos 3D, esse *website* disponibiliza modelos construídos pela comunidade externa, viabilizando a disponibilização de seu modelo para outros desenvolvedores de experiências com modelos 3D.

3.2 TurboSquid

Semelhantemente ao Free3D, o TurboSquid também é um repositório de modelos 3D. Esse *website* possui tanto modelos pagos quanto gratuitos, mas se diferencia do Free3D pelo filtro de quantidade de polígonos dos modelos. A quantidade de polígonos influencia em quão detalhado é o objeto 3D; porém, quanto mais polígonos, maior o custo computacional necessário para manipular esse objeto, tanto na etapa de construção quanto na de utilização da experiência. O Roblox Studio suporta modelos com até 15.000 polígonos. A Figura 16 ilustra a aparência desse repositório de modelos 3D.

Figura 16 – Tela do *website* TurboSquid



Fonte: O autor, 2023.

O repositório pode ser acessado através do link: <https://www.turbosquid.com/>.

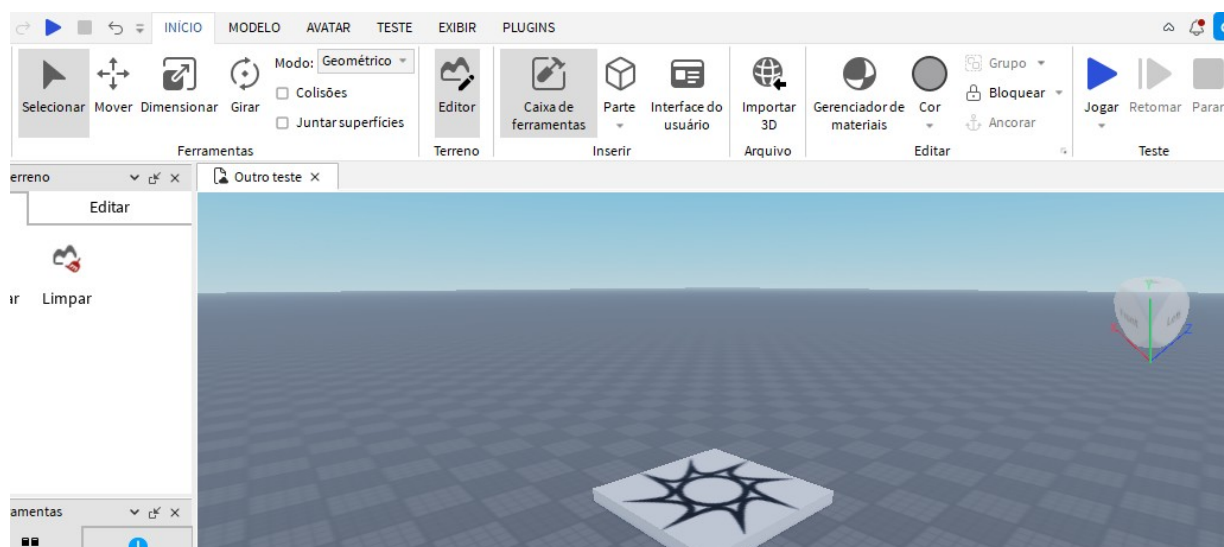
3.3 Como Incluir Modelos no Projeto

Para incluir modelos prontos no projeto, você deve acessar algum repositório ou construir seu próprio modelo, respeitando o padrão aceito pelo Roblox Studio, o .fbx ou .obj. Utilizando modelos prontos, você pode acessar algum dos repositórios descritos na seção anterior e buscar pelo item que deseja.

Para incluir o modelo na sua experiência do Roblox, basta clicar no botão **Importar 3D** do menu de ferramentas do Roblox Studio, como demonstrado na Figura 17. Em seguida, basta localizar o modelo em seu computador e clicar no botão **abrir**, conforme a Figura 18.

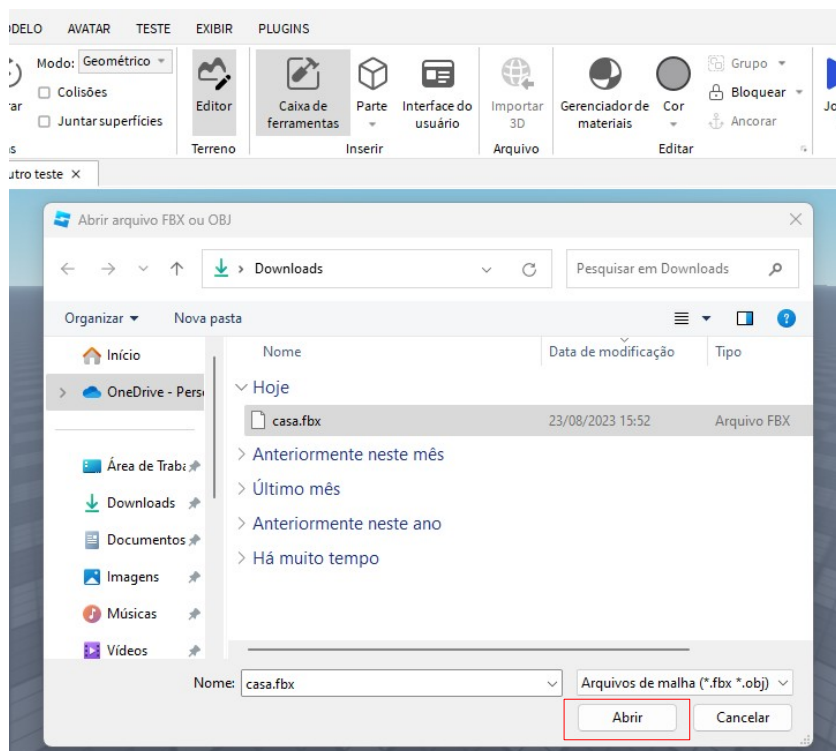
Após clicar no botão **abrir**, será exibida uma tela como apresentado na Figura 19, em que se deve clicar no botão **Importar**, destacado pelo retângulo vermelho.

Figura 17 – Importar modelo 3D no Roblox



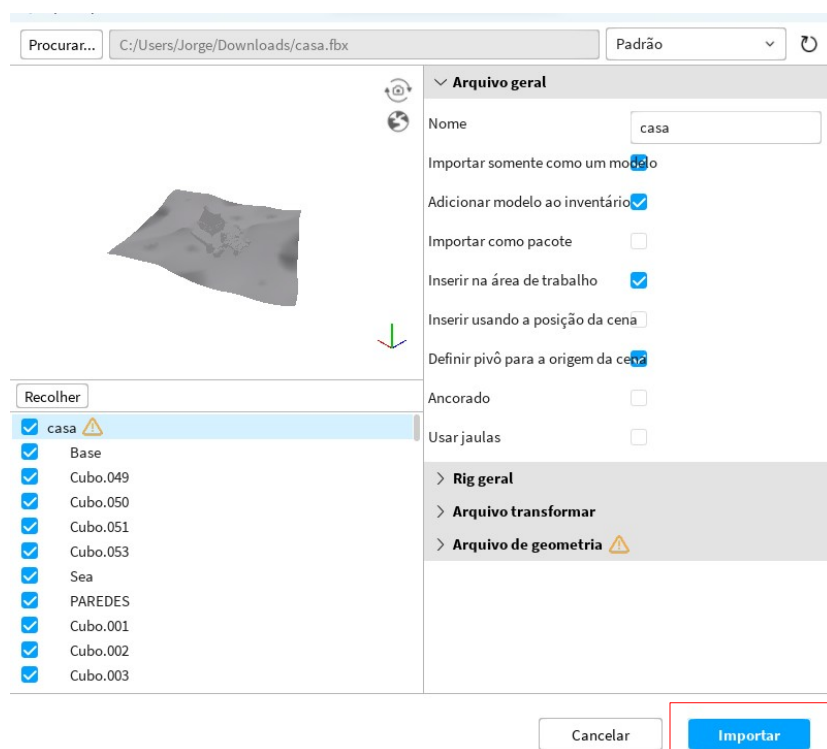
Fonte: O autor, 2023.

Figura 18 – Importar modelo 3D no Roblox



Fonte: O autor (2023).

Figura 19 – Importar Modelo

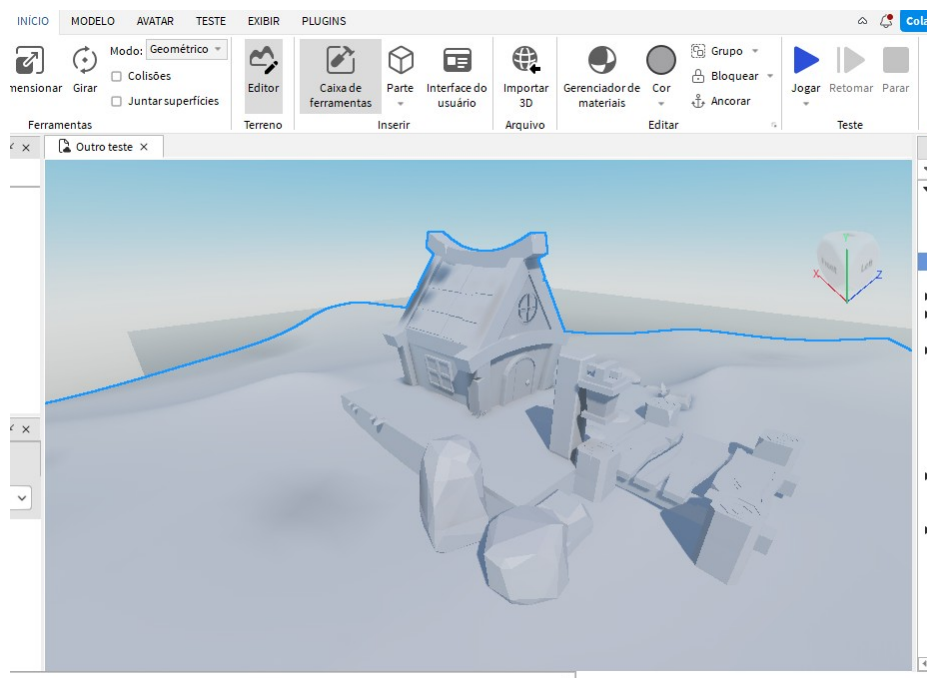


Fonte: O autor, 2023.

Após a importação, será possível visualizar o modelo e manipulá-lo, assim como

demonstra a Figura 20.

Figura 20 – Modelo Importado



Fonte: O autor, 2023.

4 Scripts

No contexto do Roblox Studio, *scripts* são trechos de código escritos em Lua, uma linguagem de programação na qual eles são usados para controlar o comportamento de objetos, personagens e cenários dentro de experiências criadas na plataforma Roblox. Os *scripts* são uma parte essencial do desenvolvimento no Roblox, pois permitem aos desenvolvedores criar interatividade e funcionalidades específicas para seus jogos.

Dentre as principais características dos *scripts* no Roblox Studio, têm-se:

- **Lua:** Roblox Studio utiliza a linguagem de programação Lua para escrever *scripts*. Lua é uma linguagem versátil e é frequentemente usada em jogos devido à sua simplicidade.
- **Eventos e Funções:** os *scripts* no Roblox Studio, frequentemente, respondem a eventos, como o clique do mouse ou a colisão de objetos. Eles também podem conter funções que são chamadas em momentos específicos para realizar ações desejadas.
- **Manipulação de Objetos:** os *scripts* podem ser usados para controlar o movimento, a aparência e o comportamento de objetos no jogo, como personagens, veículos, itens, entre outros.
- **Lógica de Jogo:** os *scripts* são essenciais para implementar a lógica do jogo, incluindo mecânicas de jogabilidade, regras, pontuação e outras funcionalidades específicas do jogo.
- **Servidores e Clientes:** no contexto de jogos multiplayer no Roblox, é comum ter *scripts* no lado do servidor (*Server Scripts*) e *scripts* no lado do cliente (*Local Scripts*). Os *scripts* no lado do servidor lidam com lógica de jogo que precisa ser executada no servidor, enquanto os *scripts* no lado do cliente lidam com a interação do jogador e a exibição de informações.

Os desenvolvedores do Roblox usam o Roblox Studio para criar e editar *scripts*, incorporando-os em seus jogos para criar experiências interativas e envolventes. O Roblox Studio fornece uma interface gráfica intuitiva para criar e manipular *scripts*, tornando o processo de desenvolvimento acessível a uma ampla gama de usuários, desde iniciantes até desenvolvedores mais experientes.

4.1 Como e Onde Incluir Scripts

No Roblox Studio, você pode incluir *scripts* para controlar o comportamento de objetos, personagens e outros elementos do seu jogo. Para isso, você deve incluí-los no Roblox Studio da seguinte maneira:

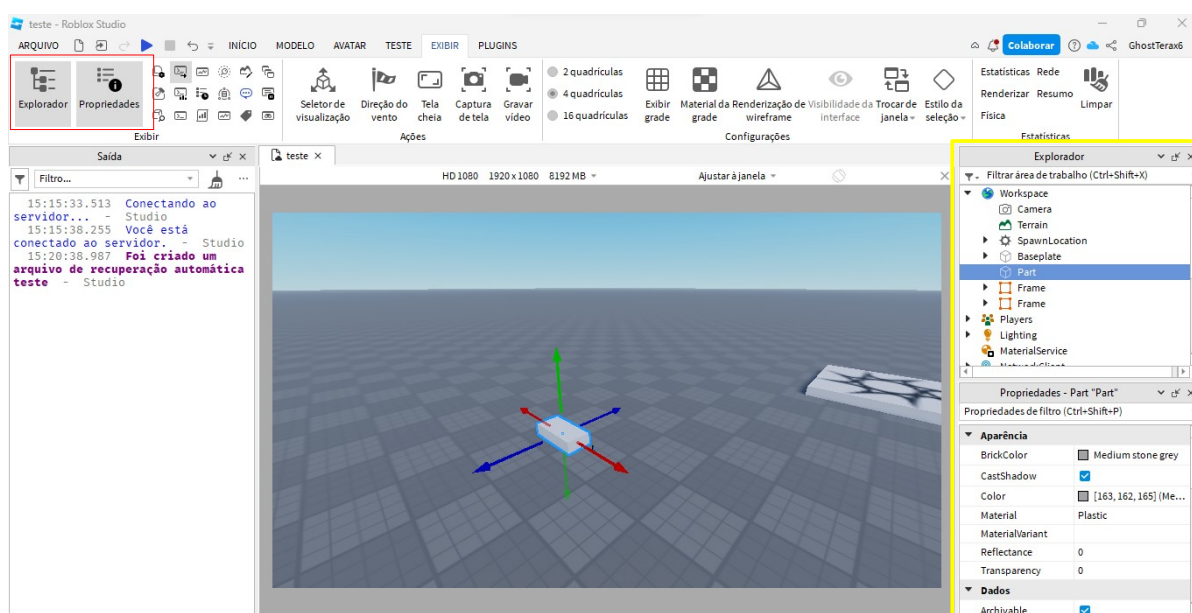
1. Abrir o Roblox Studio:

Inicie o Roblox Studio e abra a experiência que você deseja incluir *scripts*.

