

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SÃO  
PAULO - CAMPUS SÃO PAULO**

**CURSO TÉCNICO DE DESENVOLVIMENTO DE SISTEMAS INTEGRADO AO  
ENSINO MÉDIO**

LORENZO TEIXEIRA

LUCAS RODRIGUES GARCIA

MIGUEL FERNANDES CAVALCANTI DA SILVA

NICHOLAS MEDEIROS MANSANO ALCANTARA PINHEIRO

NICOLLAS AUGUSTO RODRIGUES OLIVEIRA

THALES DE BRITO ENDO

**AEQUILIBRIUM**

SÃO PAULO

2025

LORENZO TEIXEIRA

LUCAS RODRIGUES GARCIA

MIGUEL FERNANDES CAVALCANTI DA SILVA

NICHOLAS MEDEIROS MANSANO ALCANTARA PINHEIRO

NICOLLAS AUGUSTO RODRIGUES OLIVEIRA

THALES DE BRITO ENDO

## **AEQUILIBRIUM**

Documentação de Projeto Interdisciplinar  
apresentado como requisito parcial para  
avaliação nas disciplinas de Arquitetura de  
Computadores e Rede, Desenvolvimento  
Web, Informática e Ferramentas para  
Desenvolvimento e Lógica de Programação

Orientadores(as): Ana Lucia Grici Zacarin  
Mamede, André Evandro Lourenço,  
Claudete de Oliveira Alves, Claudia Miyuki  
Werhmuller, Osmar Antunes Junior, João  
Antonio Temochko Andre, Patricia Andrea  
Paladino

SÃO PAULO

2025

## SUMÁRIO

<b>1 INTRODUÇÃO.....</b>	<b>4</b>
<b>2 PLANEJAMENTO DO PROJETO.....</b>	<b>5</b>
2.1 PESQUISAS E INSPIRAÇÕES.....	5
2.2 ESCOPO DO JOGO.....	5
2.3 ATAS E REUNIÕES.....	8
<b>3 DESENVOLVIMENTO.....</b>	<b>10</b>
3.1. DESIGN E ARTE.....	10
3.2 ARQUITETURA E CÓDIGO.....	14
3.3 O SITE DE DIVULGAÇÃO.....	14
3.3.1 DEMONSTRAÇÃO DO WEBSITE.....	14
3.3.2 TECNOLOGIAS USADAS.....	17
3.3.3 COMENTÁRIOS SOBRE O WEBSITE.....	18
<b>4 PROTÓTIPOS E TESTES.....</b>	<b>19</b>
<b>5 CONCLUSÃO.....</b>	<b>20</b>
<b>REFERÊNCIAS.....</b>	<b>21</b>

## 1 INTRODUÇÃO

Foi proposto aos alunos de Desenvolvimento de Sistemas, como um projeto final, que fizessem um jogo. Um jogo que os grupos poderiam escolher o gênero, estilo de arte, engine em que o jogo seria feito e mais, um projeto sem muitas restrições, porém ainda tendo algumas, como jogos sem violência, sem temas políticos, sem mecânicas de aposta e outras coisas que entrem nesse gênero.

Nosso grupo decidiu criar o jogo Aequilibrium, um jogo básico no estilo Metroidvania (um gênero de jogo que mistura plataforma 2D com um mundo aberto para o jogador explorar), na engine de jogos GameMaker Studio 2. Nesse jogo, o protagonista pode trocar entre duas formas, baseadas no Yin Yang. Cada forma tem mecânicas e jogabilidade relativamente diferentes, e o jogador tem que trocar entre elas quando necessário.

O nosso público-alvo são pessoas que jogam regularmente e já tem uma pequena experiência com plataformas e/ou Metroidvanias. Escolhemos esse gênero de jogo pois a maioria do grupo já jogou e gosta desse tipo de gênero, também por não ser muito complexo de fazer, além de ser um estilo de jogo que pode ser expandido facilmente e sua flexibilidade para novas mecânicas.

Esse documento vai apresentar para o leitor como foi o processo de fazer o jogo, contendo o desenvolvimento, o processo artístico, criativo e também uma das dificuldades que tivemos para a criação do jogo Aequilibrium como iniciantes na programação.

