

# Zombie Park

## *Documento de Design de Jogo*

---

Autor: Kairo Sumida

2018

## Index

---

1. [Zombie Park](#)
2. [Index](#)
3. [Design do jogo](#)
4. [Técnico](#)
5. [Design de nível](#)
6. [Desenvolvimento](#)
7. [Sounds/Musics](#)
8. [Personagem](#)
9. [Tiles](#)
10. [Background](#)
11. [Ranking](#)
12. [Problemas](#)
13. [Filosofia](#)
14. [Editável](#)

## Sumário

Jogo contínuo de rápida transição entre usuários e com ranks

## Jogabilidade

O jogador é sempre obrigado a mover-se para frente, e possui um tempo para curar-se, caso não se cure, automaticamente perde-se o jogo.

O jogador deve passar por obstáculos dentro do jogo e ir o mais longe possível.

## Mentalidade

O jogo deve proporcionar ao jogador o espírito de competitividade e o conforto para jogar um jogo fácil, porém com algum desafio.

## Telas

1. Tela de logo

2. Game

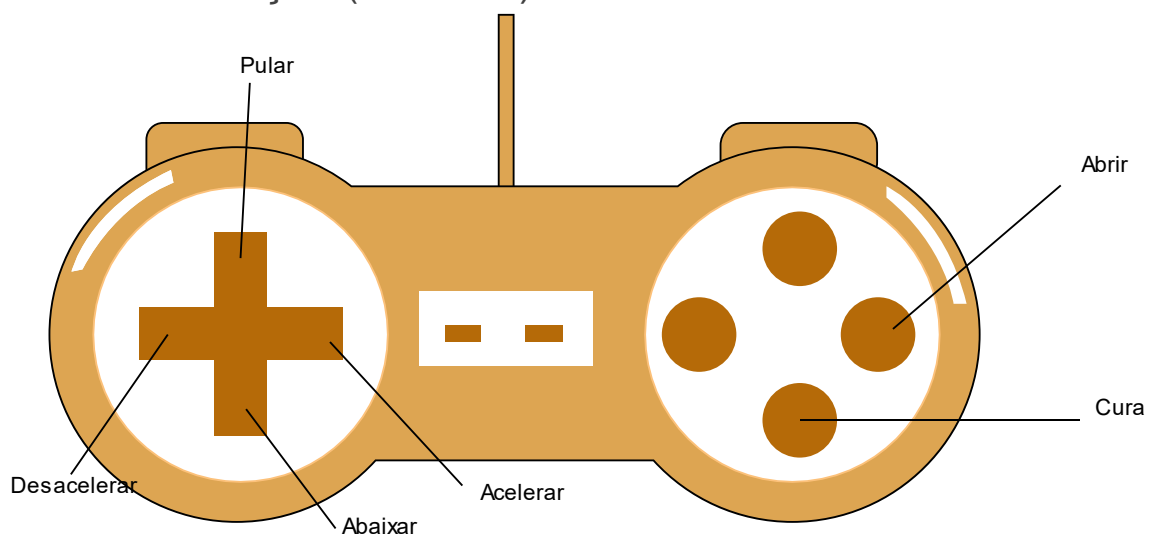
3. Ranking

Obs: Tela de logo é obrigatória pois o jogo foi desenvolvido com o plano gratuito da unity3D(Free). Automaticamente o jogo inicializa na tela de rank, ao pressionar qualquer tecla o jogo inicializa

## Controles

O jogador poderá usar o mouse ou o controle

- Pular seta para cima
- Abaixar seta para baixo
- Diminuir velocidade Seta para trás
- Aumentar velocidade Seta para frente
- Ação Tecla E(Teclado), Botão extremo direito(Controle)
- Usar cura Tecla R(Teclado), Botão a baixo da ação (Controle)



## Mecânicas

A cada 30 segundos o personagem vai se tornando vermelho, indica que o personagem está perdendo o controle e precisa ser curado. Para se curar o jogador precisa pegar a cura e apertar o botão de cura. O jogador também pode perder caso caia de uma altura muito alta, ou se cair em alguma armadilha