2. Abrir a Janela Explorador e Propriedades:

Certifique-se de ter as janelas **Explorador** e **Propriedades** abertas. Você pode encontrá-las na barra de menus **Exibir**, assim como demonstrado no retângulo vermelho, na Figura 21. Essas funcionalidades estarão abertas se o plano de fundo das janelas estiver com a cor acinzentada, e será possível visualizá-las na lateral direita do ambiente, assim como destacado no retângulo amarelo.

Figura 21 – Exibição de janelas

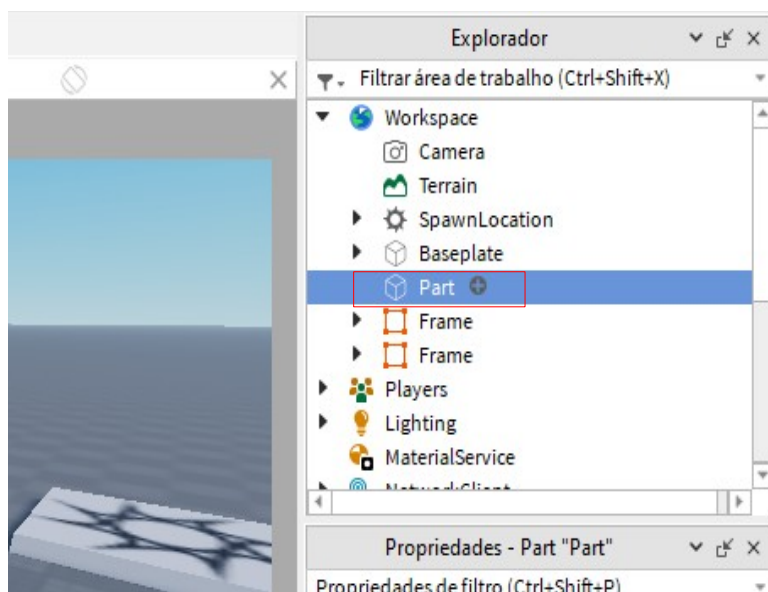


Fonte: O autor, 2023.

3. Selecionar o Objeto alvo:

No **Explorador**, selecione o objeto ao qual deseja adicionar o *script*. Ele pode ser um bloco (part), um personagem, um veículo ou qualquer outro objeto no seu jogo. Ao selecioná-lo, sua cor de fundo ficará azulada.

Figura 22 – Seleção de um objeto “part”

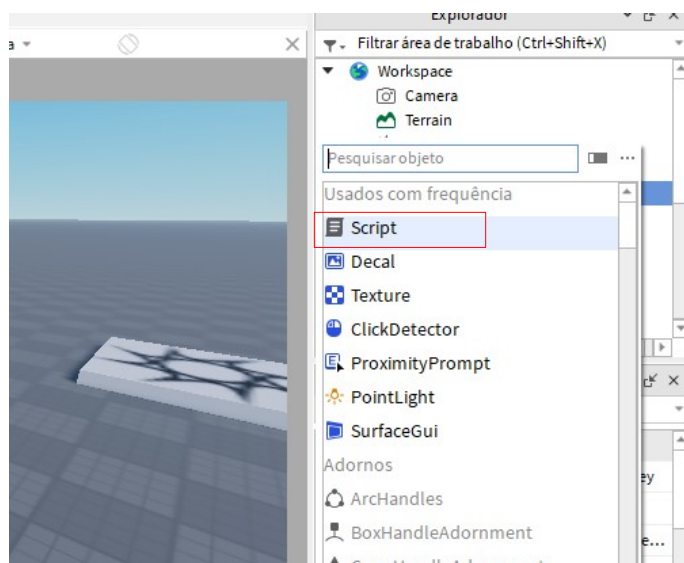


Fonte: O autor, 2023.

4. Adicionar um Script:

Com o objeto selecionado e o cursor sobre o seu nome, um ícone de + irá aparecer ao lado direito do objeto; clique sobre o + e escolha **Script** na lista, assim como demonstrado na Figura 23.

Figura 23 – Adição de Script

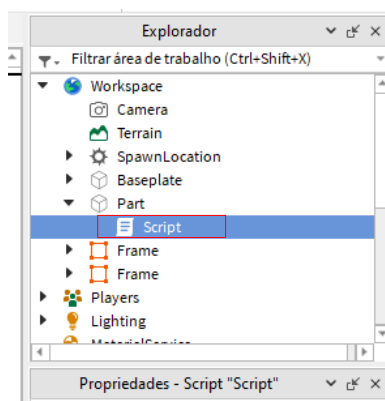


Fonte: O autor, 2023.

5. Editar o Script:

Ao adicionar o *script*, uma nova guia chamada **Script** aparecerá no **Explorador**. Clique duas vezes no *script* para abrir o editor de *scripts*, onde você pode escrever ou colar seu código Lua.

Figura 24 - Testar a Experiência



Fonte: O autor, 2023.

6. Escrever o Código Lua:

Dentro do editor de *scripts*, você pode começar a escrever seu código Lua. Este código será efetivado quando o jogo estiver em execução e o evento associado ao *script* for acionado.

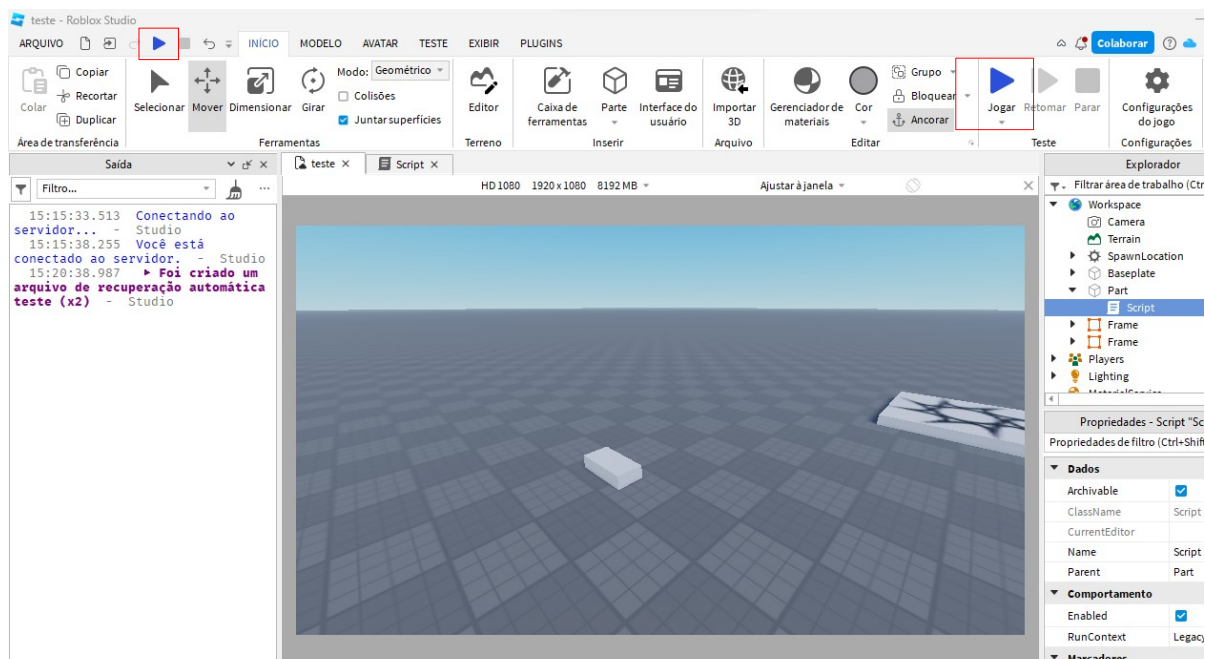
7. Salvar e Testar:

Certifique-se de salvar o lugar após adicionar ou editar *scripts*. Você pode fazer isso clicando em **Arquivo** e selecionando **Salvar**.

Para testar o *script*, clique em **Jogar**, na barra de menus, para iniciar a execução do jogo e observar o comportamento do *script*, assim como demonstrado nos retângulos vermelhos, na Figura 25. Lembre-se de que, dependendo do que você deseja alcançar, pode ser necessário adicionar *scripts* em diferentes locais. Além disso, os *scripts* podem ser associados a eventos específicos para controlar a lógica do jogo de maneira mais dinâmica.

Existem dois botões para executar a experiência no menu **Início**, os quais possuem a mesma funcionalidade e resultado.

Figura 25 – Exibição de janelas



Fonte: O autor, 2023.

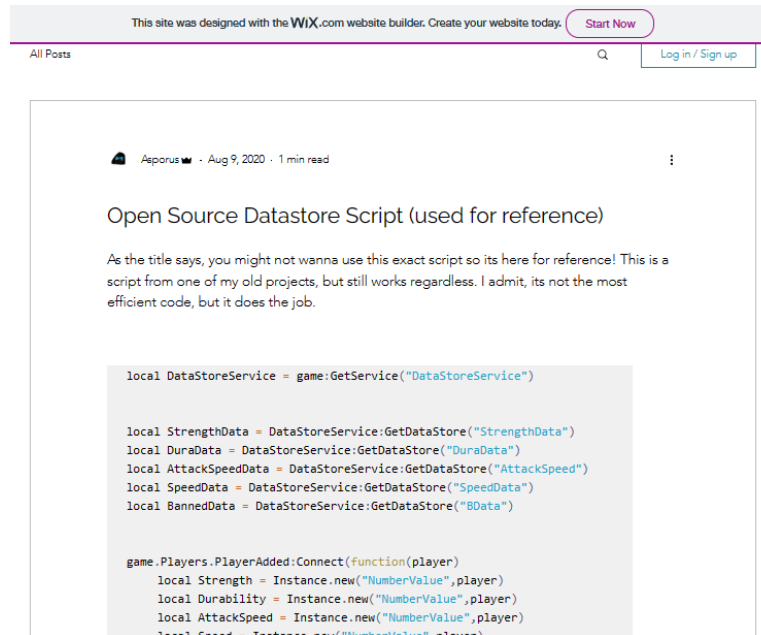
4.2 Repositórios de Scripts

Para facilitar o processo de escrita de *scripts*, existem, atualmente, dois repositórios que oferecem exemplos de código. Um deles é a *Open Source Library | Asporus*, que está em inglês e inclui seis exemplos de *scripts*.

O segundo repositório é o *Haddock - Roblox Library*, que disponibiliza um total de cento e dois exemplos de *scripts* de forma gratuita. O diferencial do *Haddock* em comparação com o *Asporus* é a presença de um chat interativo, baseado no ChatGPT 4.0, com foco exclusivo na geração de *scripts* para o Roblox Studio. Porém, esse chat é pago. As Figuras 26 e 27 ilustram esses repositórios.

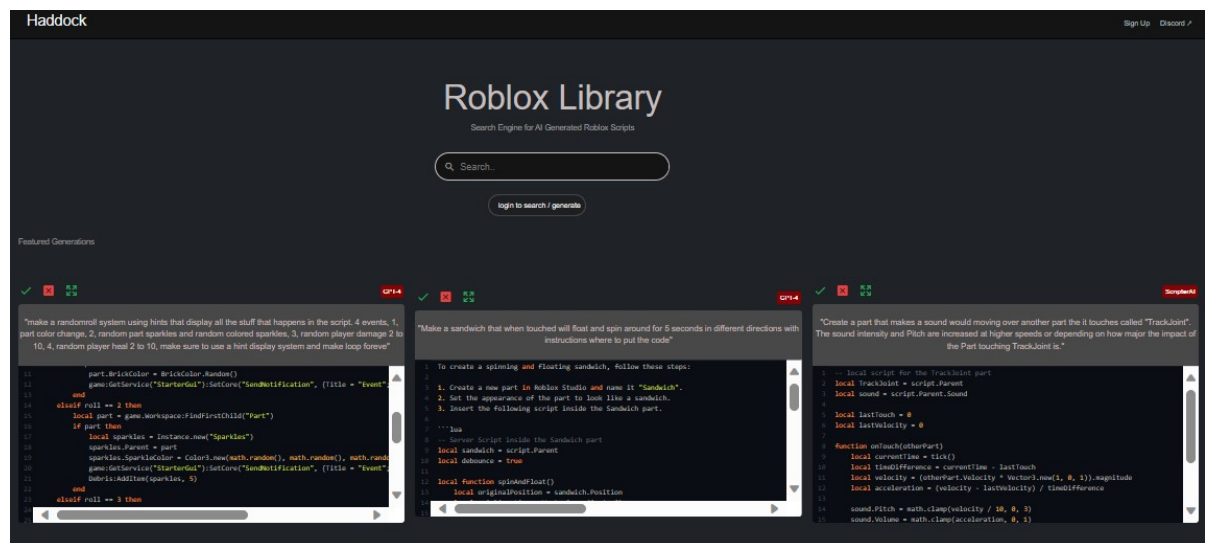
Para acessar o *Asporus*, visite [Open Source Library | Asporus \(licksaw23.wixsite.com\)](https://www.licksaw23.wixsite.com/asporus). Para explorar o *Haddock - Roblox Library*, acesse [Haddock - Roblox Library](https://www.licksaw23.wixsite.com/haddock-roblox-library). Esses recursos são valiosos para quem busca aprimorar suas habilidades de scripting no Roblox Studio.

Figura 26 – Repositório de Scripts, Haddock



Fonte: O autor (2023).

Figura 27 – Repositório de Scripts, Haddock



Fonte: O autor, 2023.

5 Como Criar uma Experiência 3D no Roblox

A experiência descrita a seguir trata-se de uma exploração do Roblox Studio como ferramenta para a construção de OVAs (Objetos Virtuais de Aprendizagem) e aborda o uso dessa plataforma como meio de ensinar o conceito de mudanças de estado físico.

Ao empregar o Roblox Studio para criar OVAs, é possível proporcionar uma abordagem prática e envolvente para a compreensão de conceitos complexos, como as mudanças de estado físico. A plataforma oferece recursos versáteis que permitem a criação de ambientes virtuais interativos nos quais os usuários podem explorar e experimentar de maneira dinâmica.

Durante esta exploração, serão destacadas as funcionalidades específicas do Roblox Studio que facilitam a representação e a interação com os conceitos relacionados às mudanças de estado físico. Vamos mergulhar nessa jornada pelo Roblox Studio e descobrir como ele pode ser uma ferramenta eficaz na construção de Objetos Virtuais de Aprendizagem.

