



UNIVERSIDADE NOVE DE JULHO  
BACHARELADO EM CIÊNCIAS DA COMPUTAÇÃO  
PROJETO EM DESENVOLVIMENTO DE SISTEMAS INTERATIVOS

# PROJECT CANDY

**Autores:**

Kayky Lima Guimarães Soares  
Osama Gazy  
Valéria Conceição dos Santos  
Vitória Akemi Corrêa Arakaki

São Paulo,  
Junho 2023

Kayky Lima Guimarães Soares - RA: 2221109568  
Osama Gazy- RA: 3022105838  
Valéria Conceição dos Santos - RA: 422107399  
Vitória Akemi Corrêa Arakaki - RA: 422101990

**PROJETO EM DESENVOLVIMENTO DE SISTEMAS**  
Project Candy - Grupo 13

Trabalho apresentado à Universidade Nove de Julho,  
UNINOVE, em cumprimento parcial às exigências da  
disciplina de Projeto em Desenvolvimento de Sistemas, sob  
orientação do Prof. Thiago Graziani Traue

São Paulo,  
Junho 2023

## **AGRADECIMENTOS**

A todos aqueles que contribuíram, de alguma forma, para a realização deste trabalho. Ao professor Thiago Graziani Traue, por ter sido nosso orientador e ter desempenhado tal função com dedicação e amizade, pela ajuda e pela paciência com a qual guiaram o nosso aprendizado.

"Nossa tecnologia passou a frente de nosso entendimento, e a nossa inteligência desenvolveu-se mais do que a nossa sabedoria." Roger Revelle

## **RESUMO**

O projeto consiste em um jogo, onde o usuário poderá realizar missões e tarefas para conseguir fazer novas atualizações em sua cafeteria. É um jogo simples, tranquilo de se jogar, que tem o intuito de relaxar o jogador com os pequenos desafios propostos para o personagem atualizar sua cafeteria. É um jogo em terceira pessoa, baseado em uma aventura da personagem principal, uma mulher, que viajará entre duas áreas chamadas de zonas e dentro dessas áreas haverá desafios para a personagem principal obter os itens para sua cafeteria, que entre eles estão os artigos de decoração, seus ingredientes para sua cozinha. A cada missão ganha a recompensa vai direto para inventário de sua loja e se a missão fracassar, ou seja, a personagem morrer, ela volta para o começo do jogo e suas recompensas anteriores não serão perdidas. O objetivo do jogo é a finalização da sua tão sonhada cafeteria, e para conseguir realizar seu objetivo, terá que derrotar o Boss final do último reino.

## **ABSTRACT**

The project consists of a game, where the user will be able to carry out missions and tasks to be able to make new updates in his cafeteria. It is a simple game, easy to play, which aims to relax the player with the small challenges proposed for the character to update his cafeteria. It is a third-person game, based on an adventure of the main character, a woman, who will travel between two areas called zones and within these areas there will be challenges for the main character to obtain the items for her cafeteria, which among them are the articles of decoration, your ingredients for your kitchen. With each mission you win, the reward goes straight to your store's inventory and if the mission fails, that is, the character dies, it returns to the beginning of the game and your previous rewards will not be lost. The objective of the game is to complete your dreamed coffee shop, and to achieve your objective, you will have to defeat the final Boss of the last kingdom.

