

# DOCUMENTO PARA DESIGN DE GAMES

<TECH PATH>

Autores: <Felipe Silberberg; Jordan Andrade; Marcos Ferreira; Ueliton Rocha; Valdomiro Filho; Vitor  
Barros; Wagner Estevam>

Data de criação:<7 de fevereiro de 2022>

## Controle do Documento

### Histórico de revisões

Data	Autor	Versão	Resumo da atividade
07/02/2022	Felipe Silberberg; Marcos Ferreira	1.0	Preenchimento dos dados iniciais do documento
10/02/2022	Grupo	1.1	Preenchimento dos campos 1.1 a 1.5
16/02/2022	Grupo	1.2	Preenchimento da seção 1.7
17/02/2022	Grupo	1.3	Preenchimento da seção 'Value proposition canvas'
22/02/2022	Grupo	1.4	Preenchimento de alguns pontos da seção 2 a 4

---

## Sumário

<b>1. Visão Geral do Projeto &lt;ADALOVE – Definir Proposta de Valor&gt;</b>	<b>6</b>
1.1 Objetivos do Jogo	6
1.2 Características gerais do Jogo	6
1.3 Público-alvo	6
1.4 Diferenciais	6
1.5 Análise do cenário: Matriz SWOT	6
1.6 Proposta de Valor: Value Proposition Canvas	6
1.7 Requisitos do Jogo <ADALOVE – Documentar requisitos>	7
1.7.1 Requisitos coletados na entrevista com o cliente	7
1.7.2 Persona	7
1.7.3 Gênero do Jogo	7
1.7.4 Histórias do jogo (Game stories) ou Histórias dos usuários (user stories)	7
1.7.5 Mecânica	7
1.7.6 Fontes de Pesquisa / Imersão	7
<b>2. Game Design &lt;ADALOVE – Elaborar fluxograma do jogo&gt;</b>	<b>8</b>
2.1 História do Jogo	8
2.2 Fluxo do Jogo e Níveis (os níveis são opcionais)	8
2.3 O Mundo do Jogo	8
2.3.1 Locações Principais e Mapa	8
2.3.2 Navegação pelo Mundo	9
2.3.3 Escala	9
2.3.4 Ambientação	9
2.3.5 Tempo	9
2.4 Base de Dados	9
2.4.1 Inventário	9
2.4.1.1 Itens Consumíveis (opcional)	9
2.4.1.2 Armamento (opcional)	10

2.4.2 Bestiário (opcional)	10
2.4.2.1 Inimigos Elementais de Água	11
2.4.3 Balanceamento de Recursos (opcional)	11
<b>3. Level Design (opcional) &lt;ADALOVE – Implementar mecânicas básicas do jogo - parte 1 a 5&gt;</b>	<b>13</b>
3.1 Fase <NOME DA FASE 1>	13
3.1.1 Visão Geral (opcional)	13
3.1.2 Layout Área (opcional)	13
3.1.2.1 Connections (opcional)	13
3.1.2.2 Layout Effects (opcional)	13
3.1.2.3 Quests e Puzzles (opcional)	13
3.1.3 Balanceamento de Recursos (opcional)	13
3.1.4 The Boss	14
3.1.5 Outros Personagens	14
3.1.6 Easter Eggs	15
<b>4. Personagens &lt;ADALOVE – Implementar mecânicas básicas do jogo - parte 1 a 5&gt;</b>	<b>16</b>
4.1 Personagens Controláveis	16
4.1.2 <NOME DO PERSONAGEM PRINCIPAL n>	16
4.1.2.1 Backstory	16
4.1.2.2 Concept Art	16
4.1.2.3 Ações Permitidas	16
4.1.2.4 Momento de Aparição	16
4.2 Common Non-Playable Characters (NPC)	17
4.2.1 <NOME DO NPC COMUM n>	17
4.3 Special Non-Playable Characters (NPC)	17
4.3.1 <NOME DO NPC ESPECIAL n>	17
<b>5. Teste de Usabilidade &lt;ADALOVE – Desenvolver relatório de resultados do playtest&gt;</b>	<b>18</b>
<b>6. Relatório - Física e Matemática</b>	<b>19</b>
6.1 Funções	19
6.2 Cinemática Unidimensional	19

6.3 Vetores	19
6.4 Cinemática Bidimensional e mais	19
<b>7. Bibliografias</b>	<b>20</b>
<b>Apêndice</b>	<b>21</b>

# 1. Visão Geral do Projeto <ADALOVE – Definir Proposta de Valor>

## 1.1 Objetivos do Jogo

O jogo tem por objetivo instruir e orientar jovens de ensino médio e jovens que querem se graduar a conhecer melhor os cursos voltados para a área de tecnologia oferecidos pelo Inteli (Engenharia de Software, Engenharia da Computação, Ciência da computação e Sistema de Informação) e, a partir da experiência e aprendizado dentro do jogo, poder responder às seguintes perguntas: O que cada uma faz? Quais suas diferenças? Quais as profissões/caminhos que cada um desses cursos proporcionam? Qual a visibilidade e relevância perante o mercado?

## 1.2 Características gerais do Jogo

- Jogo educativo.
- Sistema gamificado.
- Sistema de ranking.
- Possibilidade de mais de um caminho a ser seguido no jogo.
- Jogo mundo aberto em 2D.
- Dentro do jogo, haverá mini jogos referentes a cada profissão, explicando mais detalhadamente a função e diferença entre elas;
- Método lúdico de aprendizado.
- O jogo ensinará de maneira simples e descontraída a diferença e o que cada uma das profissões oferecidas pela inteli oferecem.

## 1.3 Público-alvo

Jovens do ensino médio/ jovens que acabaram de terminar o ensino médio interessados pela área na tecnologia, mas que não possuem um direcionamento e não sabem diferenciar os principais cursos da área da computação ( Engenharia de Software, Engenharia da Computação, Ciências da Computação e Sistema de Informação), nem quais carreiras e caminhos podem seguir a partir desses cursos.

## 1.4 Diferenciais

Descrever os diferenciais competitivos do jogo.