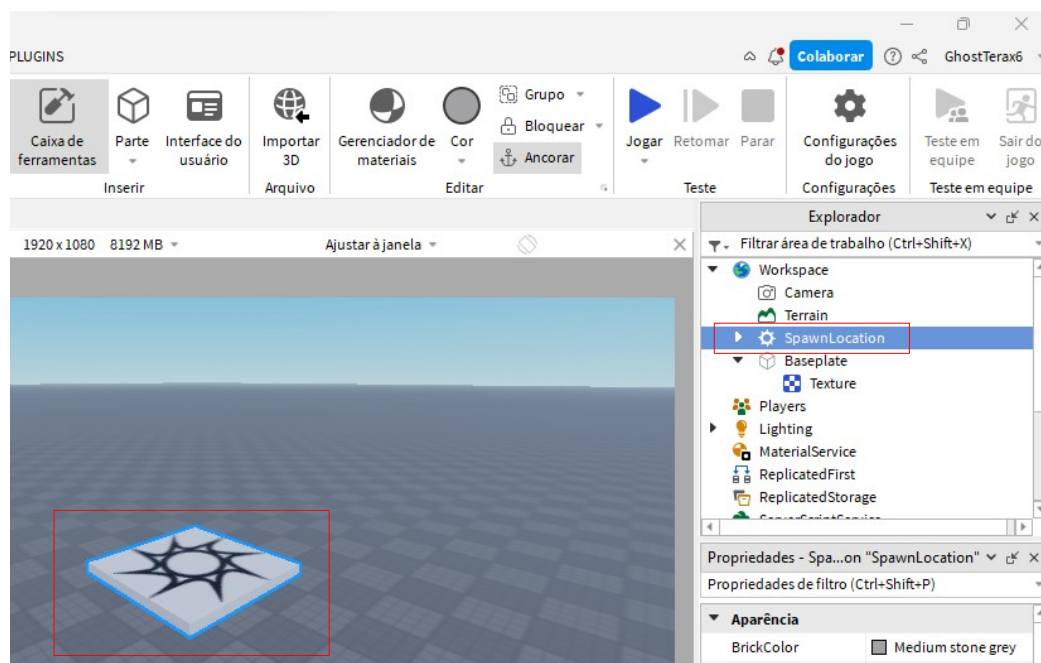
O primeiro passo na construção desta experiência é compreender como e onde o jogador aparecerá no mapa. O local de *spawn* do jogador é denominado ***spawnlocation***, que pode ser localizado na aba do **Explorador** como um filho, ou seja, um objeto que está aninhado dentro do **WorkSpace**. Para acessá-lo, clique na seta no lado esquerdo do **WorkSpace** e procure por ***spawnlocation***, como mostrado na Figura 28.

Na Figura 28, é possível visualizar o local de aparecimento do jogador tanto no **Explorador** quanto no ambiente de desenvolvimento, ambos destacados com o retângulo vermelho. Isso serve como uma referência visual para garantir que o ponto de *spawn* esteja configurado conforme desejado. Este é um passo fundamental para definir o ponto inicial da experiência do jogador no ambiente virtual que está sendo construído.

Após compreender onde o jogador aparecerá, você pode iniciar a construção da sua experiência, delimitando a área a ser utilizada por ele. Isso pode ser realizado inserindo-se objetos, como paredes. Para adicionar uma parede, acesse o menu **Exibir** e certifique-se de que a caixa de ferramentas está selecionada, conforme mostrado na Figura 29.

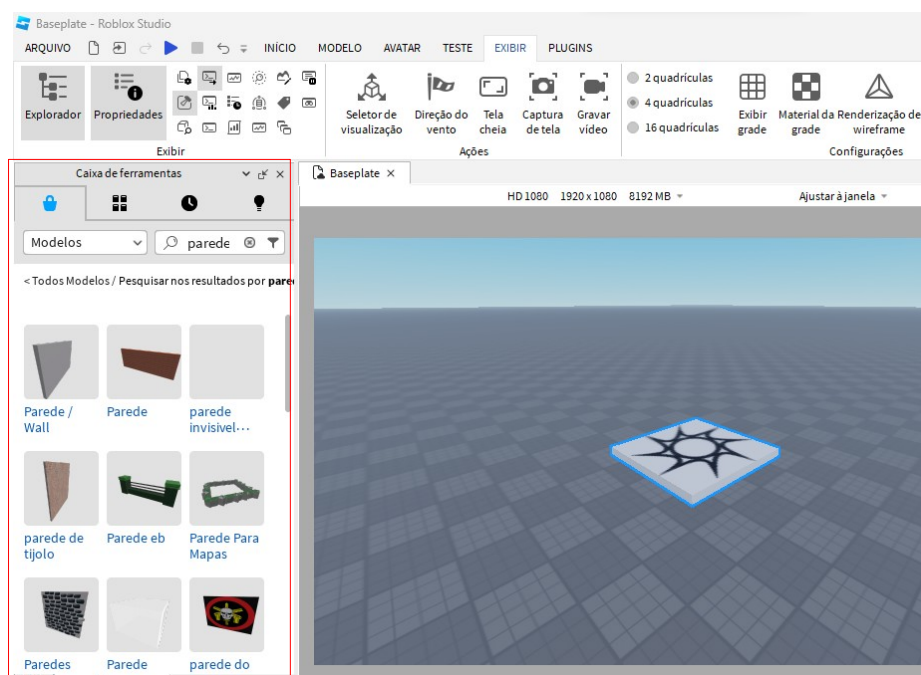
Em seguida, pesquise o nome do objeto desejado e, ao encontrá-lo, arraste-o para a área da experiência. Alternativamente, você pode inseri-lo seguindo as instruções descritas na seção 3.3. Essas paredes ajudarão a definir os limites do espaço disponível para o jogador, proporcionando uma estrutura clara para a experiência. Certifique-se de posicionar e dimensionar esses objetos de acordo com a visão geral do design que você deseja para a interação do jogador. Essa etapa é crucial para criar um ambiente controlado e envolvente.

Figura 28 – Local de Aparecimento do Jogador



Fonte: O autor (2023).

Figura 29 – Inserção de objetos via Caixa de Ferramentas

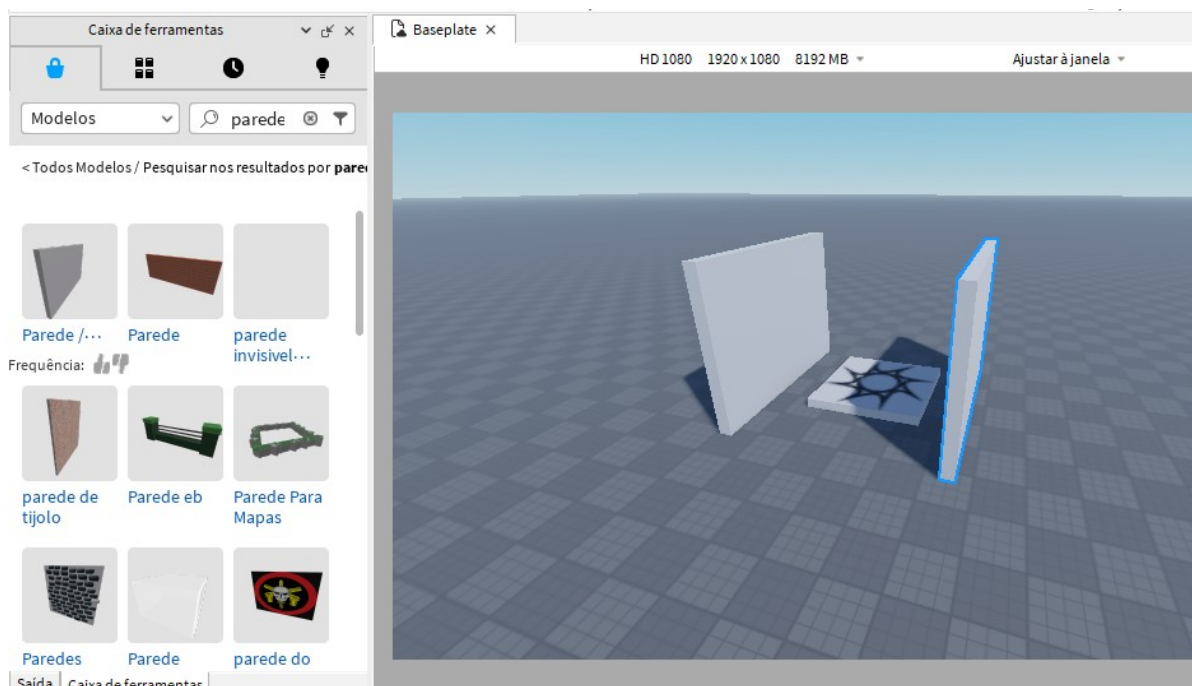


Fonte: O autor, 2023.

Ao arrastar os Objetos para a cena, você terá uma visão parecida com a demonstrada na Figura 30. Você tem a liberdade para trabalhar o cenário da forma que preferir; apenas não se esqueça de ancorar os objetos. Para isso, você deve selecionar o objeto e, no menu **Início**, selecionar a opção **ancorar** - isso faz com que os objetos permaneçam no local que

você delimitou.

Figura 30 – Inserção de objetos via Caixa de Ferramentas

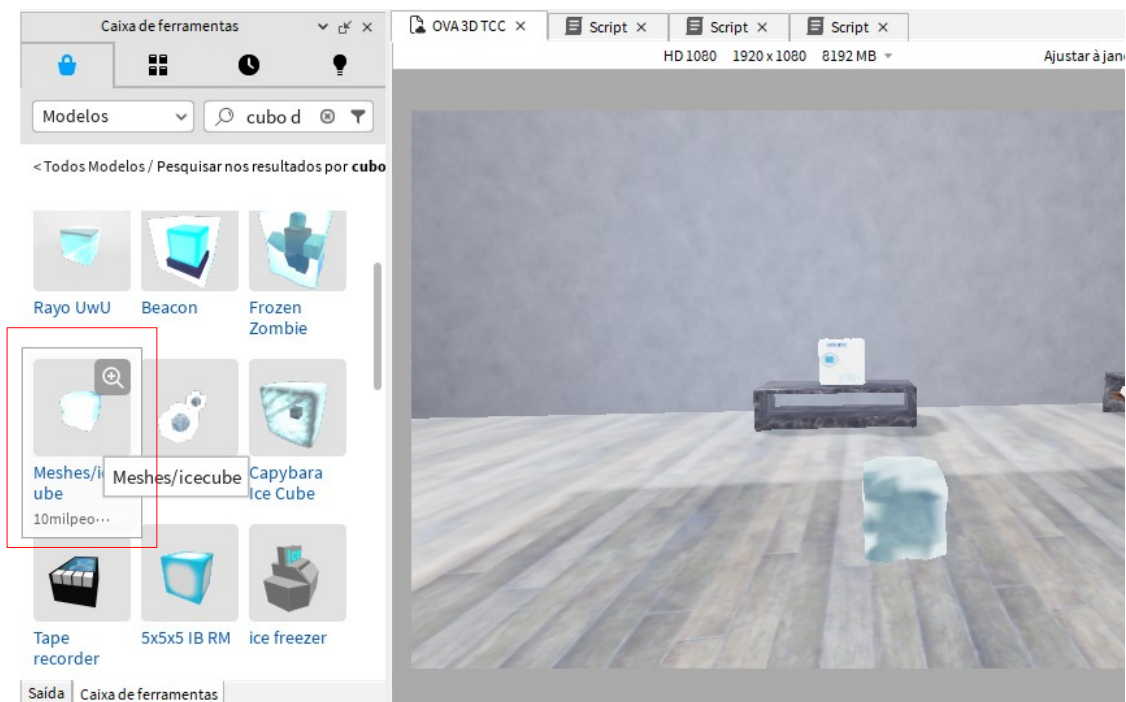


Fonte: O autor, 2023.

Após a concepção do cenário desejado, é necessário incorporar as ações que o personagem executará ao interagir com objetos específicos. Recomenda-se a construção de um ambiente semelhante à representação visualizada na Figura 31, o qual servirá como base para o desenvolvimento das demais funcionalidades abordadas neste tutorial. Esse cenário é constituído de quatro paredes estruturais, ancoradas para delimitar um espaço retangular, além de um teto e um piso. Os elementos do teto e do piso podem ser os mesmos utilizados nas paredes, bastando uma rotação de 90° e uma alocação que complemente o espaço construído.

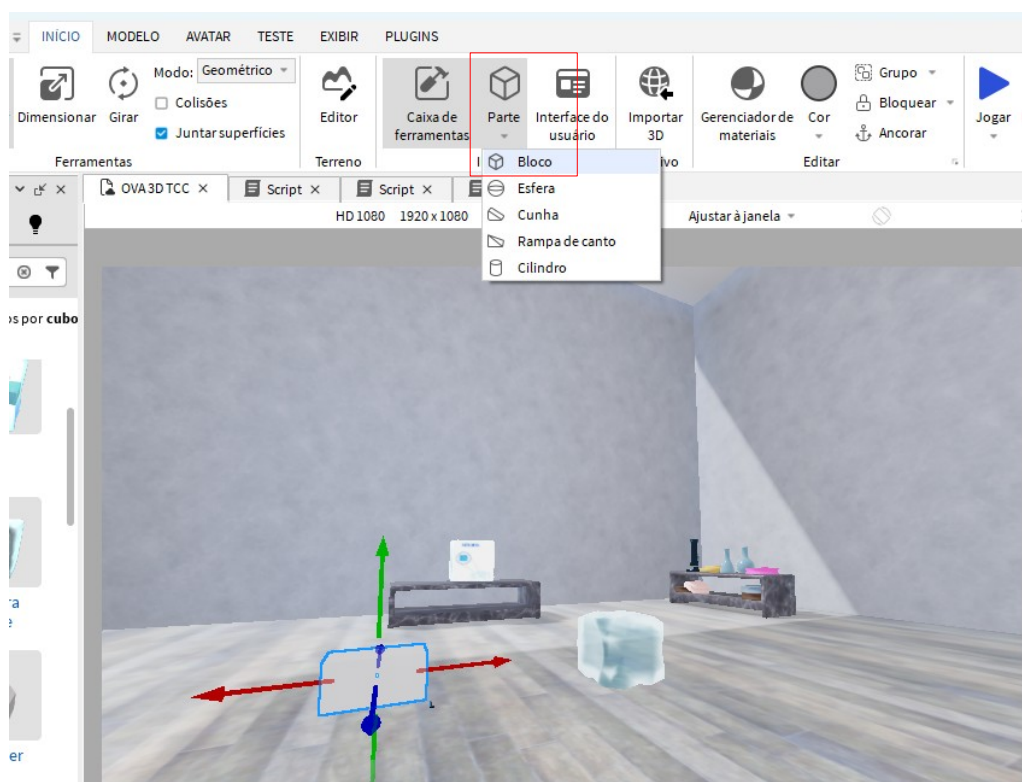
Além disso, o cenário possui três mesas e algumas estantes, de forma que se assemelhe a um laboratório de química. Esses objetos podem ser encontrados da mesma forma que as paredes, bastando buscá-los na “Caixa de Ferramentas”. Você possui liberdade criativa para construir seu cenário.