## SUMÁRIO

---

<b>1.</b>	<b>Introdução.....</b>	<b>7</b>
1.1.	Categoria.....	7
1.2.	Conceito.....	7
1.3.	Descrição.....	8
1.4.	Plataforma.....	8
1.5.	Análise Comparativa.....	8
1.6.	Público Alvo.....	8
<b>2.</b>	<b>Desenho do Processo de Negócio.....</b>	<b>9</b>
<b>3.</b>	<b>Game Design.....</b>	<b>9</b>
3.1.	Sinopse.....	9
3.2.	Cenário.....	10
3.3.	Mecânica.....	10
3.4.	Arte Conceitual.....	10
3.5.	Personagem.....	11
3.5.1.	Características.....	11
3.5.2.	Personalidade.....	11
3.5.3.	História.....	11
3.5.4.	Habilidade.....	11
3.5.5.	Ilustração Visual.....	11
3.6.	Inimigos.....	12
3.7.	Recompensas.....	13
<b>4.</b>	<b>Produção.....</b>	<b>14</b>
4.1.	Progresso.....	14
4.2.	Game Over.....	14
4.3.	Análise SWOT.....	14
<b>5.</b>	<b>Requisitos.....</b>	<b>15</b>
5.1.	Funcionabilidade.....	15
5.1.1.	Segurança.....	15
5.1.2.	Interoperabilidade.....	15
5.1.3.	Adequação.....	15
5.2.	Usabilidade.....	15
5.2.1.	Prevenção de Erros.....	15
5.2.2.	Estética e Design.....	15
5.2.3.	Apreensibilidade.....	16
5.2.4.	Documentação.....	16
5.2.5.	Suporte.....	16
<b>6.</b>	<b>Desenvolvimento Orientado a Objetos.....</b>	<b>16</b>
6.1.	Tela do sistema.....	16
6.2.	Banco de Dados.....	17
6.3.	Código do Sistema.....	18
<b>7.</b>	<b>Conclusão.....</b>	<b>21</b>
<b>8.</b>	<b>Link do Repertório.....</b>	<b>22</b>
<b>9.</b>	<b>Referências.....</b>	<b>22</b>

## 1. Introdução

Em todo o mundo, já somamos mais de 2,1 bilhões de jogadores de jogos eletrônicos. Quase um terço dos habitantes atuais deste grande mundo azul já viajou para terras alternativas ou até mesmo outro planeta. Até outros universos foram explorados nessa imensidão de realidades paralelas proporcionadas a nós pelos videogames ao longo de seus mais de 50 anos de existência.

Inspirados e motivados por essas aventuras, muitos jogadores almejam criar seu próprio jogo, fazendo com que o universo de sua própria criação seja explorado e habitado por outros milhares de jogadores.

Porém, o que muitos desconhecem é que criar um jogo não é trivial. Longe disso, é uma tarefa que exige uma variedade de conhecimentos diferentes, muitas e muitas horas de estudo, dedicação, criatividade e perseverança. Muitos desenvolvedores que hoje são mundialmente conhecidos produziram jogos pouco famosos, que venderam muito abaixo do esperado.

Além disso, o trabalho em equipe é fundamental, dificilmente hoje em dia um jogo é feito por uma única pessoa. Mesmo no setor independente, times costumam ser compostos de 3, 5 ou até 20 pessoas. Já as equipes profissionais costumam contar com centenas de profissionais, por exemplo a RockStar Game, empresa muito conhecida pela série Grand Theft Auto, mais conhecido como GTA, conta com mais de 300 profissionais para produzir os jogos desta série.

### 1.1 Categoria

- Ação e Aventura;
- Roguelike;
- Slice of life.

### 1.2 Conceito

As famílias ficam, em geral, preocupadas com jovens focados em games, e o impacto na saúde mental. No entanto, tem crescido também a possibilidade de que os jogos possam fazer bem. “Como todo bom entretenimento, o videogame pode promover bem-estar. Esquecemos um pouco da nossa realidade, entramos em um outro mundo quando estamos jogando. Além disso, pode promover relação social, trazer um relaxamento das tensões e ajudar a deixar a pessoa mais leve”, afirma Marina Noronha, mestra em psicologia do desenvolvimento humano.

Em um estudo realizado pela Universidade de Oxford comprovou que os jogos eletrônicos podem ser um grande aliado de quem sofre com transtornos mentais, como depressão e ansiedade. De acordo com o estudo realizado pela faculdade britânica, pessoas que jogam por até quatro horas por dia notam que se sentem mais alegres. Os jogos eletrônicos também tiveram um papel importante na luta contra o tédio e a ansiedade de algumas pessoas durante o período de

isolamento social.

No contexto apresentado, uma maneira que pode auxiliar no combate às doenças psicológicas, são os estímulos e exercícios mentais. E é aí que os jogos aparecem como um grande aliado ao combate contra os sintomas de depressão e ansiedade. Devemos deixar bem claro, que pacientes com transtornos mentais não devem trocar seus remédios e ir ao psicólogo pelos jogos, apenas os colocar em seu cotidiano como um passatempo e uma forma de se desestressar.