- Design semelhante ao de jogos arcade clássicos, o que desperta um sentimento de nostalgia nos usuários;
- O jogo explica a diferença entre cada curso de maneira lúdica e interativa, diferente de textos e vídeos longos e cansativos;

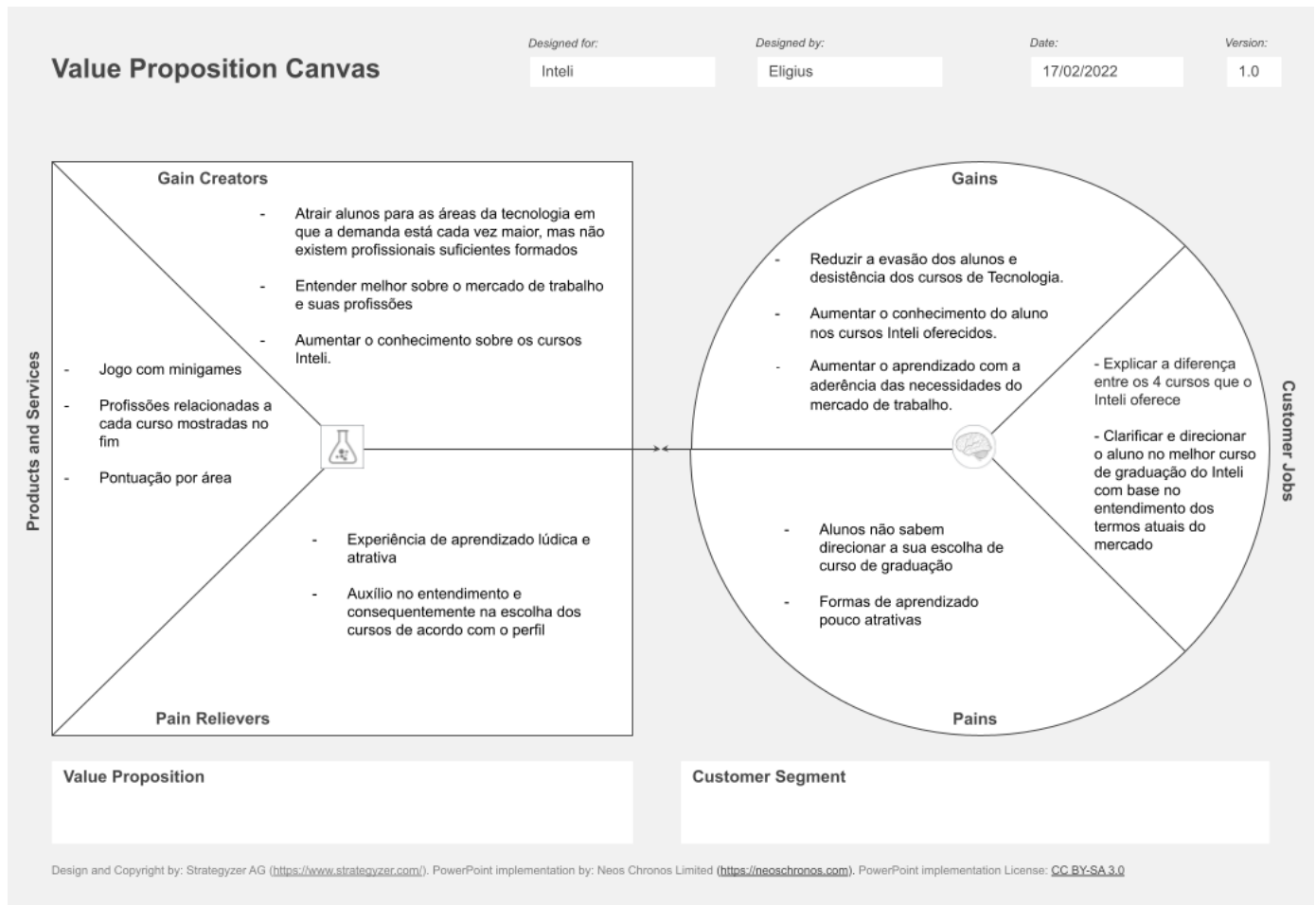
- Jogo com história interativa onde o usuário pode escolher os caminhos que quer seguir, prendendo o jogador na história.
- Método lúdico e interativo de aprendizado.

## 1.5 Análise do cenário: Matriz SWOT

Own specific factors					Environmental factors				
<b>Strengths</b>					<b>Opportunities</b>				
Jogo leve e possível de rodar em qualquer dispositivo; Fácil acesso a grandes nomes e empresas; Jogo simples, intuitivo e dinâmico; Infraestrutura moderna e tecnológica;					Ascensão do mercado tecnológico; Aumento do interesse pelas áreas de tecnologia; Demanda reprimida de informações nessas áreas; Proximidade de grandes empresas; Uso do jogo para fins didáticos nas escolas do ensino médio; Crescimento do mercado mobile;				
<b>Weaknesses</b>					<b>Threats</b>				
Jogo com público nichado; Inexperiência como faculdade					Pouco conhecida no mercado profissional Metodologia não convencional				



## 1.6 Proposta de Valor: Value Proposition Canvas



O Canvas da proposta de valor serve para ajudar a criar e organizar produtos/serviços que se alinhem com o que seu cliente realmente valoriza e precisa.

## 1.7 Requisitos do Jogo <ADALOVE – Documentar requisitos>

### 1.7.1 Requisitos coletados na entrevista com o cliente

Apresenta um resumo das respostas da entrevista com o parceiro de mercado.

- O jogo deve estimular a curiosidade do jogador, focando em sua experiência
- O jogo deve ter cuidado com os rótulos e relacionar profissões que não necessariamente possuem o nome exato de um curso.
- Deve ser abordada a questão do mercado de trabalho: relação curso x profissão/carreira, focando nos cursos de bacharelado.
- O jogo deve transmitir os valores de democracia e liberdade (autonomia de escolha).

- O objetivo do jogo é clarificar e direcionar o aluno no melhor curso de graduação do Inteli com base no entendimento dos termos atuais do mercado.
- Requisitos como plataforma, abordagem dos cursos (largura ou profundidade) e design ficam a critério dos grupos.