## 2 PLANEJAMENTO DO PROJETO

O projeto foi subdividido em múltiplas etapas, com diversas ideias e inspirações seguindo ao longo do tempo. Separaremos o planejamento em três etapas, sendo elas: Pesquisas, Escopo e Atas

### 2.1 PESQUISAS E INSPIRAÇÕES

Nossas maiores inspirações para a ideia do jogo foram os jogos independentes Hollow Knight, Dead Cells e Nine Sols, que são jogos do mesmo estilo, Celeste, um jogo plataforma com movimentação simples porém dinâmica, e Pizza Tower, um jogo plataforma frenético e rápido com várias transformações

Vimos muitos tutoriais quando tínhamos dificuldades em implementar certas mecânicas queríamos colocar, vídeos oficiais da GameMaker e sua documentação, vídeos da Sara Spalding, do Peyton Burnham, 1upIndie. Também vimos alguns vídeos do Inbound Shovel, desenvolvedor do futuro jogo independente Isadora's Edge.

### 2.2 ESCOPO DO JOGO

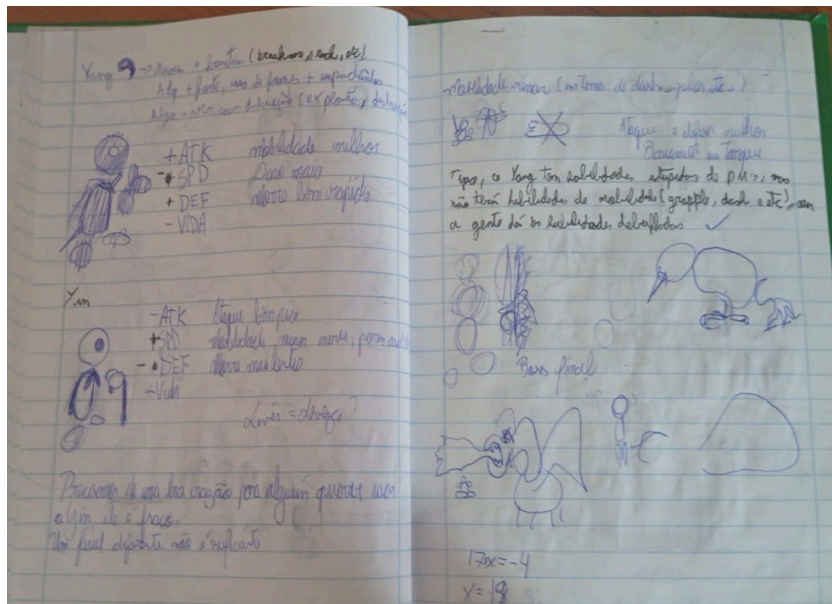
No nosso jogo, o jogador tem a opção de trocar entre o Yin e Yang, onde cada um terá uma jogabilidade diferente, ao mesmo tempo que terão mecânicas em comum.

O Yin é a parte mais calma do personagem, tendo uma jogabilidade mais lenta, porém ainda divertida, contendo duas mecânicas principais: a corrida, para o personagem poder andar mais rápido, e o parry, uma mecânica que faz o personagem principal rebater um projétil que está vindo em sua direção, podendo maximizar o dano em inimigos.

Já o Yang é a parte mais caótica do personagem, então ele tem um foco maior na mobilidade, tendo, também, duas mecânicas principais: o pulo duplo permite que o jogador pule duas vezes no ar, e o dash, uma mecânica que faz com

que o protagonista dê uma investida para a esquerda ou direita, abrindo possibilidades novas no mapa. (Figura 1)

Figura 1 - Esboço das ideias e personagens



Disponível em: Fonte do autor.

Nosso jogo tem 3 tipos de inimigos diferentes, sendo eles: O “guarda”, um inimigo simples que anda por áreas do mapa, e quando o jogador entra em contato com ele toma dano; O perseguidor, um inimigo que quando o jogador entra no alcance dele, começa a persegui-lo até atingir ele ou o personagem sair de seu alcance; E por último o atirador — ele simplesmente fica parado atirando projéteis, para esse inimigo em específico, o ideal é que ele seja eliminado com sua própria arma, ou seja, utilizando o parry.