### **1.3 Descrição**

É um jogo em terceira pessoa, baseado em uma aventura da personagem principal, uma mulher, que viajará entre três áreas chamadas de zonas e dentro dessas áreas haverá subáreas com missões e desafios para a personagem principal obter os itens para sua cafeteria, que entre eles estão os artigos de decoração, seus ingredientes para sua cozinha. A cada missão ganha a recompensa vai direto para inventário de sua loja e se a missão fracassar, ou seja, a personagem morrer, ela volta para o começo do jogo e suas recompensas anteriores não serão perdidas. O objetivo do jogo é a finalização da sua tão sonhada cafeteria, e para conseguir realizar seu objetivo, terá que derrotar o Boss final do último reino.

### **1.4 Plataforma**

Este jogo será desenvolvido para computadores, com sistema operacional Windows e exigirá uma disponibilidade de tempo correspondente ao que o usuário tiver disponível, podendo intercalar durante o período noturno, diurno e matutino.

### **1.5 Análise Comparativa**

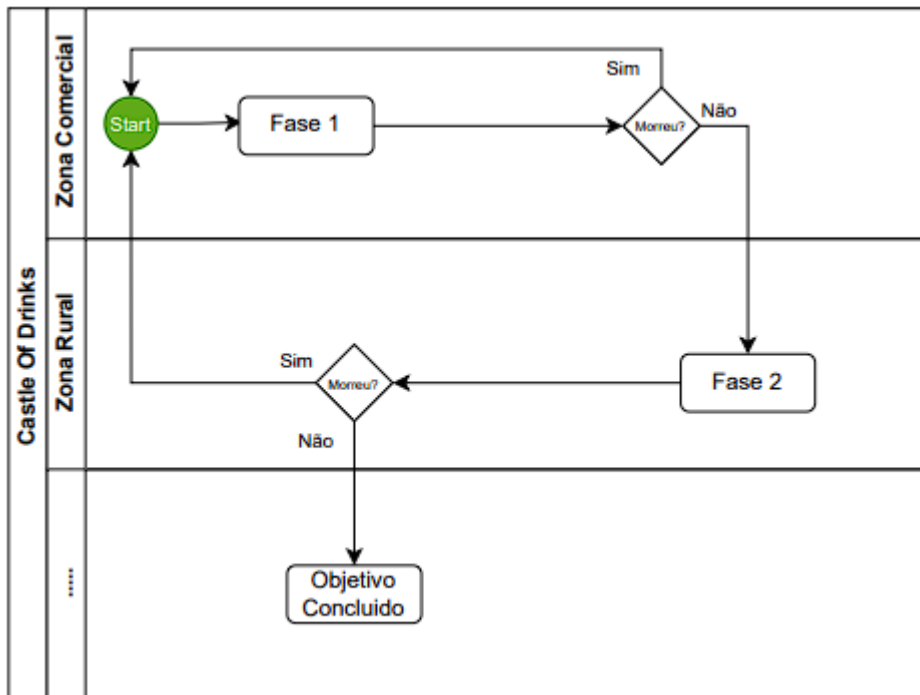
- **Animal Crossing:** 3D, interações entre personagens e estilo chibi;
- **The Binding of Isaac: Rebirth:** 2D e focado em luta.
- **Dead Cells:** 2.5D, pixel-art e Hack and slash.

### **1.6 Público alvo**

Os principais stakeholders, público alvo, do jogo são jovens acima de 14 anos e adultos, pois o jogo pode ser considerado um tanto violento e inapropriado para crianças e pré-adolescentes, visando isso, é recomendado para maiores de 14 anos.



## 2. Desenho de Processo de Negócio



## 3. Game Design

### 3.1 Sinopse

É um jogo de gênero Roguelike, que conta a história de Opal que se muda para uma pequena cidade afastada da capital, onde ela compra um pequeno comércio, que no futuro irá se tornar sua confeitaria dos sonhos, só que ela não imaginava que precisaria passar por diversos desafios para conseguir conquistar suas coisas.

### 3.2 Cenário

O jogo se passa na cafeteria, onde a personagem principal irá fazer seus updates no cenário de sua cafeteria. Os minijogos irão se passar em quatro mundos, subdivididos em Zona Comercial e Zona Rural, e cada mundo terá seu inimigo de acordo com sua temática.