### 1.7.2 Persona

Lucas Silva	18 anos	Estudante	Zona Norte/São Paulo
-------------	---------	-----------	----------------------

- Nascido na zona norte de São Paulo, Lucas é um jovem apaixonado por games. Gosta muito de utilizar dos jogos para se entreter e fazer novas amizades. Sempre teve um bom desempenho escolar e se interessou muito pelas áreas de TI, embora tenha pouco conhecimento sobre o assunto. Tem dois irmãos mais novos, Larissa e Luís, os quais gosta muito de passar o tempo jogando, mas prefere os amigos pois é muito competitivo e sempre joga para vencer. Atualmente, Lucas estuda administração na ETEC por pressão dos pais, mas tem muitas dúvidas de qual curso realmente seguir e gostaria de saber um pouco mais sobre a área de TI para que possa realmente seguir sua vocação.
- Lucas gosta de cachorros, praticar esportes e também de filmes e séries da Marvel, é apaixonado por tecnologia, é fluente em inglês, gosta de jogar em qualquer oportunidade que tem e é ansioso.
- Lucas se sente motivado em jogar para desestressar, se divertir com amigos, pelo modo que se sente quando está jogando, principalmente a adrenalina, buscar melhorar e evoluir no jogo, e passar o tempo.
- Lucas odeia jogos difíceis de entender, pouco imersivos e com gráficos fracos.
- Ele está em dúvida sobre as diferenças entre os cursos de tecnologia. Ele já fez muitas pesquisas mas mesmo assim continua confuso. Além disso, não entende quais profissões ele poderá seguir a partir da sua formação.
- Por isso, espera entender melhor tudo isso para fazer uma escolha sábia e de acordo com seu perfil. Sendo um grande fã de games, a melhor forma dele conhecer mais sobre o assunto seria através de um jogo.
- Criamos a persona baseada em pessoas das quais acreditamos que tenham interesse futuro no nosso jogo e baseado também nas pesquisas realizadas, o personagem fictício é como se fosse uma pessoa específica do público alvo. Dando assim para o nosso grupo um detalhamento maior.

### 1.7.3 Gênero do Jogo

Definir o(s) gênero(s) do jogo, justificando a escolha de acordo com as características do(s) gênero(s) escolhido(s).

- Indie

- 
- Feito por equipe pequena de autoria própria com intuito educativo e sem fins lucrativos, o jogo é de acesso democrático com gráficos antigos e leves.
- 

#### **1.7.4 Histórias do jogo (Game stories) ou Histórias dos usuários (user stories)**

GS1. Como personagem principal, devo me movimentar até o final da tela para conseguir entrar no bunker. (parte 1)  
-> prioridade média

GS2. Como personagem principal, posso me aproximar dos computadores para habilitar a opção de selecionar uma tecla e entrar em um minigame. (parte 1) -> prioridade alta

GS3. Como um adolescente morador de um bunker, eu quero participar dos testes de carreira na área da tecnologia para poder ajudar na manutenção dessa sociedade subterrânea. (parte 1) -> prioridade média

GS4. Como um NPC especial, eu posso adaptar minha fala para estar mais próximo do jogador, permitindo uma imersão maior. (parte 2) -> prioridade média

GS5. Como personagem principal, eu devo fazer os minigames no menor tempo possível para ganhar mais pontos. (partes 2, 3, 4) -> prioridade alta

GS6. Como personagem principal, eu quero adquirir habilidades específicas de cada área para aumentar os pontos nela. (parte 2, 3, 4)

GS7. Como NPCs do jogo eu posso explicar de maneira simplificada e dinâmica diferenças entre os cursos para que o personagem entenda as diferenças dos cursos e o que cada um pode alcançar com maior facilidade. (parte 3, 4 e 5) -> prioridade média

GS8. Como personagem principal, eu posso escolher caminhos e habilidades diferentes que geram pontos para cada área para no final do game receber uma lista com as profissões e qual a pontuação em cada uma. (parte 4) -> prioridade alta

GS9. Como jogador, eu quero que o jogo me traga uma lista de profissões no final para que eu possa entender com qual(is) curso(s) de tech eu mais me aproximo. (parte 4) -> prioridade alta

GS10. Como personagem principal, eu devo jogar todos os minigames para passar de fase e, assim, completar o jogo. (parte 5) -> prioridade alta

GS11. Como jogador, eu posso seguir o caminho de cores para me guiar durante o jogo (parte 5) -> prioridade alta

GS12. Como jogador, eu posso pausar o jogo para voltar ao menu ou sair do jogo (parte 5) -> prioridade média

### 1.7.5 Mecânica

Descrever a forma de interação do jogo (mecânica), nos aspectos: personificação, dimensão, período, conectividade, visão.

---

- Jogo pixelado
  - Temática pós-apocalíptica
  - Jogo 2D - plataforma
  - O jogo tem sistema parecido com de um RPG onde você pode ir subindo de nível, escolhendo caminhos diferentes e escolhendo habilidades, gerando assim várias possibilidades e caminhos diferentes para serem seguidos.
  - No início do jogo terá uma opção de escolha de diferentes personagens.
  - O jogo será composto basicamente de 12 salas em que cada uma representará um minigame.
  - O personagem será um jovem que está tentando dominar tecnologias para ser aceito em um bunker tecnológico onde apenas quem tem esses conhecimentos é aceito.
  - Terá a possibilidade de resolver puzzles para conseguir mais XP e subir de nível mais rapidamente. Quanto mais rápido resolver, mais pontos ganha.
  - Todo o foco do jogo será educativo, com o intuito de explicar de forma dinâmica e descomplicada a diferença entre cada curso oferecido pela inteli.
- 

### 1.7.6 Fontes de Pesquisa / Imersão

Indicar as principais fontes de pesquisa do jogo para a criação de conteúdo (feitos em aula): imagens, filmes, animações, livros e outros que realmente foram usados para a etapa de imersão ao tema. Fazer um sumário do conteúdo pesquisado a partir dos referidos materiais.

---

Fonte

1. Pesquisa realizada pelo grupo

- a. Pesquisas qualitativas feitas por entrevistas individuais:

<https://drive.google.com/file/d/17ehsGhWrGmCxYOQCTAeZ1d8DRPSggghG/view?usp=drivesdk>

- b. Pesquisa quantitativa feita a partir das respostas das pesquisas qualitativas:

<https://forms.gle/GBPzx82MbutNXbLb7>

2. Pesquisa sobre jogos de plataforma 2D

- a. <https://youtu.be/AQMEuVva4L8>  
b. <https://youtu.be/pQ8GXn7JIMo>  
c. [\[PDF\] up.pt](#)

[\[PDF\] Jogo de Plataformas 2D](#)

3.