Figura 32 – Inserção de cubo de gelo



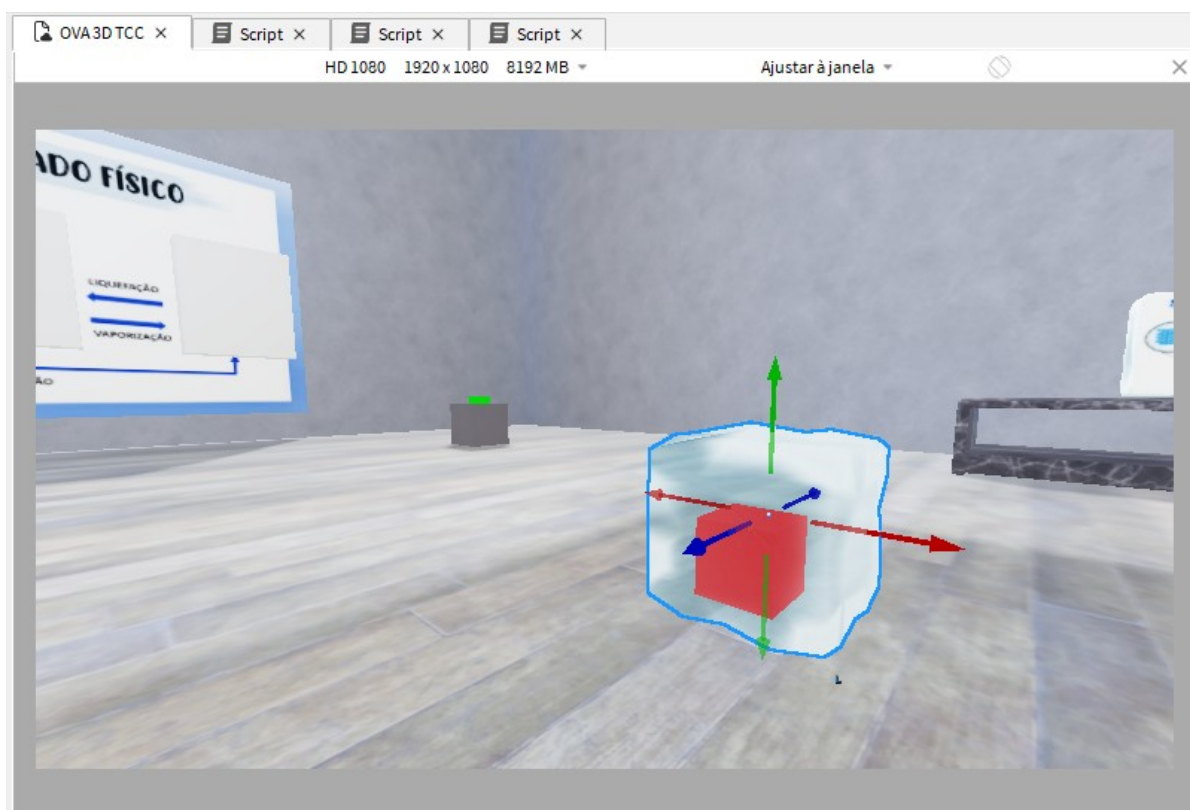
Fonte: O autor, 2023.

Figura 33 – Inserção de retângulo



Fonte: O autor, 2023.

Figura 34 – Inserção de retângulo

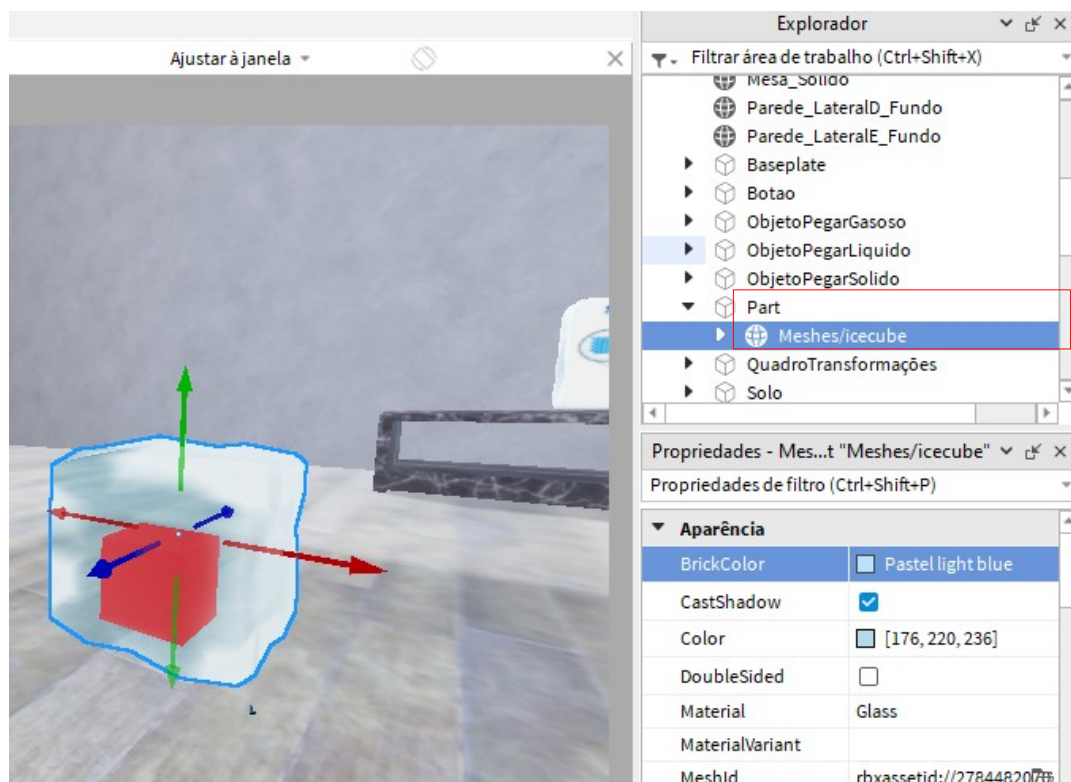


Fonte: O autor, 2023.

Após posicionar visualmente o cubo de gelo sobre o **bloco**, na janela de cenário, localize os componentes denominados **Meshes/icecube** e **Part** na janela **Explorador**. Você pode identificá-los clicando sobre seus nomes; eles serão destacados com um contorno azul no cenário, como exemplificado na Figura 34 do cubo de gelo. Ao identificar esses itens, arraste o componente **Meshes/icecube** e posicione-o sobre a **Part**, de modo que o primeiro se torne um subcomponente do segundo na hierarquia do **Explorador**, conforme ilustrado na Figura 35. Ao realizar esse procedimento, você cria um objeto único, onde o cubo de gelo funciona como uma espécie de revestimento para o **Bloco**, permitindo interações com o jogador.

Caso deseje mudar o nome do objeto **part**, você pode clicar sobre ele com o botão direito do mouse, e clicar em **Renomear**, em seguida digitar o nome que desejar, uma sugestão é utilizar **Cubo de Gelo**.

Figura 35 – Mudança de Hierarquia de objetos



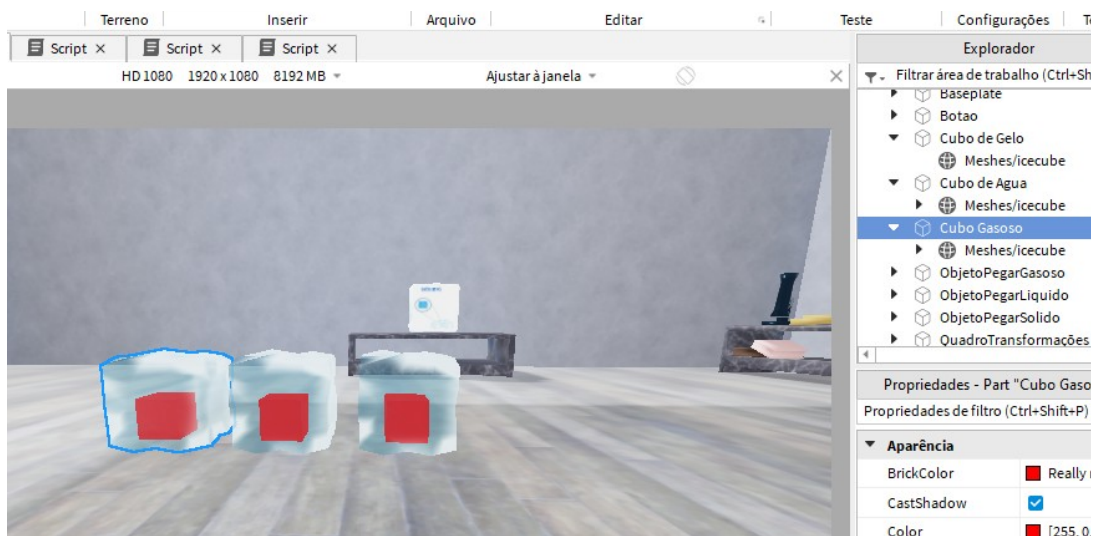
Fonte: O autor, 2023.

Para evitar retrabalho, você pode duplicar o objeto em questão. Basta novamente clicar com o botão direito do mouse sobre ele e clicar em **Duplicar**, ou apertar o atalho **ctrl+d**. Faça isso de forma que tenha como resultado final três cubos de gelo, como evidenciado na Figura 36. Você pode renomear os cubos como preferir, uma sugestão é chamá-los de **Cubo de Gelo**, **Cubo de Água** e **Cubo de Gás**.

O próximo passo para a construção da experiência é colocar, na superfície dos cubos, o que o Roblox chama de **Decal**, uma imagem. Nesse ponto, você já deve ter compreendido o porquê de serem criados três cubos, um para cada estado da matéria. Essa imagem pode ser colocada da mesma forma que as paredes da experiência, ou seja, você deve buscar, na **Caixa de Ferramentas**, imagens para serem posicionadas sobre as faces dos cubos. Isso pode ser feito alterando-se o *dropbox* para imagens e buscando por **water**. Sugere-se a busca em inglês por haver maior abundância de componentes com nomes nesse idioma, como ilustrado na Figura 37.

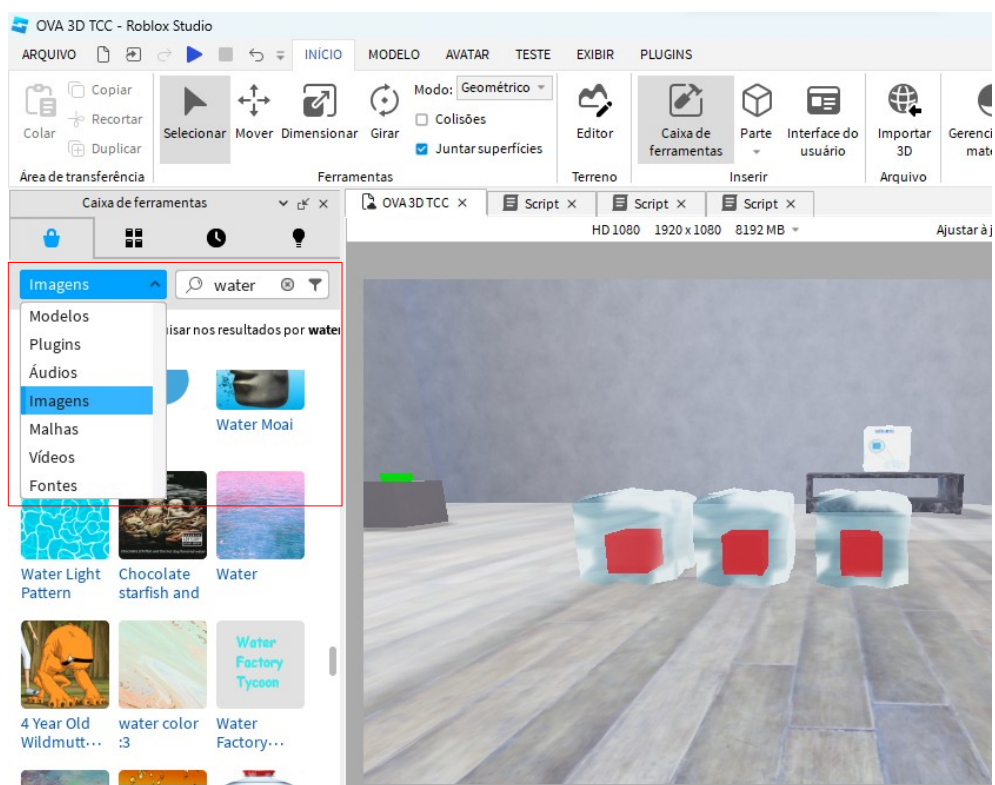
Ao fazer isso, utilize a barra de rolagem e escolha a imagem que melhor se aplica; como sugestão, procure por uma chamada **Water Light Pattern**, e, ao encontrá-la, arraste-a sobre a superfície do cubo.

Figura 36 – Cubos



Fonte: O autor, 2023.

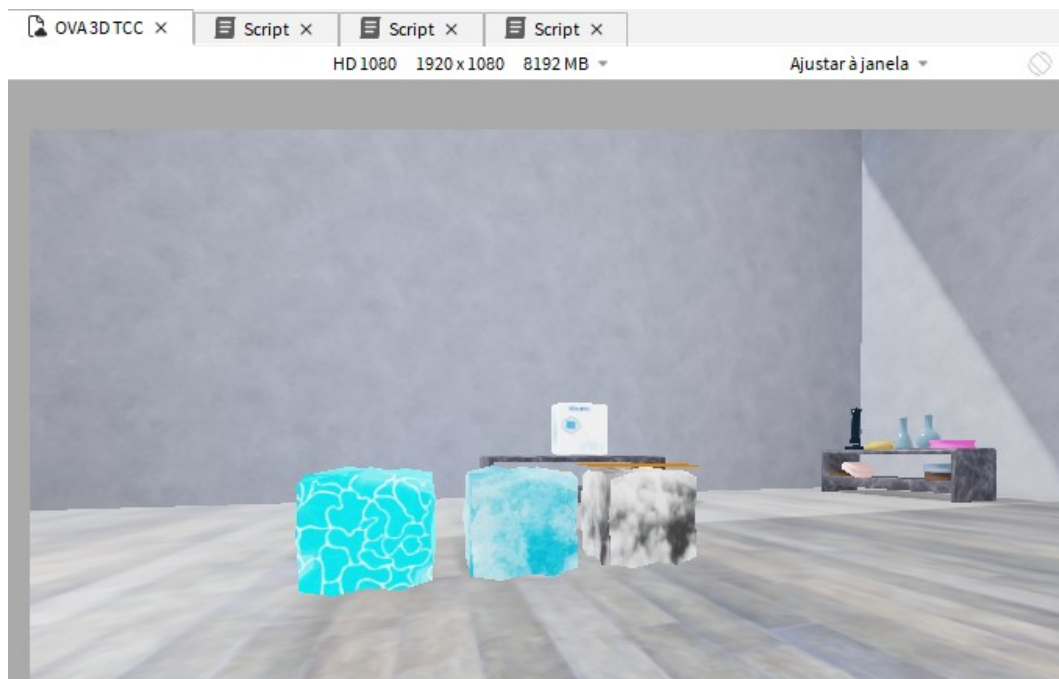
Figura 37 – Imagens na Caixa de Ferramentas



Fonte: O autor, 2023.

Posicione imagens em todas as faces do cubo, de forma que ele pareça ser inteiramente desse material, assim como demonstra a Figura 38. Faça isso, para todos os cubos.