No final do jogo tem um chefe simples, um “guardião” gigante que usa suas mão para fazer 3 ataques diferentes escolhidos aleatoriamente: o primeiro ele desliza sua mão até metade da área, esse ataque por ser pequeno tem o dobro de chance de ser escolhido; o segundo ataque, ele atira 8 projéteis em formato de círculo; e no terceiro, ele bate suas mãos na chão, fazendo cair pedras de cima. (Figura 2)

Figura 2 - Esboço do chefe



Disponível em: Fonte do autor.

Escolhemos o GameMaker Studio por conta de sua acessibilidade para programadores novos, como nós, isso sem contar que é muito versátil e fácil de usar, contendo várias funções prontas, mas não abre mão das capacidade de criar jogos grandes, como Pizza Tower, Hyper Light Drifter, Deltarune, Moonleap, Katana Zero, Forager, Hotline Miami e mais.

A linguagem do GameMaker (GameMaker Language) é bem simples de entender, mesmo se você não entender programação, e pelo fato de ter similaridades nas linguagens que conhecemos, decidimos ficar com o GameMaker.

Para os sprites, usamos o Aseprite, um programa de pixel art bem famoso, fácil de usar e manipular as imagens. O Aseprite tem uma versão paga, custando cerca de 30 reais, e uma versão open source no GitHub, porém tem que ser compilada. Nós optamos pela versão compilada do GitHub por conta da falta de dinheiro nossa.

Sobre a IA, usamos o Gemini para ajudar a compreender melhor alguns problemas que encontramos no caminho, não usamos ela para gerar sprites ou

programar para nós, apenas para explicar conceitos que não ficaram muito esclarecidos.

## 2.3 ATAS E REUNIÕES

Sobre as decisões que fizemos durante o processo do jogo, nós dividimos as funções para cada integrante: Lorenzo e Miguel com a arte, Lucas com o design do jogo, Miguel, Nicholas e Thales com a programação, e Nicollas com o website.

No início do projeto, nós queríamos fazer o jogo na Unity, que é conhecido pela sua complexidade e maior qualidade de jogos, por conta que poderíamos usar sprites feitos à mão. Mas depois de pouco tempo decidimos fazer na GameMaker por conta do quão complexa que a Unity era.

Nós discutimos a história do jogo na primeira semana, que deixamos com o Lorenzo para pensar melhor na história.

Na primeira semana, nós fizemos as mecânicas básicas, como andar e pular, além de implementar colisões básicas no jogo.

Na segunda semana, adicionamos mecânicas um pouco mais complexas, como o dash, o pulo duplo e o pulo na parede, além de criar uma versão inicial. Além de fazer alguns inimigos básicos, como o de patrulha e de perseguição.

Na terceira semana, nós tentamos implementar algumas outras mecânicas mais complexas, como plainar, deslizar, plataformas móveis e semi-sólidas, mas tivemos que descartá-las por conta que não seriam utilizadas essas mecânicas e por conta do tamanho do mapa que havíamos planejado.

Na quarta semana, nós fizemos o inimigo que atira e um protótipo do ataque do jogador. E o Lorenzo começou a fazer os sprites do personagem principal.

Na quinta semana, finalizamos o ataque do jogador e construímos o mapa no jogo em si. Além de já tentar fazer a troca de personagem.



Na sexta semana, quase terminamos a mecânica de troca de personagem. Implementamos definitivamente o dano e adicionamos a mecânica de parry, que permite o jogador a refletir a bala do inimigo que atira. E o Lorenzo terminou os sprites do personagem principal

Na sétima semana, finalizamos o jogo finalizando o sistema de troca de personagem e terminado o tileset do mapa.

### 3 DESENVOLVIMENTO

O desenvolvimento foi subdividido em três partes fundamentais, o design, a arte e a lógica.

Para melhor dinâmica do documento, resolvemos fazê-los como subseções deste tópico

#### 3.1. DESIGN E ARTE

O estilo visual escolhido é a pixel art, justamente devido à menor complexidade para sua produção.

Inicialmente a paleta de cores seria inteiramente preta e branca, entretanto, devido à inflexibilidade de tal, foi optado por uma palheta relacionada a cavernas, usando tons de verde musgo e marrom terra. Porém, não foi possível criar o cenário, e foi preciso pegar um tileset já pronto, feito pelo usuário TheFlavare.