### 3.3 Mecânica

Tendo como propósito principal derrotar seus concorrentes/inimigos, o jogador precisa executar batalhas e ganhar para que possa aumentar os índices de sobrevivência da personagem com as conquistas dadas na vitória de cada fase.

As tarefas incluem bater e atirar nos inimigos, uma mecânica fácil e de conhecimento de todos:

- W : anda para frente;
- S : anda para trás;
- A : anda para esquerda;
- D : anda para direita;
- Espaço ou Space: ataque;

### 3.4 Arte Conceitual

O *Design flat*, minimalista, bastante colorido, com cores pastéis, com degradês e sem volumes muito evidentes. Os cenários seguem a padronagem dos personagens mas sem destacar mais que os personagens ou os objetos de interação.

## **3.5 Personagem**

### **3.5.1 Características**

Tipo: humana;

Aparência física: personagem feminina, com cabelos castanhos e duas marias chiquinhas, blusa branca e saia rosa e azul;

### **3.5.2 Personalidade**

Opal é uma mulher de personalidade extremamente amigável, que consegue sempre ver o melhor das pessoas e está sempre tentando se superar.

### **3.5.3 História**

Opal é uma mulher que sempre sonhou para ter sua cafeteria desde pequena, se mudou para uma pequena cidade afastada da capital, onde ela compra um pequeno comércio, que precisa de muitas reformas, onde no futuro irá se tornar sua confeitaria dos sonhos, só que ela não imaginava que precisaria passar por diversos desafios para conseguir conquistar suas coisas, mas apesar desses desafios, ela acaba criando uma amizade com seus concorrentes.

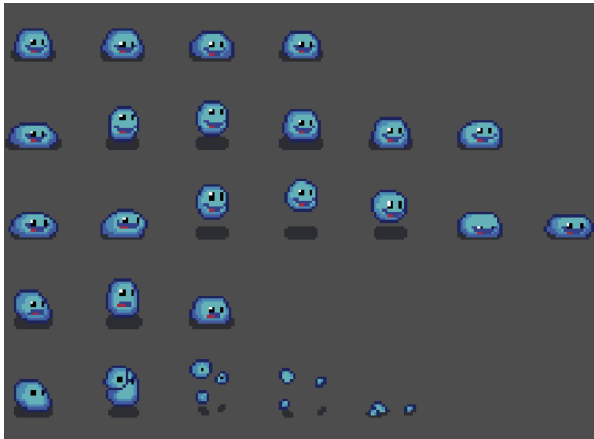
### **3.5.4 Habilidade**

Sua principal habilidade é a criação de comidas e bebidas para melhoria de sua cafeteria na jogatina.

### **3.5.5 Ilustração visual**



### 3.6 Inimigos



**Nome:** gota

**Tipo:** slime

**Aparência física:** bolinha da cor azul, com olhinhos e boquinha.

**Habilidades:** saltar;

**Dano:** dano é encostando na personagem.

**Zona:** comercial.

**Conquista:** moedas e slime



**Nome:** Pom-Pom

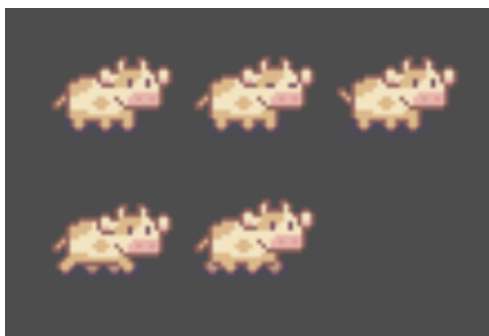
**Tipo:** coelho

**Aparência física:** coelha branca, com orelhas rosa, com uma coleira dourada.

**Habilidade:** seu dano é empurrando;

**Zona:** Rural

**Conquista:** moedas e cenoura



**Nome:** Mu-Mu

**Tipo:** vaca

**Aparência física:** vaca da cor bege, com manchas marrons e nariz rosa;

**Habilidade:** cabeçada

**Zona:** rural

**Conquista:** moedas e leite.

### 3.7 Recompensas



**Nome:** coins

**Aparência física:** objeto redondo da cor dourada

futuras compras.