# Design de nível

---

- Fase1. Deserto
  - i. Dia
  - ii. Plano
  - iii. Obstáculos
- Fase2. Random1
  - i. Subida
  - ii. Plano
  - iii. Dia
  - iv. Escuro
  - v. Obstáculos
- Fase3. Caverna
  - i. Descida
  - ii. Escuro
  - iii. Armadilhas
  - iv. Cobertura
- Fase4. Random2
  - i. Tudo do Random1
  - ii. Descida
- Fase5. Escalada
  - i. Subida
  - ii. Escuro
- Fase6. Random3
  - i. Tudo do Random2
  - ii. Buracos
- Fase7. Random4
  - i. Random3
  - ii. Portas
  - iii. Armadilhas
- Fase8. Cidade
  - i. Portas
  - ii. Áreas não visíveis
- Fase9. Random5
  - i. Random4
  - ii. Áreas não visíveis

Explicação de cada elemento das fases.

**Dia:** Significa que estará claro para o jogador, ou seja representará o dia.

**Plano:** Indica que o eixo x de construção do chão seguirá sempre o mesmo eixo y ou seja n existe descidas ou subidas, porem pode existir alguns blocos no chão que representa os obstáculos

**Obstáculos:** Representa alguns blocos no chão inofensivos, porem que pode atrapalhar o jogador, forçando-o a pular para continuar avançando.

**Subida:** Indica que o eixo y sofrerá variação positiva, forçando o jogador estar sempre pulando para continuar avançando.

**Escuro:** Significa que estará escuro para o jogador, representando a noite.

**Descida:** Indica que o eixo y sofrerá variação negativa, o jogador deve tomar cuidado para não cair de uma altura muito alta.

**Armadilhas:** Representará certos Triggers(ações ativas quando o jogador passa sobre a área) que forçará o jogador a esquivar de alguma coisa, ou do próprio trigger para não ativar a armadilha

**Cobertura:** Todo o percurso possui um teto, representando como se fosse uma caverna.

**Buracos:** Não possuem um fim, porem ao cair em um, o jogador automaticamente perde o jogo

**Portas:** São obstáculos, porem que precisam ser ativos por um botão do controle para abrir.

**Áreas não visíveis:** Ocorre dentro de casas, quando o jogador entra dentro de uma casa, não é possível enxergar, porem o personagem continua andando, a menos que seja bloqueado por um obstáculo. Pode também ter poções de cura nas áreas não visíveis

Design de nível significa que o jogador iniciará no deserto

O deserto será criado randomicamente com alguns obstáculos para o jogador pular. Não oferecendo nenhuma dificuldade

Depois ele irá para uma área randômica, significa que o ambiente será sorteado, primeiro ele sorteará se estará de dia ou noite. Digamos que foi sorteado noite. Então no jogo estará noite. Agora o jogo sorteará se será um terreno plano ou subida. Digamos que foi sorteado subida, então ele sorteará o tamanho da subida. Obstáculos apenas acontecem no plano, caso seja sorteado plano, ele então sorteia novamente se terá um obstáculo no plano ou não

Caverna será o segundo nível do jogo acrescentando a descida ao jogador e algumas armadilhas

Escalada, acrescenta a perseguição, como se tivesse algum pássaro seguindo-o, por isso o jogador não pode ficar parando ou andando na velocidade normal, ele deverá correr.

Cidade, apesar de não ter muitos problemas com buracos ou perseguições, a cidade pode ser um problema grave, visto que ao entrar em uma casa, pode existir uma mobília atrapalhando o movimento, não permitindo que o jogador se mova. Fazendo perder tempo, e eventualmente perdendo o controle, por falta de cura. A cura também pode estar escondida dentro de uma casa.

## Desenvolvimento

---

UIManager – Classe responsável por controlar todos os Canvas de interface do usuário.

GameManager – Classe que irá monitorar todo o estado do jogo

PlayerController – Responsável por realizar as ações do jogador segundo a tecla pressionada

PlayerAnimation – Realiza a tarefa animada requisitada pela PlayerController

TileConstructor – Responsável por construir o mapa em tempo de execução

PaiConstructor- Classe pai de todos os construtores exceto do TileConstructor. Deverá implementar um método de construção que irá retornar para o TileConstructor, a posição Vector do último tile Construído.