---

## 2. Game Design <ADALOVE – Elaborar fluxograma do jogo>

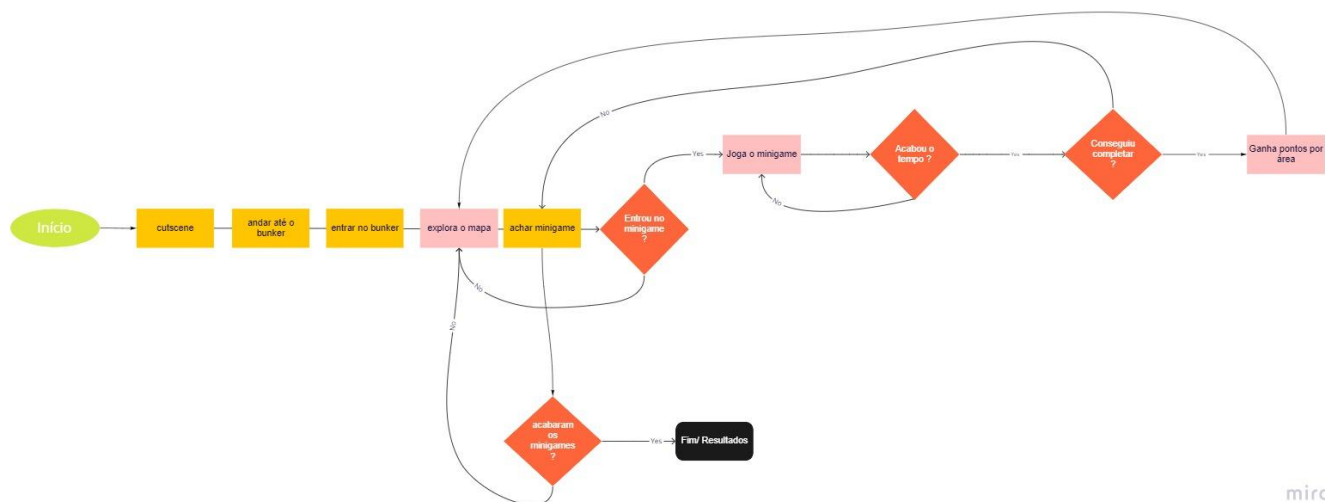
### 2.1 História do Jogo

Descrever os seguintes aspectos:

- Tema (*storyline*) - o jogador tentará se encaixar em um bunker tecnológico, exercendo alguma função essencial na área de computação.
- Conceito - O jogo tem o intuito de fazer o jogador aprender sobre as profissões oferecidas pelo Inteli. O jogador terá que explorar o mundo (bunker) com o intuito de descobrir novas salas e adquirir novas habilidades.
- Pano de fundo da história (*backstory*) - Um sistema de inteligência artificial, criado por uma cientista, hackeou os códigos dos mísseis nucleares e os lançou ao redor da terra, pois a A.I acreditava que o problema do mundo era a quantidade excessiva de pessoas. Como consequência, a superfície da terra se tornou inabitável gradualmente em decorrência do aumento da radiação. Dessa forma, um bunker tecnológico foi construído e começou a ser habitado por pessoas previamente selecionadas, que deveriam exercer funções específicas para a manutenção daquela sociedade.
- Premissa - Como um adolescente morador de um bunker, o jogador irá participar de testes de carreira na área da tecnologia para poder ajudar na manutenção de uma sociedade subterrânea, criada após a revolta das máquinas.
- Sinopse - Após um apocalipse nuclear causado por uma Inteligência Artificial, os humanos criaram um bunker tecnológico. Para garantir sua manutenção, é preciso funcionar como uma sociedade, com cada indivíduo exercendo uma função essencial. No entanto, a falta de pessoas especializadas em computação impede seu desenvolvimento e prosperidade.
- Estrutura narrativa escolhida - O jogador vai contar com a companhia de uma narrador e de NPCs, que vão acompanhá-lo durante a sua jornada explicando a história, dando conselhos e explicando conceitos sobre as profissões.
- Elementos do roteiro para a estrutura narrativa escolhida (animação, cut-scenes, McGuffin, diálogos, foreshadowing, inciting incident, etc) – Os principais elementos que o jogador vai deparar são cutscenes e diálogos com outros personagens para entender e entrar na história do jogo.
- Níveis de interatividade do jogo - O jogador terá um nível de interatividade no jogo relativamente alto, podendo desbloquear novos espaços, realizar puzzles e interagir com NPCs que vão disponibilizar novas missões ou novas habilidades, com o intuito de que o jogador vá ganhando mais conhecimento sobre as profissões ao decorrer do jogo.

## 2.2 Fluxo do Jogo e Níveis (os níveis são opcionais)

No início do jogo haverá uma introdução da história com cutscenes. Depois, o jogador terá que andar até o bunker (apenas para frente), contemplando o cenário do enredo. Entrando no bunker o jogador poderá explorar o mapa à procura de minigames. Esses minigames devem ser completados no menor tempo possível para ganhar mais pontos por curso. Depois de completar todos os minigames o jogo chegará ao fim e será mostrado um resumo dos pontos em cada área, mostrando as profissões daquela com mais pontos..



O flowchart foi baseado tanto nas ações do jogador, o que ele deve fazer para completar o jogo, o que acontece se ele perder ou ganhar um minigame, quanto na narrativa. Algumas ações representam o desenrolar da história do jogo..

Fazer o *flowchart* do jogo (grafo representando o fluxo do jogo) e descrever que tipo de flowchart escolheu: baseado em ações, em quests, na narrativa etc? Justificar de acordo com o gênero escolhido. Por fim, quanto tempo o jogador deverá despendar com o jogo?

## 2.3 O Mundo do Jogo

### 2.3.1 Locações Principais e Mapa \*\*

O jogo irá se passar dentro de um bunker tecnológico em um contexto de cenário pós-apocalíptico. O mapa onde o jogo irá se passar será o próprio bunker. Dentro do bunker existirá vários andares, e dentro dos andares existirão algumas salas, onde ficarão as missões e os minigames, além de corredores por onde o personagem pode transitar e explorar o mapa do jogo, com uma vista de cima do mapa e em terceira pessoa. Com o desenvolvimento do jogador, ele subirá para os andares mais altos do bunker, desbloqueando novas salas e desafios mais difíceis que os dos andares anteriores, além de salas escondidas em alguns lugares com desafios extras que liberam bônus para o personagem

Descrever as locações principais do jogo e o mapa do mundo dele (não são os mapas das fases e sim do mundo!).  
Apresentar um *flowchart* do mundo.