Figura 3 - Exemplo de ambientação do jogo



Disponível em: Fonte do autor.

Os personagens foram criados de maneira autoral pelo integrante Lorenzo Teixeira, que assumiu a liderança artística e criativa.

O primeiro sprite disponível foi o do inimigo de perseguição, que, junto ao seu código, foi o primeiro a ser implementado ao jogo. Seu design foi baseado em figuras de palito

Figura 4 - Design do inimigo perseguidor



Disponível em: Fonte do autor.

O segundo sprite a ser concluído foi do inimigo que anda pela sala, este design acabou passando por alterações e esse foi o produto final. O produto final foi baseado em uma foca.

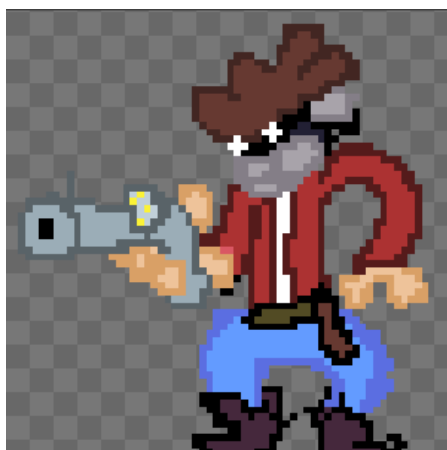
Figura 5 - Design do inimigo que vaga



Disponível em: Fonte do autor.

O próximo foi do inimigo atirador, que teve seu design baseado em filmes de faroeste.

Figura 6 - Design do atirador



Disponível em: Fonte do autor.

Depois foi iniciado o design do chefe final, que foi inspirado em estrelas e em um estilo casual, fazendo uso de um tom mais pastel. A mão e o corpo são partes separadas do corpo, para maior facilidade na programação do projeto.

Figura 7 - Design do boss



Disponível em: Fonte do autor.

O último sprite concluído foi do personagem principal, que estava disponível junto às suas animações em duas paletas devido a sua mecânica de troca.

Figura 8 - Design do personagem principal na forma yang



Disponível em: Fonte do autor.

Figura 9 - Design do personagem principal na forma yin



Disponível em: Fonte do autor.

O personagem principal é baseado no axolote, com sua paleta sendo baseada no símbolo do equilíbrio, o yin yang.

Houveram também outros sprites, entretanto, com menor necessidade de detalhamento, mas ainda de grande importância no produto final.

## 3.2 ARQUITETURA E CÓDIGO

Nós usamos muitos condicionais e laços de repetição na programação do nosso jogo para fazer as funcionalidades do jogo. Por exemplo, usamos vários ifs para fazer a colisão do jogo, onde se verifica um pouco a frente do personagem se existe uma parede ou colisão e, se existe, vai um pouco para frente até que não seja possível ir mais para frente. Para rampas, é uma lógica muito parecida, onde se verifica à frente do personagem e acima dele, onde, se existir uma rampa à frente e for possível subir nela, se sobe nela.

Nós fizemos essa parte da colisão ser uma função geral em que se podia usar mais facilmente ao invés de ter que fazer todo o código da colisão toda vez que fosse necessário uma colisão.

Como o GameMaker é muito simples, todas as mecânicas que fizemos tivemos que fazer totalmente a mão, como a gravidade, onde outras engines tinham funções próprias para isso.

Um exemplo do código que fizemos é o de movimentação, onde se o usuário estiver segurando o botão de se mover e segurando o botão de correr, ele vai ter a velocidade de corrida aplicada nele, se ele não estiver segurando o botão de corrida, ele vai ter a velocidade de andar aplicada a ele. Este estando logo abaixo

```
if xInput!=0{  
    if _sprintKey{  
        xSpd = xInput * mv[1];  
    }  
    else{  
        xSpd = xInput * mv[0];  
    }  
}  
  
else if xInput = 0{xSpd=0}
```

### 3.3 O SITE DE DIVULGAÇÃO

O website do jogo Aequilibrium tem como principal objetivo a divulgação do jogo. Faz isso por meio da sua história cativante e design minimalista.

#### 3.3.1 DEMONSTRAÇÃO DO WEBSITE

Para melhor entendimento da dinâmica do website, serão usadas figuras prévias às explicações. Começaremos a explicação pela landing page.