Figura 38 – Cubos com Texturas

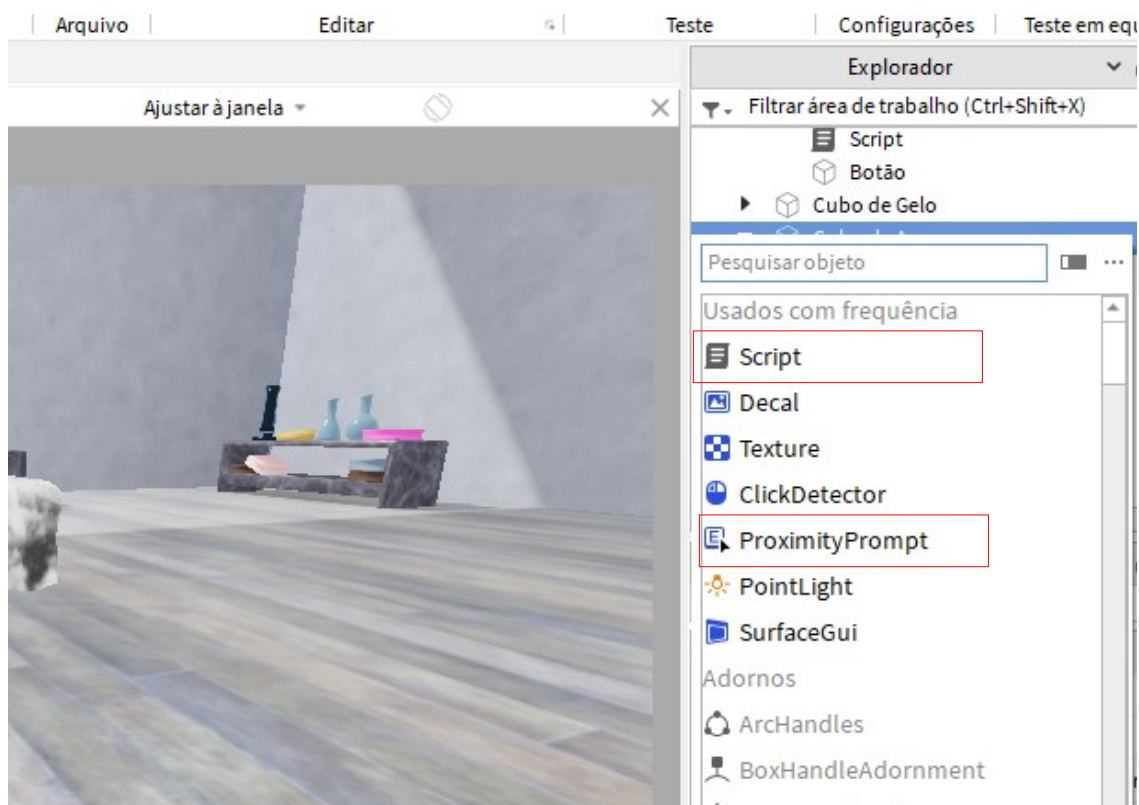


Fonte: O autor, 2023.

Nesse ponto, os cubos estão visualmente prontos. Basta, agora, que o comportamento deles seja definido, e isso será feito por meio de *scripts* e componentes auxiliares. Identifique, na hierarquia, os cubos criados. Usaremos como exemplo o cubo de água. Ao encontrá-lo, posicione o mouse sobre ele e acrescente um objeto chamado **ProximityPrompt** clicando no + que aparece do lado direito do objeto; acrescente, também, o objeto *script*. A Figura 38 evidencia esses objetos.

O **ProximityPrompt** trata-se de um objeto que permite identificar a aproximação do jogador com relação ao objeto que o possui como filho. Sendo assim, permite uma iteração, via teclado, de jogadores que estão a uma certa distância dele.

Figura 39 – Inserir Objetos Script e ProximityPrompt



Fonte: O autor, 2023.

Após inserir os objetos, dê dois cliques sobre o objeto **Script**. Isso fará com que o espaço para escrita de códigos em Lua apareça. Nessa aba, você encontrará um trecho **print** (“**Hello world!**”), que exibirá **Hello world!** no console do Roblox Studio. Substitua esse trecho pelo código a seguir.

As três primeiras linhas desse código dizem respeito à configuração das variáveis locais, as quais serão utilizadas para se referir aos objetos no contexto do **Script**. Pode-se dizer que, na primeira linha, temos a seguinte interpretação: a variável local **Prompt_proximidade** pode ser encontrada acessando-se o pai do **script**, ou seja, o objeto ao qual o **script** foi inserido. O mesmo se repete para as linhas seguintes, em que é atribuído como o pai da variável o pai do **script**, identificando-se a variável **plr** como o jogador local, que pode ser encontrado no contexto global do game, dentre os *players* disponíveis.

Em Lua, assim como em várias outras linguagens, o **.** indica o acesso a propriedades de um objeto ou classe. Sendo assim, ao utilizar **Prompt_proximidade.ActionText**, é o mesmo que dizer que você está alterando o texto de ação quando o jogador se aproxima de um objeto, ou seja, o texto que será exibido.

De maneira geral, o Script 1 faz com que seja mostrado **Pegar Líquido** quando o jogador se aproxima do cubo de água. Após se aproximar, o jogador terá a possibilidade de interagir com o objeto por meio da Tecla **E** que é definida por padrão no **ProximityPrompt**. Ao pressionar essa tecla por 1 segundo, ele equipará o item e fará com que o bloco desapareça.

Script 1 – Script para Coletar o Item Líquido

```

1. local Prompt_proximidade = script.Parent.ProximityPrompt
2. Prompt_proximidade.Parent = script.Parent
3. local plr = game.Players.LocalPlayer
4. --CONFIGURAÇÕES DO PROMPT
5. Prompt_proximidade.ActionText = "Pegar"
6. Prompt_proximidade.ObjectText = "Liquido" --Mude a sua preferencia o
   nome do item
7.
8. Prompt_proximidade.HoldDuration = 1 --Tempo de segurar a tecla! mude
   para sua preferencia
9. Prompt_proximidade.MaxActivationDistance = 20 --Quanto voce quer que
   a prompt apareca
10.
11.     Prompt_proximidade.Triggered:Connect(function(plr)
12.         local Item = game.GetService("ReplicatedStorage").Liquido
   --Mude aqui para o nome EXATO do item!
13.         local Clone = Item:Clone()
14.         Clone.Parent = plr.Character
15.         local pai = script.Parent["Meshes/Block 1"]
16.         pai.Transparency = 1
17.         script.Parent.Transparency = 1
18.
19.         for _, filho in pairs(pai:GetChildren()) do
20.             if filho:IsA("Decal") then
21.                 filho.Transparency = 1
22.             end
23.         end
24.         Prompt_proximidade.Enabled = false
25.     end)

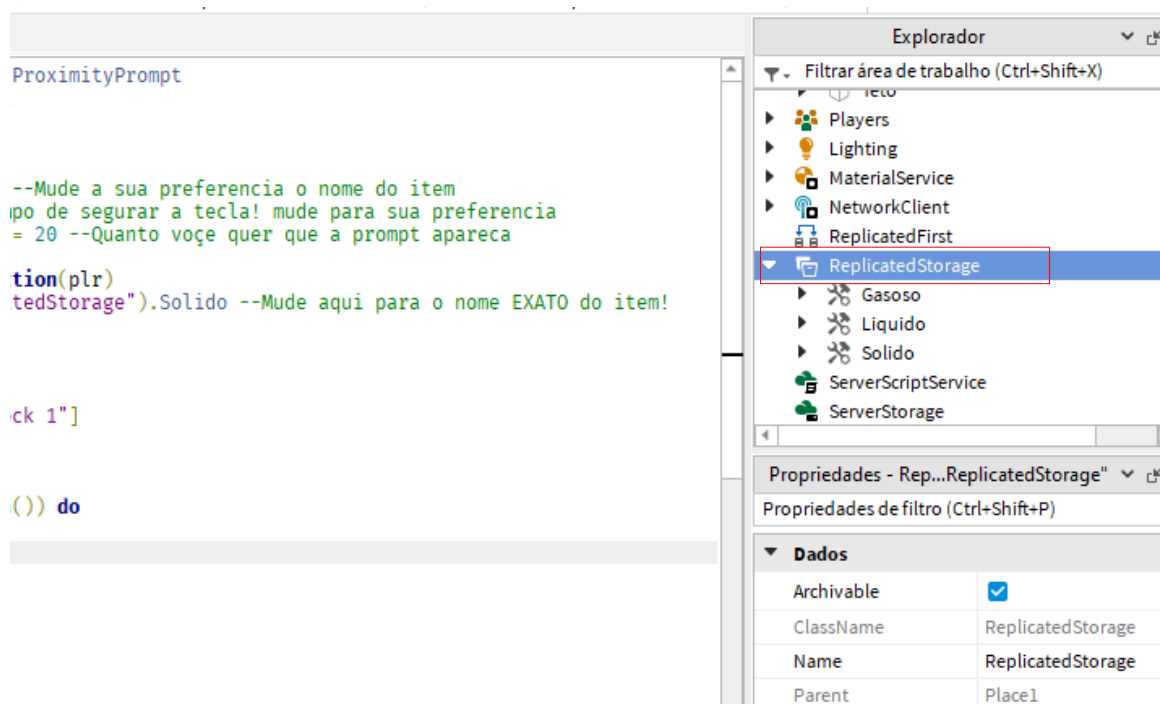
```

Fonte: O autor, 2023.

Para que este código funcione adequadamente, você deve adicionar, no **Replicated Storage**, na aba do **Explorador**, um objeto do tipo **Tool**, o qual pode conter uma **Mesh**, que trata da aparência do item. A Figura 40 revela onde se encontra o **Replicated Storage**. Para adicionar a **Tool**, basta seguir o mesmo procedimento previamente explicitado, em que se deve posicionar o mouse sobre o objeto e clicar em +; após isso, buscar por **Tool** na barra de

pesquisas.

Figura 40 – Localizar ReplicatedStorage



Fonte: O autor, 2023.

Acrescente um **Tool** para cada cubo e os renomeie como Líquido, Gasoso e Solido. Após efetuar esses passos, repita o processo para os outros dois cubos, acrescentando um **script** e um **ProximityPrompt** para cada e teste a experiência.

O próximo passo é criar um local para depositar os objetos que serão pegos pelos jogadores; nesse caso, um quadro com as transformações da matéria. Você tem liberdade para criar o plano de fundo da forma que preferir, uma vez que ele é meramente ilustrativo. O mais importante são os locais em que o jogador interagirá.

Você pode seguir o mesmo modelo utilizado para criar os cubos, onde três **parts** devem ser inseridas no contexto da experiência e, nelas, devem ser inseridos, por meio do "+", na aba do **Explorador**, um **script** e um **ProximityPrompt**. Elas podem ser posicionadas assim como demonstrado na Figura 41.

Figura 41 – Quadro de Transformações



Fonte: O autor, 2023.

Para o quadro de transformações, você deve adicionar em cada componente *script* o código do Script 2. Lembre-se de alterar os nomes dos componentes de acordo com os nomes que você definiu, pois assim evita que ocorram erros, garantindo que a experiência funcione de forma adequada.

Script 2 – Script para posicionar o Item Líquido

```

1. local Prompt_proximidade = Instance.new("ProximityPrompt")
2. Prompt_proximidade.Parent = script.Parent
3. local plr = game.Players.LocalPlayer
4. local obj = script.Parent
5. --CONFIGURAÇÕES DO PROMPT
6. Prompt_proximidade.ActionText = "Pegar"
7. Prompt_proximidade.ObjectText = "Item" --Mude a sua preferencia o nome do
   item
8. Prompt_proximidade.HoldDuration = 1 --Tempo de segurar a tecla! mude para
   sua preferencia
9. Prompt_proximidade.MaxActivationDistance = 20 --Quanto voce quer que a
   prompt apareca
10.
11. Prompt_proximidade.Triggered:Connect(function(plr)
12.     local itemEquipado = plr.Character:FindFirstChildWhichIsA("Tool")
13.     if(itemEquipado)then
14.         local clone = itemEquipado:Clone()
15.         clone.Parent = plr.Backpack
16.         itemEquipado:Remove()
17.         local itemMochila = plr.Backpack:FindFirstChildWhichIsA("Tool")
18.

```

```

19.         if(obj.Decal.Texture == "http://www.roblox.com/asset/?
id=15558862920") then
20.             if(itemMochila.Name=="Solido")then
21.                 obj.Decal.Texture = "rbxassetid://15177194321"
22.                plr.Backpack:ClearAllChildren()
23.             elseif(itemMochila.Name == "Liquido")then
24.                 obj.Decal.Texture = "rbxassetid://15177207526"
25.                plr.Backpack:ClearAllChildren()
26.             elseif(itemMochila.Name==" Gasoso")then
27.                 obj.Decal.Texture = "rbxassetid://15177208468"
28.                plr.Backpack:ClearAllChildren()
29.             end
30.         else
31.             local screenGui = Instance.new("ScreenGui")
32.             screenGui.Parent = plr:FindFirstChild("PlayerGui") or
plr:WaitForChild("Backpack")
33.             clone.Parent = plr.Character
34.             -- Crie um objeto "TextLabel" com a mensagem
35.             local mensagem = Instance.new("TextLabel")
36.             mensagem.Size = UDim2.new(0, 800, 0, 50) -- Tamanho da mensagem
37.             mensagem.Position = UDim2.new(0.5, -100, 0.5, -25) -- Posição
central
38.             mensagem.BackgroundColor3 = Color3.new(0, 0, 0) -- Cor de fundo
39.             mensagem.TextColor3 = Color3.new(1, 1, 1) -- Cor do texto
40.             mensagem.FontSize = Enum.FontSize.Size24
41.             mensagem.Text = "Já existe um Objeto aqui, posicione em outro
local" -- Texto da mensagem
42.             mensagem.Parent = screenGui
43.
44.             -- Remova a mensagem após um tempo (opcional)
45.             wait(5) -- Espera por 5 segundos
46.             mensagem:Remove() -- Remove a mensagem
47.         end
48.     end
49. end)

```

Fonte: O autor, 2023.

Agora, basta criar um botão para reiniciar a experiência. O design dele ficará por sua conta; um treinamento para reforçar os conceitos demonstrados neste tutorial. O Script 3 apresenta o código que define o comportamento do botão.