**Uso:** usada como recompensa e para



**Nome:** cenoura

**Aparência física:** objeto semelhante a uma cenoura e da cor dourada

**Uso:** recompensa ganha na derrota de um inimigo e utilizada como recurso para sua cafeteria.



**Nome:** slime

**Aparência física:** objeto redondo, como degrade de diferentes tons de roxo

**Uso:** recompensa ganha na derrota de um inimigo e utilizada como recurso para sua cafeteria.



**Nome:** Milk

**Aparência física:** objeto oval, semelhante a um copo, transparente, com um conteúdo da cor laranja e bolinhas na cor marrom, contendo uma tampa da cor bege e canudo roxo.

**Uso:** recompensa ganha na derrota de um inimigo e utilizada como recurso para sua cafeteria.

## 4. Produção

### 4.1 Ambientes

**Mundo 1** - Zona comercial, com ruas e estruturas de uma cidade;

- Boss 1 - Gota;

**Mundo 2** - Zona dos Rural, temática mais alegre, com ruas e infraestrutura cheia de cores e um aspecto mais fofo, com vários inimigos atirando;

- **Boss 2** - Pom-Pom e Mu-Mu

### 4.2 Progresso

Como o jogo progride?

O jogo progride de acordo com a passagem de cada fase, em cada level, o jogador irá ganhando conquistas para o update de sua cafeteria e recompensas para a personagem. O jogo termina quando a personagem principal conseguir vencer o último boss e ganhando sua última conquista para sua tão sonhada cafeteria ficar completa.

### 4.3 Game Over

Como ocorre? E quando ocorre?

O game over ocorre quando a personagem principal perde a batalha contra seu inimigo ou contra vários inimigos, em algum das zonas, na sua derrota, a personagem volta para a base, começando o jogo novamente, não importando em qual fase ela se encontra, porém suas conquistas anteriores e suas recompensas não serão perdidas.

### 4.4 Análise de SWOT (Strengths, Weakness, Opportunities, Threats)

- **Pontos fortes:** Ótimo passatempo, humor e piadas, tem como principal objetivo é entreter o usuário, gratuidade e facilidade de acesso.
- **Pontos fracos:** equipe reduzida e pouco tempo para criação.
- **Oportunidades:** As oportunidades que o jogo traria para os desenvolvedores seria um bom portfólio e também serviria de referência de capacidade. E as oportunidade para usuário são uma ótima forma de entretenimento, auxiliar no raciocínio lógico, tomada rápida de decisões e uma ótima forma de relaxar por não ser um jogo repetitivo.
- **Ameaças:** Os principais riscos que podem conter no jogo são Copyright, por causa dos direitos autorais, bugs no códigos, mas que podem ser resolvidos ao longo do tempo, animação lenta e a queda de FPS (Frames per Second).

## **5. Requisitos**

### **5.1 Funcionalidade**

A funcionalidade de um software diz respeito à satisfação de necessidades que deram origem ao projeto. Abrange requisitos implícitos e explícitos e está intimamente ligada à qualidade do código criado e estão divididos nos seguintes tópicos:

#### **5.1.1 Segurança**

O jogo vai ser somente acessível para usuários de Windows, usuários de iOS ainda não vão poder ter acesso, por causa da licença exigida pela Apple para poder fazer a disponibilidade para seus usuários. Para o começo, o jogo vai ser vulnerável a ameaças, porém com constantes atualizações para esse tipo de problema não ocorrer.

#### **5.1.2 Interoperabilidade**

O jogo será de alta resolução, com gráficos todos em 2D, só não irá suportar a interação com outros sistemas operacionais a não ser o Windows.

#### **5.1.3 Adequação**

O produto está de acordo com os objetivos principais que se originou, provavelmente contará com mais coisas que das ideias originais e algumas mudanças durante o percurso de programação do jogo, mas ele está de acordo com as ideias originais.

### **5.2 Usabilidade**

O software precisa ser de fácil utilização, principalmente porque a grande maioria dos usuários não entende de linguagem de programação. Possui diversas subcategorias, entre elas:

#### **5.2.1 Prevenção de erros**

Verificação maior em relação aos códigos, distância favorável entre os objetos para não haver bugs e prevenção contra a queda de FPS.