Herdará da classe pai

- SubidaConstructor
- DescidaConstructor
- CavernaConstructor
- CidadeConstructor
- EscaladaConstructor
- RetaConstructor

DaySelector – Responsavel por selecionar dia ou noite no jogo

SelectBackGround – Responsavel por selecionar o background do jogo

JsonData – Responsável por salvar e ler os dados no formato Json

JsonEncript – Responsável por criptografar os dados no formato Json, para que não seja possível alterar o rank manualmente.

## Musica/Som

---

### 1. Possível musica em loop

<https://freesound.org/people/Greek555/sounds/444885/>

### 2. Possivel som de pulo

Som deve ser cortado e usado quando o personagem tocar o chão ao pular ou cair

<https://freesound.org/people/CUeckermann/sounds/444823/>

Musicas ainda não foram selecionadas

# Personagem

---

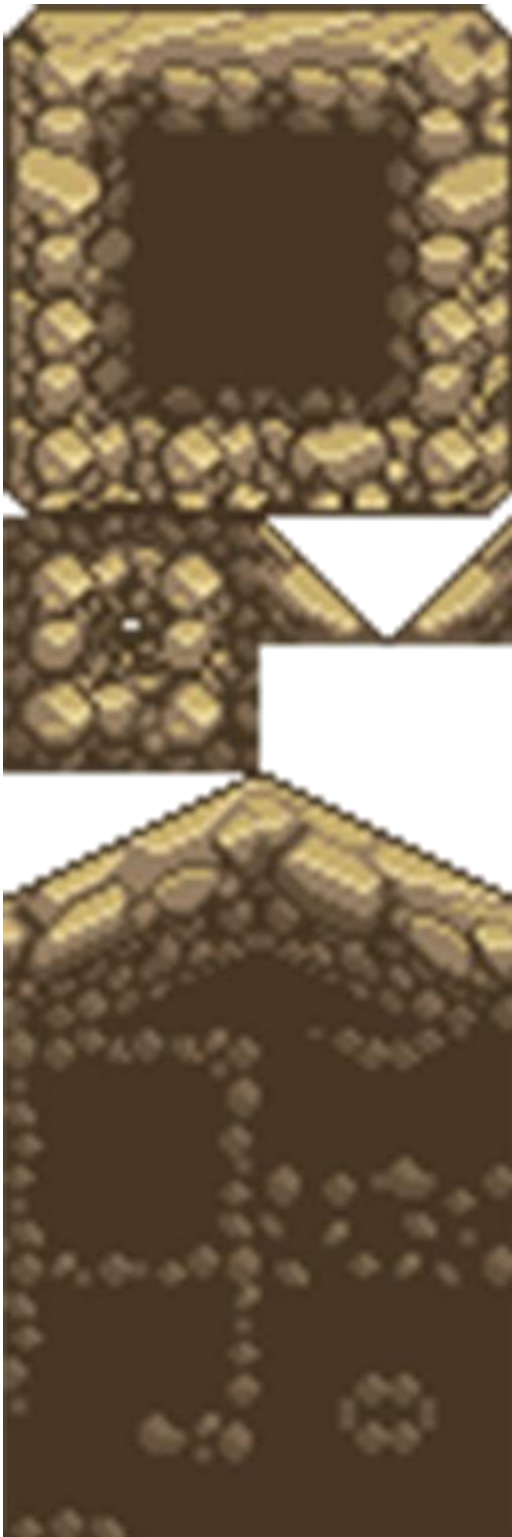


Mover.pptx

## Tiles

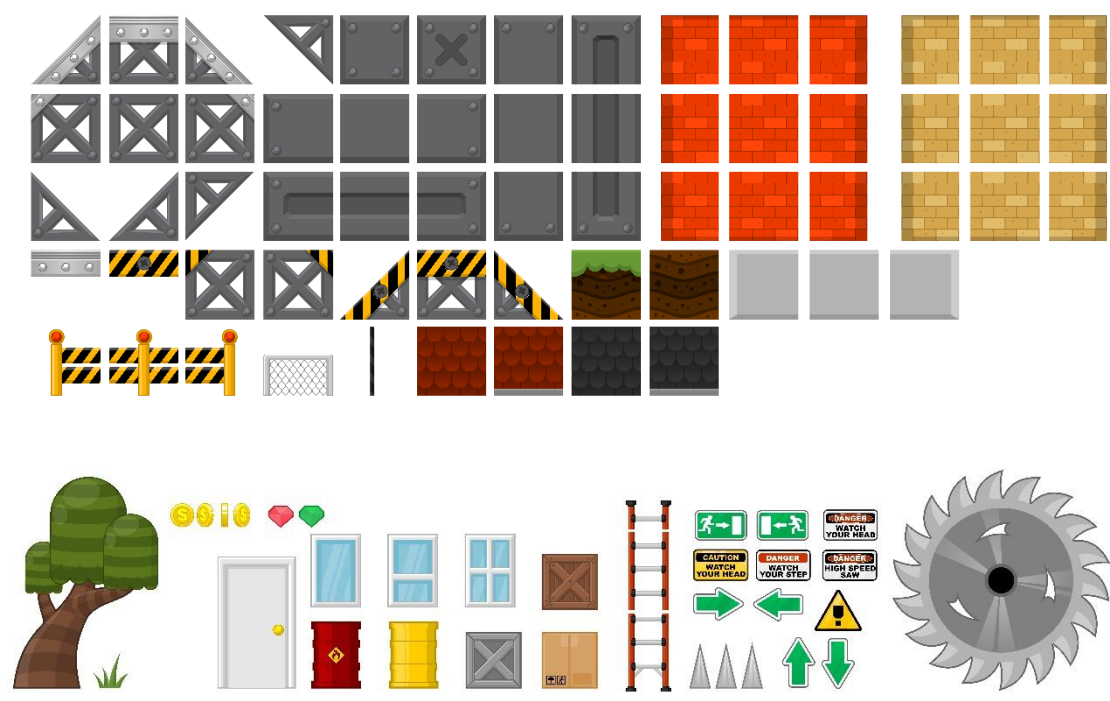
---

### Deserto



Cidade

Cidade



# Escalada

Não definido

# Caverna

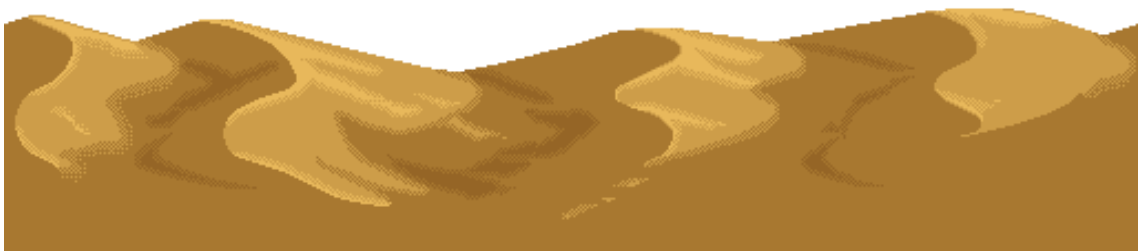
Não definido



## Background

---

### Deserto



### Cidade



**Escalada – Não definido**

**Caverna – Não definido**



## Ranking

---

O ranking é armazenado na máquina, em um arquivo no formato Json

Esse arquivo é codificado para que uma pessoa não possa alterar manualmente