Script 3 – Script Reiniciar a experiência

```

1. local Prompt_proximidade = Instance.new("ProximityPrompt")
2. Prompt_proximidade.Parent = script.Parent
3. local plr = game.Players.LocalPlayer
4. --CONFIGURAÇÕES DO PROMPT

```

```

5. Prompt_proximidade.ActionText = "Pegar"
6. Prompt_proximidade.ObjectText = "Item" --Mude a sua preferencia o nome do
   item
7. Prompt_proximidade.HoldDuration = 1 --Tempo de segurar a tecla! mude para
   sua preferencia
8. Prompt_proximidade.MaxActivationDistance = 20 --Quanto você quer que a
   prompt apareca
9. Prompt_proximidade.Triggered:Connect(function(plr)

10.     local gasoso = script.Parent.Parent:FindFirstChild("SoltarGasoso")
11.     local liquido = script.Parent.Parent:FindFirstChild("SoltarLiquido")
12.     local solido = script.Parent.Parent:FindFirstChild("SoltarSolido")
13.     local pegarSolido =
       script.Parent.Parent:FindFirstChild("ObjetoPegarSolido")
14.     local pegarLiquido =
       script.Parent.Parent:FindFirstChild("ObjetoPegarLiquido")
15.     local pegarGasoso =
       script.Parent.Parent:FindFirstChild("ObjetoPegarGasoso")
16.
17.     if(gasoso.Decal.Texture ~= "http://www.roblox.com/asset/?
       id=15558862920" and liquido.Decal.Texture ~= "http://www.roblox.com/asset/?
       id=15558862920" and solido.Decal.Texture ~= "http://www.roblox.com/asset/?
       id=15558862920") then
18.
19.         if(gasoso.Decal.Texture == "rbxassetid://15177208468") then
20.             gasoso.Decal.Color3 = Color3.new(0, 1, 0)
21.         else
22.             gasoso.Decal.Color3 = Color3.new(1, 0, 0)
23.         end
24.
25.         if(liquido.Decal.Texture == "rbxassetid://15177207526") then
26.             liquido.Decal.Color3 = Color3.new(0, 1, 0)
27.         else
28.             liquido.Decal.Color3 = Color3.new(1, 0, 0)
29.         end
30.
31.         if(solido.Decal.Texture == "rbxassetid://15177194321") then
32.             solido.Decal.Color3 = Color3.new(0, 1, 0)
33.         else
34.             solido.Decal.Color3 = Color3.new(1, 0, 0)
35.         end
36.
37.         wait(3)
38.         pegarGasoso.ProximityPrompt.Enabled = true
39.
40.         pegarSolido.ProximityPrompt.Enabled = true
41.         pegarLiquido.ProximityPrompt.Enabled = true
42.

```

```

43.     local pais = {}
44.     table.insert(pais, pegarGasoso["Meshes/Block 1"])
45.     table.insert(pais, pegarSolido["Meshes/Block 1"])
46.     table.insert(pais, pegarLiquido["Meshes/Block 1"])
47.
48.
49.     script.Parent.Transparency = 0
50.     for _, pai in pairs(pais) do
51.         pai.Transparency = 0
52.         for _, filho in pairs(pai:GetChildren()) do
53.             if filho:IsA("Decal") then
54.                 filho.Transparency = 0
55.             end
56.         end
57.     end
58.
59.     gasoso.Decal.Texture = ""
60.     liquido.Decal.Texture = ""
61.     solido.Decal.Texture = ""
62.     solido.Decal.Color3 = Color3.new(0.898039, 0.898039, 0.898039)
63.     liquido.Decal.Color3= Color3.new(0.898039, 0.898039, 0.898039)
64.     gasoso.Decal.Color3= Color3.new(0.898039, 0.898039, 0.898039)
65. end
66. end)
67.

```

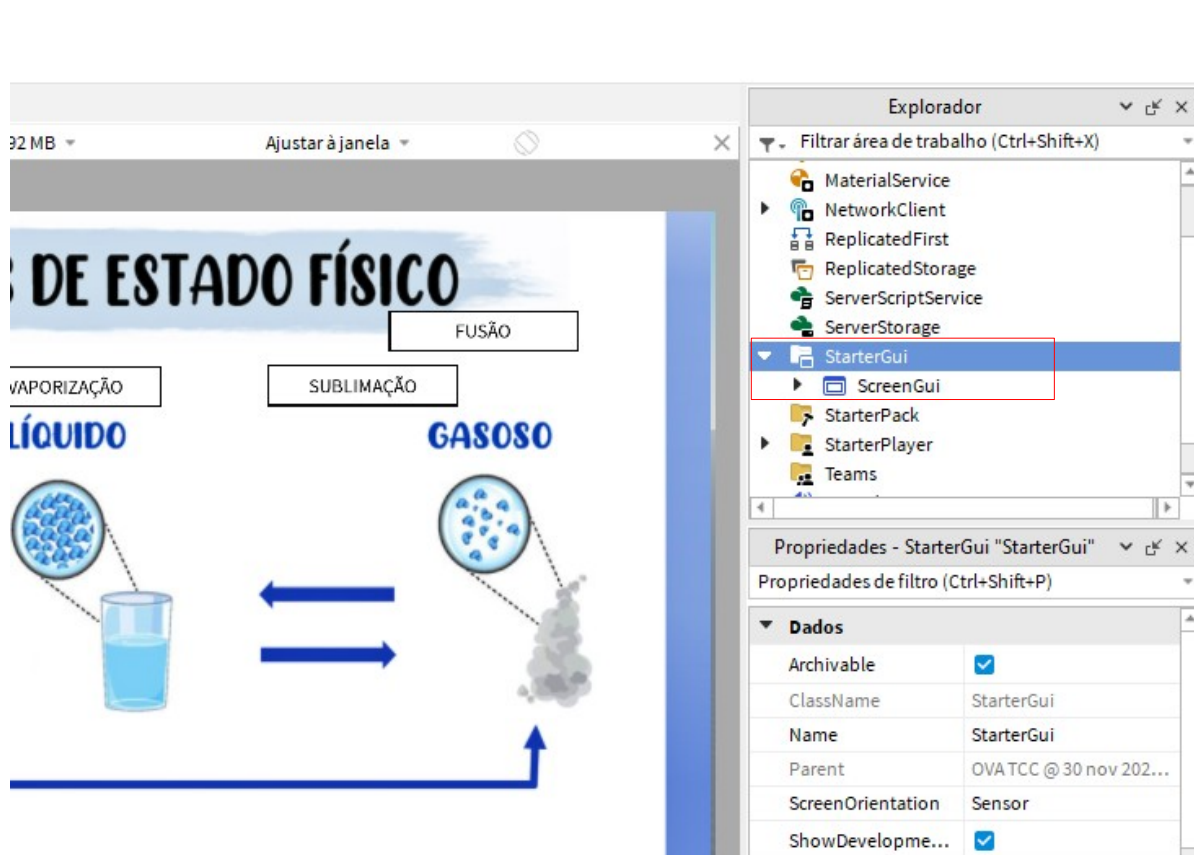
Fonte: O autor, 2023.

Agora, basta testar sua experiência e adaptá-la da forma mais adequada às suas necessidades. A experiência evidenciada aborda a maioria das funcionalidades utilizadas; sendo assim, você possui uma base e um exemplo para realizar suas próprias construções 3D.

6 Construção de uma Experiência 2D

Para construir uma experiência 2D, o primeiro passo é localizar, na aba **Explorador**, o campo **StarterGui**, responsável por exibir na tela os componentes da Gui Interface, ou seja, os componentes 2D do Roblox. Após encontrar esse componente, inclua uma **ScreenGui**, assim como demonstrado na Figura 42.

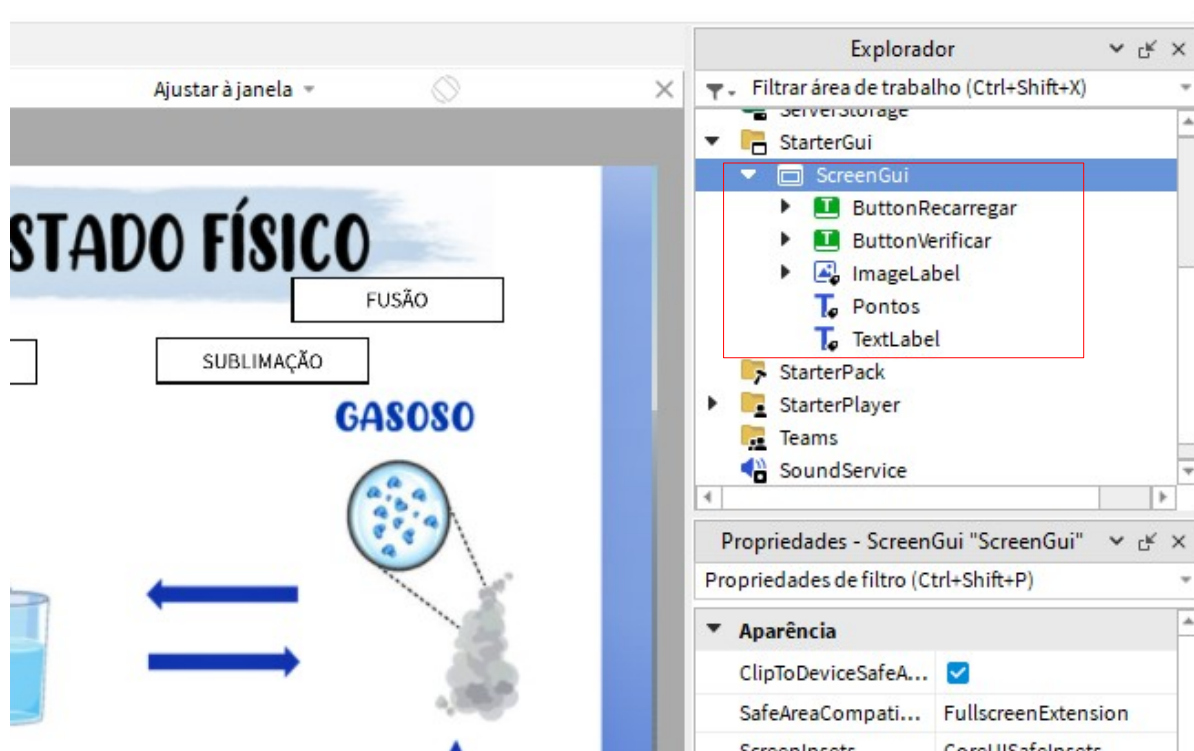
Figura 42 – Incluir ScreenGui



Fonte: O autor, 2023.

Após inserir o **ScreenGui**, adicione dentro dele dois **TextButton**, duas **TextLabel** e uma **ImageLabel**. Você pode renomeá-los como achar necessário, porém uma sugestão é seguir o exemplo da Figura 43. Deixe sua estrutura exatamente como a demonstrada, para que os **scripts** funcionem de forma adequada.

Figura 43 – Estrutura da Interface

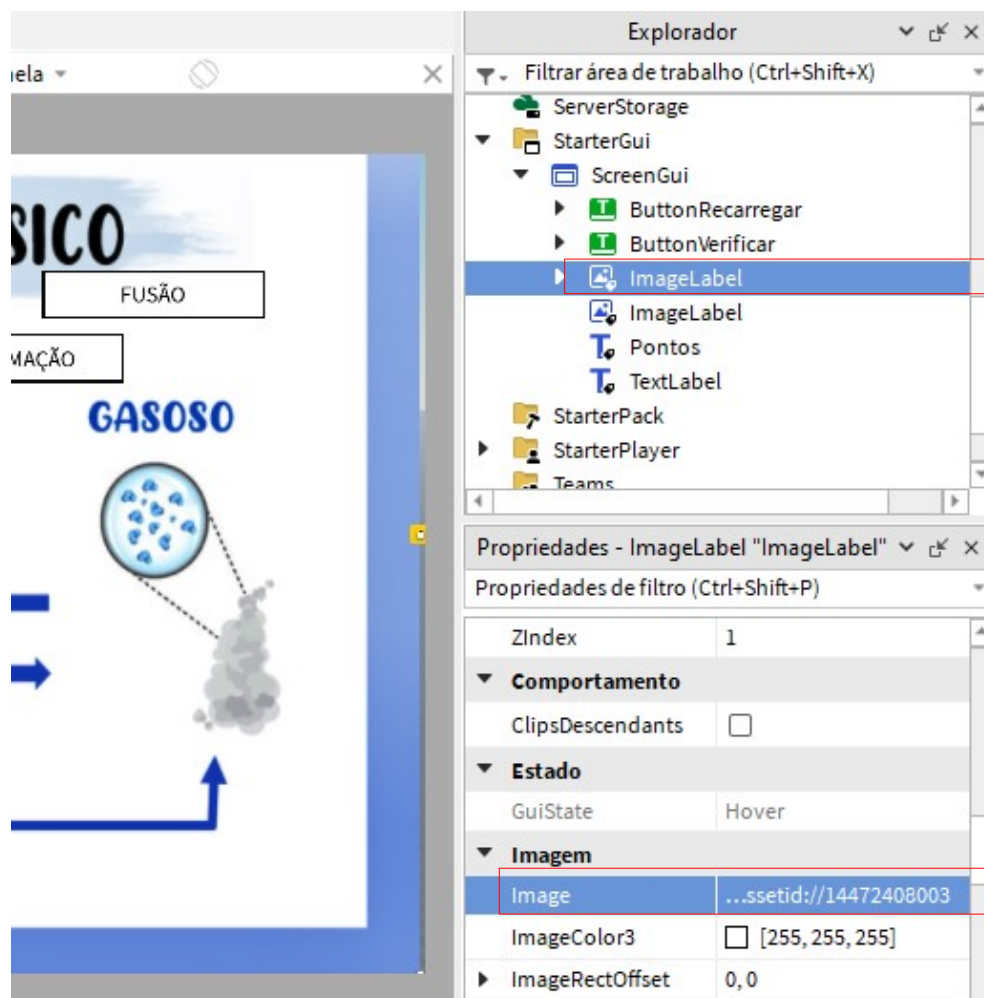


Fonte: O autor, 2023.

Após inserir todos os componentes citados anteriormente, localize sua **ImageLabel** e o campo **Imagem** desta, localizado na janela **propriedades**. Lembre-se de conferir se essa janela está habilitada, assim como exibido na Figura 21. Ao fazer isso, acesse o link [Images/ova h5p - Mercado de Criações \(roblox.com\)](https://www.roblox.com/asset/?id=14472408003). Esse link possui a imagem utilizada por padrão nas experiências desse tutorial, clique em **Tentar no Studio**, e ela será adicionada à sua coleção. Após isso, clique no campo **Imagem**, assim como demonstrado na Figura 44. Para acessar esse campo primeiro você deve clicar sobre a **ImageLabel** e depois encontrá-lo na janela **Propriedades**.

Após adicionar a imagem de plano de fundo na sua coleção, cole a seguinte URL: `rbxassetid://14472408003`, na janela exibida ao clicar na propriedade **Imagem**. Isso fará com que o plano de fundo do **ImageLabel** seja definido como o padrão utilizado nas experiências desse tutorial, assim como demonstrado na Figura 45. Após isso, redimensione-o para ocupar toda a tela do dispositivo.

Figura 44 – Estrutura da Interface



Fonte: O autor, 2023.

Certifique-se de que o plano de fundo foi inserido adequadamente, isso pode ser verificado visualmente, em que a imagem para plano de fundo substituirá a imagem padrão do componente **ImageLabel**.