### **2.3.2 Navegação pelo Mundo**

O jogador se movimenta livremente pelo mapa, podendo se mover em todas as direções (esquerda, direita, para cima e para baixo), de modo a explorar o mapa e procurar por missões, desafios e salas escondidas. Ao se aproximar de uma sala desbloqueada, deverá apertar uma tecla para entrar na sala, tendo assim um corte de transição de um ambiente para o outro. Os minigames possuem dinâmicas e controles diversos dependendo do objetivo daquele game. Além disso, o tempo de realização dos minigames será contado, podendo haver recompensas para o desempenho do jogador. Ao finalizar todos os minigames essenciais do andar, uma porta com um novo andar é desbloqueada, cabendo ao jogador se dirigir à essa porta para ir para o próximo andar.

### **2.3.3 Escala**

As escalas usadas vão variar de acordo com o andar, mas todas estão entre 1:1,5 e 1:2 dependendo do andar e do objeto que está no cenário.

\*Descrever a escala usada para representar o mundo do jogo. Exemplo: os personagens são minúsculos em relação ao mundo, sendo que as portas são, por exemplo, 2 vezes maiores que a altura deles. Escala 1:2.

### **2.3.4 Ambientação**

Por a ambientação do jogo ser dentro de um bunker, será possível encontrar um ambiente com temática mais escura, tecnológica, lugares empoeirados. O bunker será dividido em andares onde cada andar terá a sua determinada funcionalidade (por exemplo, pode-se encontrar um andar com foco em tecnologia onde terá uma ambientação com mais laboratórios e computadores, em outro podemos encontrar uma ambientação como se fosse a parte de armazenamento então encontraremos muitas comidas enlatadas).

Condições climáticas do mundo do jogo (se aplicáveis – verão, inverno? Dia ou noite?), condições vegetais, animais. Definir se tais condições serão estáticas (por exemplo, acontecerá uma “chuva” sempre no mesmo ponto) ou dinâmicas (por exemplo, em determinado trecho de uma fase pode aparecer uma “chuva” com uma determinada probabilidade).



### 2.3.5 Tempo

A pontuação dos minigames será estabelecida através da qualidade do trabalho feito pelo usuário, bem como do tempo levado para concluí-lo. Além disso, a pontuação será inversamente proporcional ao tempo gasto no minigame.

## 2.4 Base de Dados


### 2.4.1 Inventário

O jogo não terá itens coletáveis.


Descrever cada um dos itens do jogo, apresentando suas características principais e usadas para programar o jogo (características que de fato afetam o funcionamento do jogo; por exemplo, suponha que o peso seja fator relevante no jogo de forma que o jogador não possa carregar muitos objetos ao mesmo tempo, portanto, listar o peso de cada item é importante porque esse é um fator relevante no jogo). É necessário que se agrupem os itens por similaridade, para facilitar consulta e organização: arma, dinheiro, itens consumíveis (cura, magia etc) entre outros. Fazer uma tabela a distribuição dos objetos do jogo todo. A seguir um exemplo de listagem de itens do jogo.

#### 2.4.1.1 Itens Consumíveis (*opcional*)


Poção Medicinal Pequena

	Descrição	Uma pequena poção que cura um pouco de energia.
	Peso	5
	Valor de cura	10

Poção Medicinal Média


	Descrição	Uma pequena poção que cura um pouco de energia.
	Peso	10
	Valor de cura	15

Poção Medicinal Grande


	Descrição	Uma pequena poção que cura um pouco de energia.
	Peso	20
	Valor de cura	25

### 2.4.1.2 Armamento (opcional)


#### Espada de Gelo

	Descrição	Permite congelar o inimigo. Chance de 30%.
	Peso	20
	Ataque	20

#### Espada de Fogo

	Descrição	Permite lançar fogo nos inimigos. Chance de 30%.
	Peso	20
	Ataque	20

#### Espada Imaterial


	Descrição	Permite atacar monstros do tipo <i>Fantasma</i> .
	Peso	15
	Ataque	10

### 2.4.2 Bestiário (opcional)


Descrever os inimigos do jogo apresentando, da mesma forma que foi feita para a listagem de itens, os fatores realmente relevantes para o jogo. A seguir alguns exemplos.

#### 2.4.2.1 Inimigos Elementais de Água

##### Geleca Azul

	Descrição	Uma geleca azul. Tocar sua pele fria pode te congelar!
	HP	1600
	Defesa	10

##### Peixe Esfomeado

	Descrição	Um peixe faminto e raquítico.
	HP	2500
	Defesa	20

### 2.4.3 Balanceamento de Recursos (*opcional*)

Apresentar as tabelas de balanceamento aprendidas em sala (depende do estilo do jogo). Apresentar outros aspectos que as tabelas não conseguem solucionar e mostrar as soluções adotadas. A seguir estão os exemplos de tabelas vistas em sala.

#### Enemy Chart

Tipo Inimigo	W*	Fase 1	Fase 2	Fase 3	Fase 4	Fase 5	Fase 6	Total
Geleca Verde	1	3	5	2				10
Geleca Azul	2	1	5	5	5	2		15
Morcego	5			1	5	5	10	21
Flor-bomba	10					2	5	7
<b>Total</b>		5	15	17	35	49	100	

(No caso do *level design*, essa tabela desmembra-se em “inimigos x área por fase”).

\*Lembre-se que devemos colocar a quantidade de itens vezes o seu peso (W). A equação de dificuldade do inimigo ou peso (W) deve levar em consideração diversas questões, como: sua IA, seu HP, sua resistência, sua velocidade de ataque, itens que pode liberar (*drop*) para o jogador etc.