#### **5.2.2 Estética e Design**

Estética “fofa” por causa dos seus tons em cores pastéis, com objetos minimalistas, com personagem animal e inimigos feitos com objetos de acordo com a temática de cada mundo.

### 5.2.3 Apreensibilidade

O jogo contará com tutoriais iniciais e uma mecânica de fácil compreensão, sem que o usuário precise de muito esforço ou tempo para aprender o funcionamento do jogo.

### 5.2.4 Documentação

Documentos bem estipulados com etapas, se baseiam em uma ótima organização do que a gente já fez, o que já foi feito e o que será feito.

### 5.2.5 Suporte

Conterá com suporte ao cliente por meio de comunicação entre o usuário e os desenvolvedores através de redes sociais e email.

## 6. Desenvolvimento Orientado a Objetos

### 6.1 Tela do Sistema

- Tela de início:



- Ambientes do jogo:





- Tela de Game Over:



## 6.2 Banco de Dados

```

1 • CREATE DATABASE INVENTARIO;
2 • USE INVENTARIO;
3
4 • CREATE TABLE OPAL(
5     opal int primary key,
6     recompensas int
7 );
8
9 • CREATE TABLE SLIME(
10    slime int primary key,
11    coinsR int,
12    slimer int,
13    opal int,
14    foreign key(opal) references OPAL(opal)
15 );
16
17 • CREATE TABLE MUMU(
18    mumu int primary key,
19    coinsR int,
20    milk int,
21    opal int,
22    foreign key(opal) references OPAL(opal)
23 );
24
25 • CREATE TABLE POMPOM(
26    pompom int primary key,
27    coinsR int,
28    opal int,
29    foreign key(opal) references OPAL(opal)
30 );
31
32 • CREATE TABLE ARMAZEM(
33    armazem int primary key,
34    opal int,
35    slime int,
36    mumu int,
37    pompom int,
38    foreign key(opal) references OPAL(opal),
39    foreign key(slime) references SLIME(slime),
40    foreign key(mumu) references MUMU(mumu),
41    foreign key(pompom) references POMPOM(pompom)
42 );
43

```

## 6.3 Código do Sistema

- **Script Opal:**

```
extends KinematicBody2D
```

```
onready var collision: CollisionShape2D = get_node("AttackArea/Collision")
```

```
onready var animation: AnimationPlayer = get_node("animation")
```

```
onready var sprite: Sprite = get_node("Sprite")
```

```
var velocity: Vector2
```

```
var can_die: bool = false
```

```
var can_attack: bool = false
```

```
export (int) var speed
```

```
func _physics_process(_delta: float) -> void:
```

```
    move()
```

```
    attack()
```

```
    animate()
```

```
    direction()
```

```
func move() -> void:
```

```
    var direction_vector: Vector2 = Vector2(
```

```
        Input.get_action_strength("d") - Input.get_action_strength("a"),
```

```
        Input.get_action_strength("s") - Input.get_action_strength("w")
```

```
    ).normalized()
```

```
    velocity = direction_vector * speed
```

```
    velocity = move_and_slide(velocity)
```

```
func attack() -> void:
```

```
    if Input.is_action_just_pressed("attack") and not can_attack:
```

```
        can_attack = true
```

```
func animate() -> void:
```

```
    if can_die:
```

```
        animation.play("dead")
```

```
        set_physics_process(false)
```

```
    elif can_attack:
```

```
        animation.play("attack")
```

```
        set_physics_process(false)
```

```
    elif velocity != Vector2.ZERO:
```

```
        animation.play("run")
```

```
    else:
```

```
        animation.play("idle")
```

```
func direction() -> void:
```

```
    if velocity.x > 0:
```

```
        sprite.flip_h = false
```

```
        collision.position = Vector2(20, 8)
```

```
    else:
```

```
        sprite.flip_h = true
```

```
        collision.position = Vector2(-20, 8)
```

```

func kill() -> void:
    can_die = true
    get_tree().change_scene("res://scenes/game over.tscn")

func on_animation_finished(anim_name):
    if anim_name == "dead":
        var _reload: bool = get_tree().reload_current_scene()
    elif anim_name == "attack":
        can_attack = false
        set_physics_process(true)