Figura 45 – Plano de Fundo ImageLabel

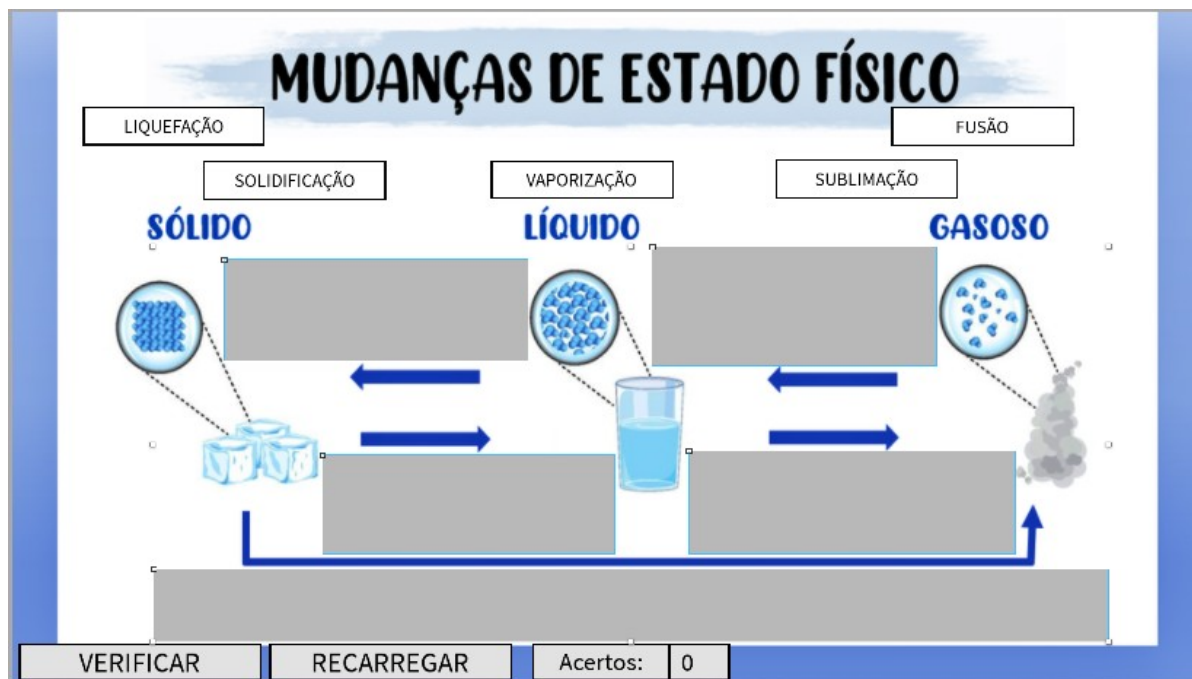


Fonte: O autor, 2023.

Após redimensionar a **ImageLabel** para ocupar toda a tela, faça com que o layout da experiência que está construída fique o mais próximo possível daquele demonstrado na Figura 46. Para isso, insira como um filho da **ImageLabel** cinco **TextLabel** e cinco **Frames**. Na figura, eles estão destacados na cor cinza; porém, por padrão, não terão essa cor, não sendo necessário que você a altere. Isso foi feito apenas para facilitar a visualização de onde serão posicionados.

Para alterar o texto das **Labels**, você deve clicar sobre elas na janela do **Explorador** e localizar, na janela de propriedades, o campo **Text**, e, assim, substituir o conteúdo pelo que você desejar.

Figura 46 – Layout da experiência 2D



Fonte: O autor, 2023.

Após isso, inclua em cada componente **Frame**, um objeto **BoolValue** e renomeie como **TemLabel**. Aproveite e inclua também um **Script** para cada **Frame**, um para a **ImageLabel** e também um para cada **TextButton**. Eles serão utilizados para controlar o comportamento da experiência.

Para o *script* de movimentação das **Labels**, aquele que está dentro da **ImageLabel**, você deve utilizar o código evidenciado no *Script 4* o qual faz com que seja possível movimentar uma quantidade variável de **Labels**. Desse modo, ele é generalista e faz com que a posição do objeto se altere de acordo com o arraste do mouse.

Script 4 – Script para movimentação das Label's

```

1. -- Referências aos objetos
2. local screenGui = script.Parent -- A ScreenGui que contém as TextLabels
3. local mouse = game.Players.LocalPlayer:GetMouse() -- Objeto mouse
4. local sublimacao = script.Parent.Sublimacao
5. -- Tabela para armazenar informações de arrasto para cada TextLabel
6. local draggingInfo = {}
7. local fusao = script.Parent.Fusao
8. local solidificacao = script.Parent.Solidificacao
9. local liquefacao = script.Parent.Liquefacao
10. local vaporizacao = script.Parent.Vaporizacao

```

```

11.

12. -- Obtém a resolução da tela do jogador
13. local player = game.Players.LocalPlayer
14. local screenWidth = player.PlayerGui.ScreenGui.AbsoluteSize.X
15. local screenHeight = player.PlayerGui.ScreenGui.AbsoluteSize.Y
16.
17. -- Configura o tamanho da ImageLabel para cobrir toda a tela
18. local imageLabel = script.Parent -- Assumindo que o script está dentro da
    ImageLabel
19. imageLabel.Size = UDim2.new(0, screenWidth, 0, screenHeight)
20.
21. -- Funções para manipulação do arrasto
22. local function onMouseButton1Down(label)
23.     local offset = label.Position - UDim2.new(0, mouse.X, 0, mouse.Y)
24.     draggingInfo[label] = {
25.         isDragging = true,
26.         offset = offset
27.     }
28. end
29.
30. local function onMouseButton1Up(label)
31.     draggingInfo[label] = nil
32. end
33.
34. local function onMouseMove()
35.     for label, info in pairs(draggingInfo) do
36.         if info.isDragging then
37.             local newPosition = UDim2.new(0, mouse.X, 0, mouse.Y) +
                info.offset
38.             label.Position = newPosition
39.             sublimacao.BackgroundColor3 = Color3.new(0.827451, 0.827451,
                0.827451)
40.             fusao.BackgroundColor3 = Color3.new(0.827451, 0.827451,
                0.827451)
41.             solidificacao.BackgroundColor3 = Color3.new(0.827451, 0.827451,
                0.827451)
42.             liquefacao.BackgroundColor3 = Color3.new(0.827451, 0.827451,
                0.827451)
43.             vaporizacao.BackgroundColor3 = Color3.new(0.827451, 0.827451,
                0.827451)
44.         end
45.     end
46. end
47. -- Conectar eventos para cada TextLabel
48. for _, label in ipairs(screenGui:GetDescendants()) do
49.     if label:IsA("TextLabel") then
50.         label.InputBegan:Connect(function(input)

```

```

51.     if input.UserInputType == Enum.UserInputType.MouseButton1 then
52.         onMouseButton1Down(label)
53.
54.     end
55. end)
56.
57. label.InputEnded:Connect(function(input)
58.     if input.UserInputType == Enum.UserInputType.MouseButton1 then
59.         onMouseButton1Up(label)
60.         sublimacao.BackgroundColor3 = Color3.new(1, 1, 1)
61.         fusao.BackgroundColor3 = Color3.new(1, 1, 1)
62.         solidificacao.BackgroundColor3 = Color3.new(1, 1, 1)
63.         liquefacao.BackgroundColor3 = Color3.new(1, 1, 1)
64.         vaporizacao.BackgroundColor3 = Color3.new(1, 1, 1)
65.     end
66. end)
67. end
68. end
69.
70. mouse.Move:Connect(onMouseMove)
71.

```

Fonte: O autor, 2023.

O *script* utilizado nos **Frames** serve para criar uma espécie de imã ao qual a **Label** cola na posição definida ao se aproximar da área do frame. Ela é determinada manualmente; sendo assim, cabe a você defini-la da forma que melhor lhe atender.

Atente-se para o comando `label.Position = UDim2.new(0.63, 0, 0.73, 0)`, que é responsável por definir uma coordenada para a **Label**. Assim, você deve encontrar a melhor posição de seu objeto de acordo com a propriedade **Position** da **Label** que você selecionar. Dessa forma, acrescente o conteúdo do *Script* 5 para cada um dos **Frame's**, lembre-se de adequar a posição de acordo com cada um deles.

Script 5 – Script de verificação de Label Correta

```

1. local meuFrame = script.Parent
2. local TemLabel = meuFrame.TemLabel
3. local labels = {} -- Crie uma tabela para armazenar todas as labels que
   você deseja controlar
4. -- Adicione todas as labels que você deseja controlar à tabela "labels"
5. table.insert(labels, meuFrame.Parent:FindFirstChild("LabelFusao"))
6. table.insert(labels, meuFrame.Parent:FindFirstChild("LabelSolidificacao"))
7. table.insert(labels, meuFrame.Parent:FindFirstChild("LabelLiquefacao"))
8. table.insert(labels, meuFrame.Parent:FindFirstChild("LabelSublimacao"))
9. table.insert(labels, meuFrame.Parent:FindFirstChild("LabelVaporizacao"))
10. -- Função para ajustar a posição de uma label específica

```

```

11. local function ajustarPosicao(label)
12.   label.Position = UDim2.new(0.63, 0,0.73, 0)
13. end
14. -- Função para verificar se a label está dentro do frame
15. local function verificarLabelNoFrame(label)
16.   local labelScreenPosition = label.AbsolutePosition
17.   local labelSize = label.AbsoluteSize
18.   local frameScreenPosition = meuFrame.AbsolutePosition
19.   local frameSize = meuFrame.AbsoluteSize
20.   if (labelScreenPosition.X >= frameScreenPosition.X and
21.     labelScreenPosition.X + labelSize.X <= frameScreenPosition.X +
     frameSize.X and
22.     labelScreenPosition.Y >= frameScreenPosition.Y and
23.     labelScreenPosition.Y + labelSize.Y <= frameScreenPosition.Y +
     frameSize.Y) then
24.     -- A label está dentro do frame
25.     if(TemLabel.Value == false)then
26.       ajustarPosicao(label)
27.       TemLabel.Value = true
28.     else if (TemLabel.Value== true) then
29.       if(label.Position ~= UDim2.new(0.63, 0,0.73, 0))then
30.         if(TemLabel.Value==true and label.Name == 'LabelFusao')then
31.           label.Position = UDim2.new(0.743, 0,0.2, 0)
32.         else if(TemLabel.Value==true and label.Name ==
'LabelSolidificacao')then
33.           label.Position = UDim2.new(0.166, 0,0.274, 0)
34.         else if(TemLabel.Value==true and label.Name ==
'LabelLiquefacao')then
35.           label.Position = UDim2.new(0.065, 0,0.199, 0)
36.         else if(TemLabel.Value==true and label.Name ==
'LabelSublimacao')then
37.           label.Position = UDim2.new(0.645, 0, 0.273, 0)
38.         else if(TemLabel.Value==true and label.Name ==
'LabelVaporizacao')then
39.           label.Position = UDim2.new(0.407, 0,0.274, 0)
40.           end
41.         end
42.       end
43.     end
44.   end
45.   TemLabel.Value=false
46. end
47. end
48. end
49. end
50. end
51. -- Conecte a função verificarLabelNoFrame ao evento MouseMove
52. game:GetService("UserInputService").InputChanged:Connect(function(input,

```

```

gameProcessedEvent)
53. if input.UserInputType == Enum.UserInputType.MouseMovement then
54.   for _, label in pairs(labels) do
55.     verificarLabelNoFrame(label)
56.   end
57. end
58. end)
59.

```

Fonte: O autor, 2023.

Por fim, tem-se os *script's* dos botões de verificação e de recarregamento da experiência, os quais fazem uso de posições fixas das **Label's**. **Dessa forma, utilize a mesma** dos *scripts* anteriores, e, para o botão de verificação, o código do *Script 6*.

Script 6 – Script para o botão de verificação

```

1. -- Conectar o evento MouseButton1Click do botão a uma função
2. local button = script.Parent
3. local solidificacao =
   script.Parent.Parent.ImageLabel:FindFirstChild("LabelSolidificacao")
4. local fusao = script.Parent.Parent.ImageLabel:FindFirstChild("LabelFusao")
5. local liquefacao =
   script.Parent.Parent.ImageLabel:FindFirstChild("LabelLiquefacao")
6. local vaporizacao =
   script.Parent.Parent.ImageLabel:FindFirstChild("LabelVaporizacao")
7. local sublimacao =
   script.Parent.Parent.ImageLabel:FindFirstChild("LabelSublimacao")
8. local pontos = 0
9. local pontuacao = script.Parent.Parent.Pontos
10. local function OnButtonClicked()
11.   pontos = 0
12.   if(solidificacao.Position == UDim2.new(0.232, 0,0.451, 0))then
13.     solidificacao.BackgroundColor3 = Color3.new(0, 1, 0)
14.     pontos = pontos+1
15.   else
16.     solidificacao.BackgroundColor3 = Color3.new(1, 0, 0)
17.   end
18.   if(fusao.Position == UDim2.new(0.311, 0,0.730, 0))then
19.     fusao.BackgroundColor3 = Color3.new(0, 1, 0)
20.     pontos = pontos+1
21.   else
22.     fusao.BackgroundColor3 = Color3.new(1, 0, 0)
23.   end
24.   if(liquefacao.Position == UDim2.new(0.59, 0,0.46, 0))then
25.     liquefacao.BackgroundColor3 = Color3.new(0, 1, 0)
26.     pontos = pontos+1
27.   else

```

```

28. liquefacao.BackgroundColor3 = Color3.new(1, 0, 0)
29. end
30. if(vaporizacao.Position == UDim2.new(0.63, 0,0.73, 0))then
31.   vaporizacao.BackgroundColor3 = Color3.new(0, 1, 0)
32.   pontos = pontos+1
33. else
34.   vaporizacao.BackgroundColor3 = Color3.new(1, 0, 0)
35. end
36. if(sublimacao.Position == UDim2.new(0.431, 0, 0.857, 0))then
37.   sublimacao.BackgroundColor3 = Color3.new(0, 1, 0)
38.   pontos = pontos+1
39. else
40.   sublimacao.BackgroundColor3 = Color3.new(1, 0, 0)
41. end
42. pontuacao.Text = pontos
43. end
44. button.MouseButton1Click:Connect(OnButtonClicked)
45.

```

Fonte: O autor, 2023.

Já para o botão de recarregamento da experiência utilize o código do Script 7:

Script 7 – Script para o botão de recarregamento

```

1. -- Conectar o evento MouseButton1Click do botão a uma função
2. local button = script.Parent
3. local solidificacao =
   script.Parent.Parent.ImageLabel:FindFirstChild("LabelSolidificacao")
4. local fusao = script.Parent.Parent.ImageLabel:FindFirstChild("LabelFusao")
5. local liquefacao =
   script.Parent.Parent.ImageLabel:FindFirstChild("LabelLiquefacao")
6. local vaporizacao =
   script.Parent.Parent.ImageLabel:FindFirstChild("LabelVaporizacao")
7. local sublimacao =
   script.Parent.Parent.ImageLabel:FindFirstChild("LabelSublimacao")
8. local pontuacao = script.Parent.Parent.Pontos
9.
10. local function OnButtonClicked()
11.   solidificacao.Position = UDim2.new(0.166, 0,0.274, 0)
12.   solidificacao.BackgroundColor3 = Color3.new(1, 1, 1)
13.
14.   fusao.Position = UDim2.new(0.743, 0,0.2, 0)
15.   fusao.BackgroundColor3 = Color3.new(1, 1, 1)
16.
17.   liquefacao.Position = UDim2.new(0.065, 0,0.199, 0)
18.   liquefacao.BackgroundColor3 = Color3.new(1, 1, 1)
19.
20.
21.   vaporizacao.Position = UDim2.new(0.407, 0,0.274, 0)

```

```
22. vaporizacao.BackgroundColor3 = Color3.new(1, 1, 1)
23.
24. sublimacao.Position = UDim2.new(0.645, 0, 0.273, 0)
25. sublimacao.BackgroundColor3 = Color3.new(1, 1, 1)
26.
27. pontuacao.Text = 0
28. end
29.
30. button.MouseButton1Click:Connect(OnButtonClicked)
31.
```

Fonte: O autor, 2023.