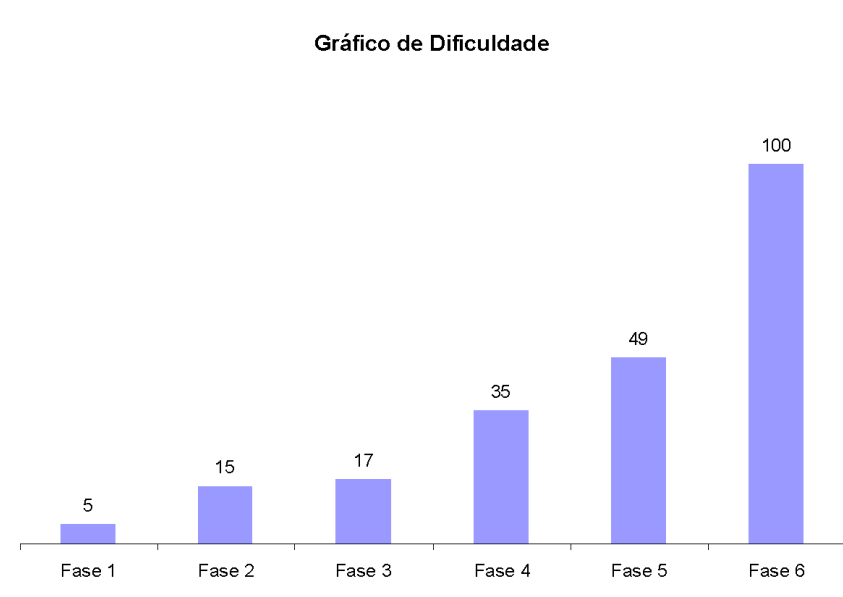


Figura 2. Gráfico de dificuldade para o jogo NOME DO JOGO.

Discutir, baseado no gráfico acima, se o balanceamento escolhido está de acordo com as teorias apresentadas por Mihaly.

Deve repetir a tabela “Enemy chart” para itens, quests, skills etc, ou seja, dependendo do tipo de jogo podem surgir outras tabelas bem como algumas desaparecer.

#### Item Chart

Tipo de Item	Fase 1	Fase 2	Fase 3	Fase 4	Fase 5	Fase 6	Total
Moeda	20	20	50	20	40	50	200
Sorvete	3	5	5	2	2	3	20
Bombinha			2			3	5
Super Estrela	1	1	1	1	1	1	6

---

## 3. Level Design (opcional) <ADALOVE – Implementar mecânicas básicas do jogo - parte 1 a 5>

Não aplicável

Apresentação do World Diagram para, logo a seguir, especificar cada fase. Pode-se acrescentar o gameflow nessa parte, levando-se em consideração o tipo de gameflow a ser trabalhado (quest, narrativa, ação etc).

### 3.1 Fase <NOME DA FASE 1>

#### 3.1.1 Visão Geral (opcional)

Construção do *layout área* dessa fase com informações de quais áreas estão ligadas a quais. Descrever o cenário desta fase: onde no mundo fica o local, como o personagem chegou ali, como é a vegetação, a temperatura etc.

Definir a meta (objetivo) do jogador na fase. Detalhar micro metas, se houver.

Descrição de onde o personagem inicia a fase, o que ele deve fazer para concluir a fase.

#### 3.1.2 Layout Área (opcional)

Construção do *layout área* dessa fase com informações de quais áreas estão ligadas a quais áreas, sem se importar com itens ou o formato e detalhes de objetos da área.

##### 3.1.2.1 Connections (opcional)

Construção do cenário usando *connections*. Também apresentar, se for o caso, do uso das técnicas de visibilidade de cena adotadas (caixotes obstruindo visão, escadas verticais, corredores/donut rooms, portas, ambientes obscuros etc).

##### 3.1.2.2 Layout Effects (opcional)

Legenda com informações de efeitos visuais/sons/animações CG no jogo.

##### 3.1.2.3 Quests e Puzzles (opcional)

Construção de quests/puzzles utilizando o quest/puzzle flow.

#### 3.1.3 Balanceamento de Recursos (opcional)

Posicionamento de itens na fase, utilizando o layout area como base. Usar legenda especificando todos os itens/inimigos (pode-se criar uma classificação, por exemplo, de inimigos tipo 1 – fáceis e inimigos tipo 2 – difíceis). Inserir tabela com os inimigos e a quantidade destes inimigos na fase divididos por área, para controle do balanceamento da mesma. Exemplo:

---

### Enemy Chart

Tipo Inimigo	W*	Área1	Área 2	Área 3	Área 4	Área 5	Área 6	Total
Geleca Verde	1	1	1	2		1	2	7
Geleca Azul	2		1		1	2	1	5
Morcego	5						1	1
Flor-bomba	10							0
<b>Total</b>		1	3	2	2	5	9	

Do mesmo modo, inserir tabela com os itens e a quantidade destes itens na fase por área, para controle do balanceamento da mesma.

### Item Chart

Tipo de Item	Área1	Área 2	Área 3	Área 4	Área 5	Área 6	Total
Moeda	5	2	3	5			15
Sorvete	1		1		1		3
Bombinha			2			1	3
Super Estrela					1		1

#### 3.1.4 The Boss

Descrever o chefe da fase (se houver) e seu comportamento de ataque/defesa, bem como o modo previsto para o jogador derrotá-lo e a recompensa adquirida (power-up, vida, itens diversos etc).

#### 3.1.5 Outros Personagens

Descrever quais e onde estão os NPCs e as ações que eles assumem perante o jogador/situação.

Definir como se dá a interação com o personagem. Criar os diálogos do NPC para a fase.

### **3.1.6 Easter Eggs**

Haverá salas escondidas no jogo com minigames,mas ainda não foi definido o que terá em cada uma delas.

Descrever locais/itens secretos na fase (se houver) e a forma para alcançá-los, bem como a recompensa adquirida (power-up, vida, itens diversos etc).

---

---

## 4. Personagens <ADALOVE – Implementar mecânicas básicas do jogo - parte 1 a 5>

Aqui, descrever brevemente a relação dos personagens na história e uma tabela com os personagens do jogo, apontando a fase em que aparecem (se houver personagens).