Figura 10 - Página inicial do website



Disponível em: <https://aequilibriumgame.netlify.app/>

Nessa primeira figura é possível observar a página inicial do website ao abri-lo, contando com sua barra de navegação e introdução à história, com uma figura ilustrativa ao lado do texto.

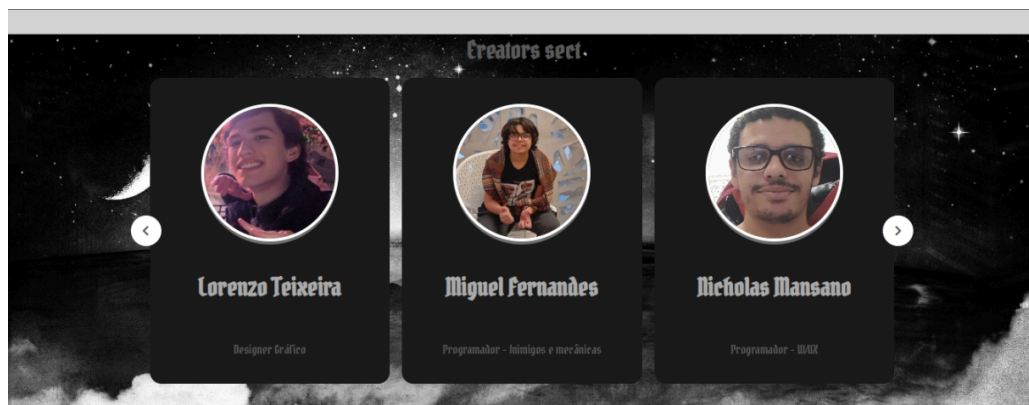
Figura 11 - Subseção dos personagens



Disponível em: <https://aequilibriumgame.netlify.app/>

Na segunda figura é possível observar a seção explicativa dos websites, ela conta com um carrossel que passa pelas figuras automaticamente, demonstrando os diferentes sprites do jogador de maneira animada, graças aos gifs.

Figura 12 - Subseção dos criadores



Disponível em: <https://aequilibriumgame.netlify.app/>

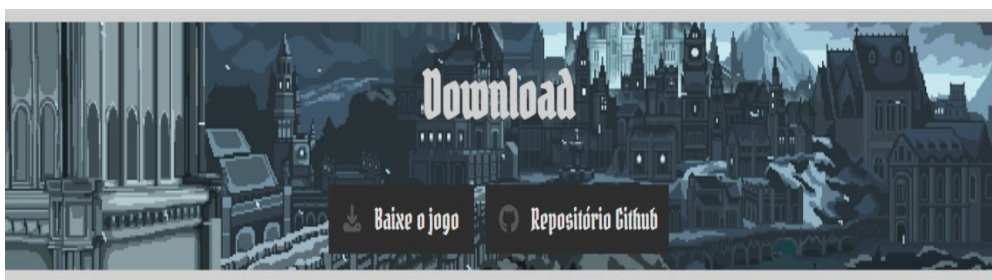
Nesta terceira figura é possível observar a seção dos criadores, que conta também com um carrossel de botões.



Para a realização de tal feito foi necessário a integração ao Javascript, que permitiu a funcionalidade dos botões e a sua função de “scroll infinito”.

Também é possível observar a foto dos criadores e suas respectivas funções resumidas de maneira breve.

Figura 13 - Subseção do download



Disponível em: <https://aequilibriumgame.netlify.app/>

A seção de download é a responsável por entrelaçar o repositório github e o executável do jogo, permitindo assim que o usuário teste o jogo e veja seu código fonte.

Figura 14 - Subseção do footer



Disponível em: <https://aequilibriumgame.netlify.app/>

É a sessão final do website, em que é possível visualizar as redes sociais dos criadores, contactá-los e usar os links rápidos para ir às outras partes da página.

### **3.3.2 TECNOLOGIAS USADAS**

No website as tecnologias usadas são o padrão do desenvolvimento web, que são compostas de: HyperText Markup Language em sua 5ª versão (HTML5), Cascading Style Sheets na 3ª versão (CSS3) e a linguagem de programação Javascript (JS).