Ao construir essas duas experiências, você possuirá o conhecimento necessário para desenvolver tanto no contexto 2D quanto no 3D. Assim, poderá utilizar os OVAs construídos no contexto que melhor lhe atender, bastando utilizar sua imaginação e explorar novos conceitos.

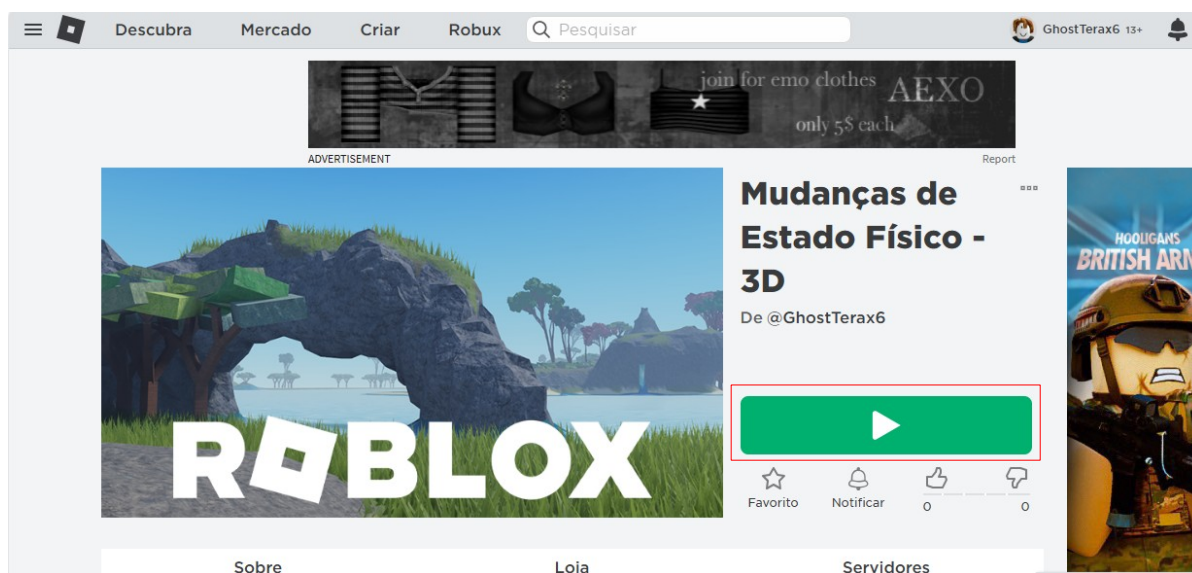
7 Experiências Disponibilizadas

Os Objetos Virtuais de Aprendizagem discutidos neste tutorial estão acessíveis para aplicação e análise e podem ser encontrados nos seguintes endereços:

- Experiência 2D: [Mudanças de Estado Físico - 2D - Roblox](#)
- Experiência 3D: [Mudanças de Estado Físico - 3D - Roblox](#)

Para utilizá-los como experiências, você deve acessar o *link* de um deles e clicar em um botão semelhante aos utilizados comumente para dar *play* em músicas, vídeos e diversas outras aplicações. Este botão está destacado por um retângulo vermelho, na Figura 47.

Figura 47 – Acessar a Experiência

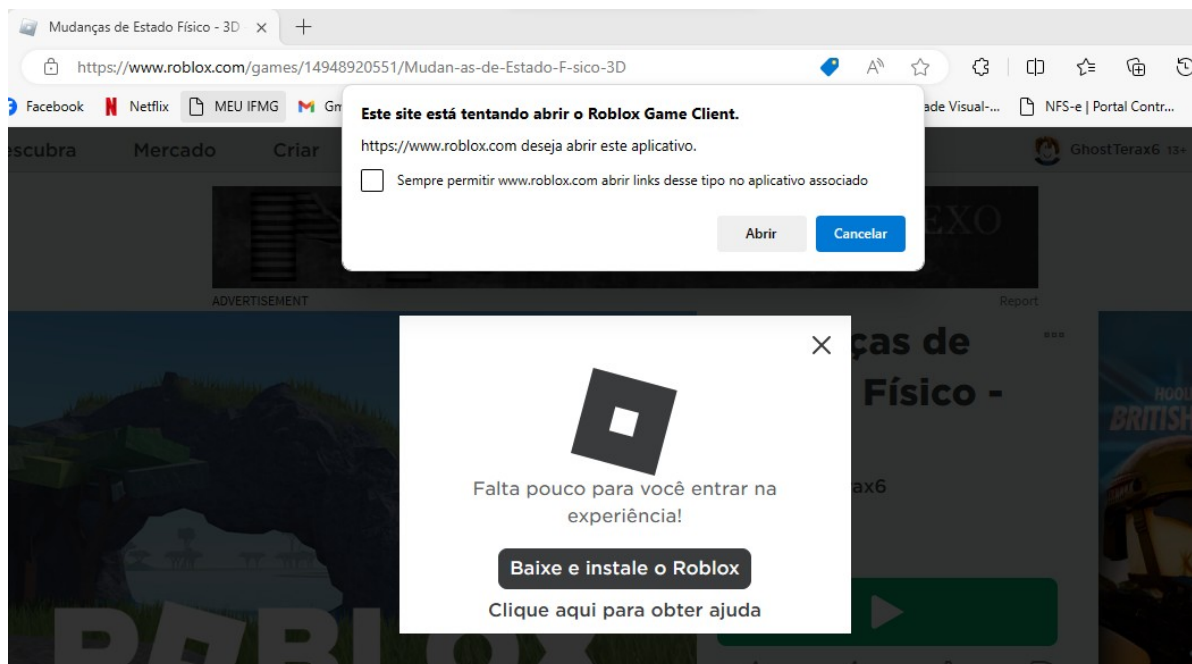


Fonte: O autor, 2023.

Após clicar no botão, uma caixa de diálogo será exibida solicitando a abertura da experiência no Roblox Player, conforme demonstrado na Figura 48. Lembre-se de baixar essa ferramenta antes de seguir esses passos. Clique em **abrir** para desfrutar da experiência.

Ao seguir esses passos, o OVA iniciará em seu ambiente, permitindo que você interaja e siga o fluxo de atividades propostas por ele, conforme evidenciado na Figura 49. Esses passos são os mesmos para os dois OVAs; portanto, você pode realizá-los para ambos.

Figura 48 – Abrir Roblox Player



Fonte: O autor, 2023.

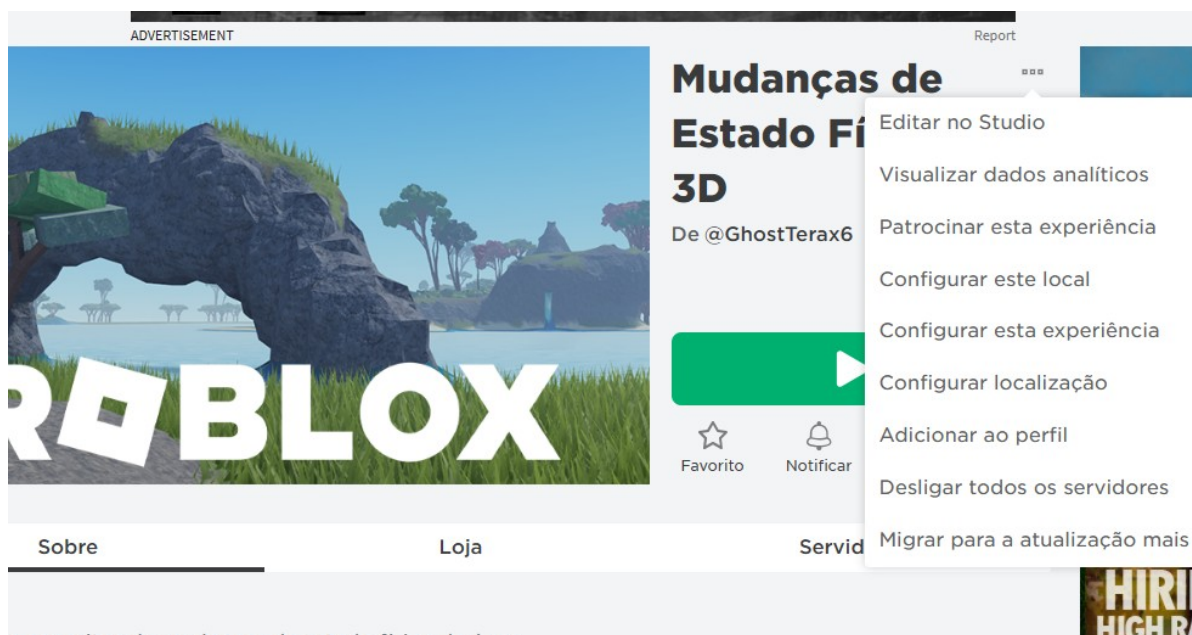
Figura 49 – Interagir com o OVA 3D



Fonte: O autor, 2023.

Outra forma de utilizar as experiências é por meio da sua modificação pelo Roblox Studio, o que pode ser feito acessando qualquer um dos links acima listados. Porém, em vez de clicar no ícone de play, você clicará nos três pontos no canto superior direito da experiência e, em seguida, em **Editar no Studio**, conforme demonstrado na Figura 50.

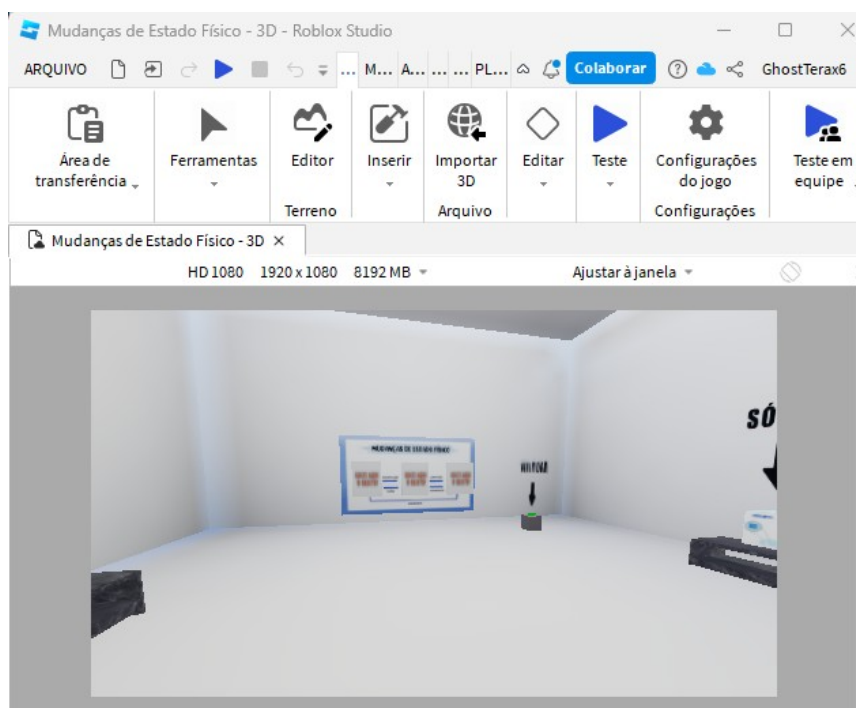
Figura 50 – Editar OVA 3D



Fonte: O autor, 2023.

Você realizará um passo semelhante ao demonstrado na Figura 48. Após clicar em **abrir**, você se deparará com um cenário semelhante ao demonstrado pela Figura 51. Neste ambiente, você terá a liberdade de modificar a experiência construída previamente neste tutorial, ajustando-a conforme necessário. Recomenda-se que você acompanhe os passos de criação neste tutorial enquanto a experiência estiver aberta em seu ambiente de desenvolvimento, para uma compreensão mais clara de onde cada parte da construção está localizada.

Figura 51 – Editar OVA 3D no Roblox Studio



Fonte: O autor, 2023.

Ao concluir este tutorial, espera-se que você tenha adquirido uma compreensão abrangente sobre como utilizar e modificar os Objetos Virtuais de Aprendizagem (OVAs) no Roblox. Ao seguir os passos descritos, você foi capaz de acessar as experiências, interagir com elas e até mesmo modificá-las no Roblox Studio. Por meio deste processo, você pôde explorar as possibilidades educacionais e criativas oferecidas por essa plataforma. Lembre-se de que a prática e a experimentação são fundamentais para aprofundar seu conhecimento e habilidades neste contexto. Continue explorando e criando, e aproveite ao máximo as oportunidades que o Roblox proporciona para aprimorar a aprendizagem e a criatividade.

8 Onde aprender a programar

Durante a construção das experiências mencionadas anteriormente, é possível que você tenha enfrentado algumas dificuldades com conceitos de programação. Para ajudá-lo a aprofundar seu conhecimento nessa área, abaixo, estão algumas plataformas que podem ser úteis em sua busca por aprendizado.

8.1 +IFMG

O IFMG disponibiliza uma plataforma com uma variedade de cursos gratuitos, incluindo alguns relacionados à programação. Embora não haja cursos específicos sobre a linguagem Lua, utilizada no Roblox, você pode explorar conceitos básicos e aprimorar sua compreensão sobre lógica de programação. Os cursos disponíveis podem ser encontrados no seguinte link:

- <https://mais.ifmg.edu.br/maisifmg/course/index.php?categoryid=10>

8.2 Medium

Na plataforma Medium, há um artigo escrito por Fly_san, um jogador conhecido na comunidade da plataforma que destaca pontos importantes para quem deseja iniciar na programação do Roblox. O artigo aborda conceitos básicos, como a utilização das funções *output* e *print()*, declaração e uso de variáveis dentro de funções, manipulação de objetos e suas propriedades, condicionais, *loops* e *tables*. Você pode acessar o artigo através do seguinte link:

- <https://medium.com/roblox-developer-portugu%C3%AAs/aprenda-a-programar-na-roblox-em-lua-com-o-desenvolvedor-fly-san-518ce9d6aa6c>

8.3 YouTube

No YouTube, você encontrará uma ampla variedade de vídeos abordando conceitos de programação no Roblox. Se preferir essa abordagem para aprender a programar, é recomendado buscar por vídeos que cubram conceitos básicos de programação. Para começar, sugiro que você inicie seus estudos pelo canal ShadowHxD, que possui uma playlist demonstrando a construção de uma experiência no Roblox. Você pode acessar essa playlist através do seguinte link:

- <https://www.youtube.com/playlist?list=PLsgfDS2ns-byZepJXgOksfNhZmk1GQiGn>