**Character Appearance Chart**

Personagem	Fase 1	Fase 2	Fase 3	Fase 4	Fase 5	Fase 6	Fase 7
Mario							
Luigi							
Toadstool							
Koppa							

### 4.1 Personagens Controláveis

#### 4.1.2 <Thomas Griffin>

Para cada personagem (se houver mais de um), descrever como foi criado, qual é a sua *backstory*. É interessante que apareça os esboços (desenhos) do mesmo. Deve existir algum mecanismo inicial para a seleção de personagem, quando for o caso. Deve permitir seleção de itens básicos iniciais para o personagem, quando for o caso. Para cada personagem, detalhar:

##### 4.1.2.1 Backstory

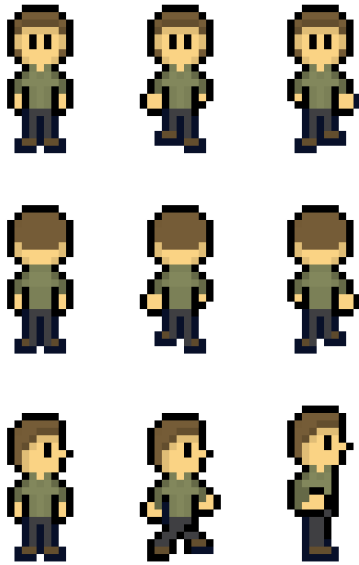
Backstory (pano de fundo) do personagem.

Thomas Griffin é um personagem adolescente, de 18 anos, que começou a viver em um bunker após uma A.I hackear os códigos de lançamento de mísseis nucleares, pois acreditava que o problema do mundo era a quantidade excessiva de pessoas. Desse modo, ele cresceu alguns anos em uma base subterrânea, porque a superfície se tornou inabitável por causa da radiação.

Thomas cresceu com alguns outros adolescentes, seguindo uma educação restrita, pois cada pessoa do bunker, ao se tornar adulta, deveria assumir uma função específica para a manutenção daquela sociedade. Os pais dele morreram no “apocalipse nuclear” e esse é um dos motivos pelo qual ele tem interesse pela área da tecnologia, pois quer entender o que deu errado com a inteligência artificial. Além disso, o interesse dele foi ao encontro das necessidades do bunker, que estava necessitando de pessoas nessa área para a proteção e comunicação do lugar.



#### 4.1.2.2 Concept Art



Esboços do personagem.

#### 4.1.2.3 Ações Permitidas

Andar em todas as direções( esquerda e direita; para cima e para baixo ); abrir áreas do mapa; abrir minigames.  
Habilidades físicas/ações no jogo (tem que estar relacionadas à psicologia e à sociologia do personagem).

#### 4.1.2.4 Momento de Aparição

O personagem irá aparecer logo após a primeira cutscene do jogo. Após isso ele estará presente por todo o jogo, menos durante a realização dos minigames, que terão uma vista representando uma visão em primeira pessoa.

Momento em que o personagem vai aparecer (identificar de acordo com as fases planejadas, utilizar o apoio do flowchart, se necessário)

### 4.2 Common Non-Playable Characters (NPC)

#### 4.2.1 <NOME DO NPC COMUM *n*>

Ainda não foram definidos os NPCs do jogo.

Para cada NPC comum, descrever sua história, comportamento perante o personagem (agressivo, amistoso, indiferente etc), seus dados pessoais como pontos de vida e outros que forem implementados (pontos de magia,

habilidades etc). O comportamento do personagem será estático ou dinâmico. Se dinâmico definir como o mesmo deve ser alterado.

## **4.3 Special Non-Playable Characters (NPC)**

### **4.3.1 <NOME DO NPC ESPECIAL *n*>**

O jogo terá um NPC especial que será um guia para o personagem principal, que trará dicas e informações para ajudá-lo durante a história. Informações mais específicas sobre esse NPC não foram definidas.

Para cada NPC especial (mini-boss, boss, mentor/guia etc), descrever sua história, comportamento perante o personagem (agressivo, amigável, indiferente etc), seus dados pessoais como pontos de vida e outros que forem implementados (pontos de magia, habilidades etc). O comportamento do personagem será estático ou dinâmico. Se dinâmico definir como o mesmo deve ser alterado.

---

## 5. Qualidade de Software <ADALOVE – Desenvolver relatório de resultados do playtest>

### 5.1 Teste de Usabilidade

Ao longo dos encontros vocês tiveram oportunidade de colocar pessoas para testar seu jogo.

Descreva aqui quantas pessoas testaram o jogo, quem são elas e os principais pontos de aprendizado.

**Número de testes:**

**Pontos positivos (observados nos testes em geral):**

**Pontos de melhoria (observados nos testes em geral):**

Número do teste:

Nome e perfil dos tester:

**O que observar e perguntar durante o teste:**

**-Observar e registrar:**

Conseguiu começar o jogo? Entendeu a mecânica do jogo? Aprendeu como jogar?

Conseguiu controlar o jogo?

Progrediu no jogo? Passou de fase? Fez pontos? Chegou ao final? Perdeu rápido?

Entendeu as regras do jogo? Teve dificuldade de compreensão? Teve dificuldade ao jogar?

Foi muito fácil? O jogo foi desafiador?

**-Perguntar a quem testou:**

Numa escala de 0 a 10, quanto você se divertiu nesse jogo?

O que você gostou no jogo?

O que poderia melhorar no jogo?

### 5.2 Métricas de Qualidade

*CARACTERÍSTICAS*	*SUBCARACTERÍSTICAS*	*SIGNIFICADO*	PRODUTO
*Funcionalidade*			
O conjunto de funções satisfazem as necessidades explícitas e implícitas para a finalidade a que se destina o produto?	Adequação	Propõe-se a fazer o que é apropriado?	Por enquanto, as funções básicas já implementadas cumprem com a proposta estabelecida
	Acurácia	Gera resultados corretos ou conforme	O jogador consegue executar todas as funções descritas no código.