O website é organizado por tipagem dinâmica, contando com tags como <nav>, <main>, <footer> e etc. O website como um todo segue uma estrutura simples, usando class="classe-wrap" para referir-se a um container e class="classe-card"

Além disso, não houve o uso de frameworks como o Bootstrap, Bulma ou Tailwind, apenas uma tentativa prévia, que foi descartada, do framework Slick Slider.

### **3.3.3 COMENTÁRIOS SOBRE O WEBSITE**

Graças à irresponsabilidade de um dos membros da equipe, encarregado de ficar com o website inicialmente, foi necessário uma manutenção completa de tal e a responsabilidade da tarefa a outro participante.

O website completo foi refeito nos 6 dias prévios à entrega do projeto, saindo com menos funções do que originalmente planejado.

## **4 PROTÓTIPOS E TESTES**

Durante os testes do jogo foram encontrados muitos erros (os chamados bugs), principalmente de colisão, onde o jogador simplesmente atravessava as paredes ou se arrastava nas paredes, por exemplo.

Além desses bugs, houveram, também, problemas com o ataque do personagem, em que, se ele estivesse correndo e começasse a atacar, ele iria atacar infinitamente e flutuar caso ele saísse do chão, outro problema que houve com o ataque foi que a colisão do ataque quando o personagem atacava para a esquerda, aparecia na direita do personagem.

Também aconteceram muitos bugs com o conceito de instâncias, bugs envolvendo a direção dos projéteis e o parry, e as mãos do boss, que tiveram muitos bugs em relação a direção do primeiro ataque. Mas quando esse conceito ficou claro, nós conseguimos trabalhar melhor, envolvendo portas e gatilhos.

Esses são só alguns dos bugs que ocorreram durante os testes, porque houveram problemas na animação, problemas na própria programação que fazia com que o jogo “colapsasse” e fechasse.

## 5 CONCLUSÃO

Em conclusão, nós aprimoramos muito nosso conhecimento na área de programação por conta do quanto que tivemos que resolver problemas e consertar erros que existiam no nosso código.

Gostamos muito de fazer o jogo por conta do quão livre estávamos para fazer o que quiséssemos no jogo, do jeito que a gente queria e, também, podíamos aprender como os jogos que jogávamos antes funcionam de forma parcial, já que estávamos fazendo as mecânicas do jogo nós mesmos.

Nós gostamos do resultado final que nosso jogo teve, mas queríamos ter mais tempo para que pudéssemos deixar o jogo mais polido e mais completo, já que tínhamos muitas outras ideias em mente mas que não puderam ser implementadas por conta da restrição de tempo.

Tivemos alguns problemas com um dos integrantes da equipe por conta da irresponsabilidade dele, já que ele foi encarregado de fazer o website e ele não era nem responsivo e uma porção não desejável do website foi feita por IA. Além disso, ele ficou encarregado, também, de fazer a documentação, mas ela estava incoerente com o que realmente havia acontecido e, ainda mais, estava completamente fora do modelo que foi proposto.

## REFERÊNCIAS

BURNHAM, Peyton, **How to Make a 2D Platformer in GameMaker!**. Disponível em: <<https://youtube.com/playlist?list=PL14Yj-e2sgzxXOwdMYC0IDuG9m-VHEMW2&si=2ZSUYNMgDeFxAAbi>> Acesso em: 22 out. 2025.

GAMEMAKER, **How to make a main menu in GameMaker**. Disponível em: <<https://youtu.be/Us5GSddVedY>> Acesso em: 27 out. 2025.

**GameMaker Manual**. Disponível em: <<https://manual.gamemaker.io/monthly/en/>> Acesso em: 5 nov. 2025

NONECLASS, **GAMEFEEL: HITSTOP [ Game Maker ]**. Disponível em: <<https://www.youtube.com/watch?v=GurGFptvGk&t=162s>> Acesso em: 4 nov. 2025

SPALDING, Sara, **GameMaker Studio 2: Melee Attacks Tutorial**. Disponível em: <<https://www.youtube.com/watch?v=93Dz2hgsZy0>> Acesso em: 15 out. 2025

SPALDING, Sara. **How to make an Air Dash mechanic in GameMaker - Tutorial**. Disponível em: <<https://www.youtube.com/watch?v=LRt2K9459MM>>. Acesso em: 6 out. 2025