```

### • **Script Inimigos:**

extends KinematicBody2D

```

onready var animation: AnimationPlayer = get_node("Animation")
onready var sprite: Sprite = get_node("Sprite")

```

```

var player_ref = null
var velocity: Vector2
var can_die: bool = false

```

```

export(int) var speed

```

```

func _physics_process(_delta: float) -> void:
    move()
    animate()
    verify_direction()

```

```

func move() -> void:
    if player_ref != null:
        var distance: Vector2 = player_ref.global_position - global_position
        var direction: Vector2 = distance.normalized()
        var distance_lenght: float = distance.length()
        if distance_lenght <= 5:
            player_ref.kill()
            velocity = Vector2.ZERO
        else:
            velocity = speed * direction
    else:
        velocity = Vector2.ZERO

    velocity = move_and_slide(velocity)

```

```

func animate() -> void:
    if can_die:
        animation.play("die")
        set_physics_process(false)
    elif velocity != Vector2.ZERO:
        animation.play("walk")
    else:
        animation.play("idle")

```

```

func verify_direction() -> void:
    if velocity.x > 0:
        sprite.flip_h = false
    elif velocity.x < 0:
        sprite.flip_h = true

func on_body_entered(body):
    if body.is_in_group("player"):
        player_ref = body

func on_body_exited(body):
    if body.is_in_group("player"):
        player_ref = null

func kill(area):
    if area.is_in_group("player_attack"):
        can_die = true

func on_animation_finished(anim_name):
    if anim_name == "die":
        queue_free()

```

- **Script Tela Game Over:**

```

extends Node2D

func _on_reset_pressed():
    get_tree().change_scene("res://scenes/world.tscn")

func _on_exit_pressed():
    get_tree().change_scene("res://scenes/menu.tscn")

```

- **Script Tela de Menu:**

```

extends Node2D

var velocity: Vector2
onready var animation: AnimationPlayer = get_node("animation")

func _on_play_pressed():
    get_tree().change_scene("res://scenes/world.tscn")

func _on_options_pressed():
    pass # Replace with function body.

func _on_exit_pressed():
    get_tree()

```

## **7. Conclusão**

Neste projeto foi desenvolvido um jogo simples, de maneira que o jogador pudesse facilmente jogar sem estresse, pois os jogos em geral, além de serem um meio de lazer, alguns também auxiliam no combate a doenças psicológicas, pois são estímulos e exercícios mentais, pessoas que jogam por até quatro horas por dia notam que se sentem mais alegres, mas devemos deixar bem claro, que pacientes com transtornos mentais não devem trocar seus remédios e ir ao psicólogo pelos jogos, apenas os colocar em seu cotidiano como um passatempo e uma forma de se desestressar.

A estrutura do jogo é fixa, possui um número fixo de ambientes, objetos não interativos e interativos. O Castle Of Drinks é um jogo em 2D, do tipo ação e aventura. Neste jogo o jogador caminha pelos dois ambientes com um único objetivo, matar seus inimigos, para assim conseguir sua tão sonhada cafeteria dos sonhos.

## 8. Link do Repertório

**GitHub:** [https://github.com/vitoriaarakaki/Project\\_Candy](https://github.com/vitoriaarakaki/Project_Candy)

**Drive:**

<https://drive.google.com/drive/folders/1vdk5FWivmDt4b51ApHC4BQifkiD86THx>

## 9. Referências

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. ABNT NBR 10719: Informação e documentação — Relatório técnico e/ou científico — Apresentação. 4ª edição. Rio de Janeiro, 11 p. 2015.

JOGOS E SAÚDE MENTAL – Escrito por André Araújo e Rodrigo Fontão – 09 de Setembro de 2022.

O QUE É UM BOM SOFTWARE? - nata.house –

<https://natahouse.com/pt/o-que-e-um-bom-software-identifique-essas-5-caracteristicas>

LEVEL UP: UM GUIA PARA O DESIGN DE GRANDES JOGOS – Edição Português por Scott Rogers (Autor). Blucher; 1ª edição (1 janeiro 2013).

INTRODUÇÃO A JOGOS DIGITAIS - <https://central3.to.gov.br/arquivo/453377/>