		acordados?	
	Interoperabilidade	É capaz de interagir com os sistemas especificados?	O jogador consegue interagir com as máquinas no bunker para realizar os minigames
	Segurança de acesso	Evita o acesso não autorizado, acidental ou deliberado a programas e dados?	Não possui nenhum tipo de segurança contra ações externas
	Conformidade	Está de acordo com normas e convenções previstas em leis e descrições similares?	Sim, o jogo segue todas as normas e leis previstas
*Confiabilidade*	Maturidade	Com que frequência apresenta falhas?	Após efetivado o código, roda normalmente o jogo
O desempenho se mantém ao longo do tempo e em condições estabelecidas?	Tolerância a falhas	Ocorrendo falhas como ele reage?	Como consequência das falhas, o jogo é impedido de iniciar
	Recuperabilidade	É capaz de recuperar dados após uma falha?	Por estar salvo nos computadores de todos componentes do grupo, a recuperação de dados após falhas é mais fácil
*Usabilidade*	Inteligibilidade	É fácil entender os conceitos utilizados?	Está sendo utilizado uma linguagem clara e de fácil compreensão para facilitar o entendimento dos conceitos do jogo
É fácil utilizar o software?	Apreensibilidade	É fácil aprender a usar?	O software possui uma opção de ver os controles utilizados no jogo, o que serve como um tutorial de como usá-lo.
	Operacionalidade	É fácil de operar e controlar a operação?	Os comandos utilizados no jogo são simples e de fácil compreensão
*Eficiência*	Comportamento em relação ao tempo	Qual é o tempo de resposta e de processamento?	Indisponível no momento
Os recursos e os tempos utilizados são compatíveis com o nível de			

desempenho requerido para o produto?			
	Comportamento em relação aos recursos	Quanto recurso utiliza?	Indisponível no momento
*Manutenibilidade*			
Há facilidade para correções, atualizações e alterações?	Analisabilidade	É fácil encontrar uma falha quando ocorre?	Quando ocorre uma falha, o Godot informa qual é o erro. Isso facilita a identificação do local e quais são as falhas, podendo ser corrigidas facilmente.
	Modificabilidade	É fácil modificar e remover defeitos?	Os códigos são divididos de acordo com a funcionalidade. Assim, quando ocorre algum defeito é fácil modificá-lo ou removê-lo, pois é fácil identificar onde está o erro.
	Estabilidade	Há grandes riscos de *bugs *quando se faz alterações?	Não, pois os bugs ficam restritos a cena do objeto em questão
	Testabilidade	É fácil testar quando se faz alterações?	O Godot possui a função de "rodar" a cena, possibilitando o teste após modificações.
*Portabilidade*			
É possível utilizar o produto em diversas plataformas com pequeno esforço de adaptação?	Adaptabilidade	É fácil adaptar a outros ambientes sem aplicar outras ações ou meios além dos fornecidos para esta finalidade no software considerado?	Indisponível no momento
	Capacidade para ser instalado	É fácil instalar em outros ambientes?	Para instalar, é preciso apenas baixar a versão executável localizada no GitHub
	Capacidade para substituir	É fácil substituir por outro software?	Nunca testamos isso
	Conformidade	Está de acordo com padrões ou convenções de portabilidade?	Indisponível no momento

## 6. Relatório - Física e Matemática

### 6.1 Funções

Quais funções são usadas no jogo desenvolvido neste projeto?

Uma função utilizada é a de velocidade em função de speed:  $v(\text{speed}) = v * \text{speed}$ . Ela é usada para determinar a rapidez do personagem.

```
velocityPlayer = velocity.normalized() * speed
```

Outra função é a de velocidade em função de delta:  $v(d) = v * \text{delta}$ . Ela é usada para possibilitar as colisões do personagem com partes do cenário, fazendo com que este pare de se mover quando detectar uma colisão.

```
velocityPlayer = move_and_collide(velocity * delta)
```

Outra função é a de pontuação por tempo:  $\text{pontos}(t) = \text{pontos} * t$ . Ela é usada para calcular a pontuação em cada minigame a partir do tempo que foi levado para resolvê-lo.

```
if Globals.cButton == 3:  
    Globals.pointsCompSci = time * Globals.pointsCompSci
```

Coloque os trechos do programa no Godot onde elas aparecem e explique sua utilidade no jogo.

<[ADALOVE](#) - Definir as funções matemáticas que serão utilizadas no jogo>

### 6.2 Cinemática Unidimensional

A grandeza da cinemática utilizada é a de velocidade. Ela é usada tanto para determinar a rapidez do personagem quanto para possibilitar as colisões do personagem com partes do cenário, fazendo com que este pare de se mover quando detectar uma colisão.

```
velocityPlayer = velocity.normalized() * speed  
velocityPlayer = move_and_collide(velocity * delta)  
pass
```

Quais grandezas da cinemática são usadas no jogo desenvolvido neste projeto?

Coloque os trechos do programa no Godot onde elas aparecem e explique sua utilidade no jogo.

<[ADALOVE](#) - Aplicar os conceitos matemáticos no jogo>

## 6.3 Vetores

Um vetor utilizado é o Vector2. Ele é utilizado para definir a velocidade do personagem, representando posições em espaços 2D.

```
func _physics_process(delta):
```

```
    velocityPlayer = Vector2()
```

Quais vetores são usados no jogo desenvolvido neste projeto?

Coloque os trechos do programa no Godot onde eles aparecem e explique sua utilidade no jogo.

*Obs.: Anexar ao relatório as atividades realizadas em aula (prints, fotos, etc.).*

<[ADALOVE - Definir as funções matemáticas que serão utilizadas no jogo](#)>

## 6.4 Cinemática Bidimensional e mais

As grandezas físicas utilizadas nesse jogo são espaço (posição) e velocidade (ambas já mencionadas na seção 6.1).

O espaço serve para fazer com que o personagem possa se mover no jogo. Position.x serve para movê-lo para a esquerda e para a direita, e position.y serve para movê-lo para cima e para baixo.

A velocidade é usada tanto para determinar a rapidez do personagem quanto para possibilitar as colisões do personagem com partes do cenário, fazendo com que este pare de se mover quando detectar uma colisão.

Quais as grandezas físicas (não trabalhadas nos encontros anteriores) são usadas no jogo desenvolvido neste projeto?

Coloque os trechos do programa no Godot onde elas aparecem e explique sua utilidade no jogo.

<[ADALOVE - Fazer orientação espacial \(2D\)](#)>

---

## 7. Bibliografias

Toda referência citada no texto deverá constar nessa seção, utilizando o padrão de normalização da ABNT). As citações devem ser confiáveis e relevantes para o trabalho. São imprescindíveis as citações dos *sites* de *download* das ferramentas utilizadas, bem como a citação de algum objeto, música, textura ou outros que não tenham sido produzidos pelo grupo, mas utilizados (mesmo no caso de licenças gratuitas, *royalty free* ou similares).



---

## Apêndice

Os apêndices representam informações adicionais que não caberiam no documento exposto acima, mas que são importantes por alguma razão específica do projeto. Em geral, os apêndices do GDD podem incluir os rascunhos das fases, outros *concept arts* do jogo, diagramas diversos etc.