O computador pode ser desligado, que o ranking continuará armazenado

O ranking funcionará da seguinte forma, dependendo da distância percorrida, ao finalizar o jogo se o jogador conseguiu um placar maior que um dos 10 jogadores, o jogo então pausa e pede para o jogador digitar o nome, ele pode cancelar, e então será salvo em branco, caso a pessoa consiga bater o top 1, 2 ou 3 deverá aparecer uma mensagem mais destacada ou um troféu ou um título.

Caso o parque queira oferecer algo para o top 1 2 ou 3, fica a critério.

Na tela de rank será possível ver a data, a pontuação e o nome do jogador.

Para manter a data em funcionamento correto é necessário que o relógio do computador esteja correto



## Problemas

---

1. O nome do jogador não pode ser perguntado ao iniciar o jogo, devido a dificuldade para digitar o nome por um controle, visto que um controle não possui letras.

Solução. Foi colocado no final, exatamente para digitar apenas as pessoas que terão os nomes armazenados. E filtrar também pessoas que não saiba escrever. (Para escrever o nome a pessoa deverá rolar a setinha para cima até encontrar a letra desejada, passando para o lado e escolhendo a outra e assim por diante)

2. Para evitar que o jogador memorize os locais, foi tudo randomizado, tornando o jogo mais complexo tanto ao jogar como desenvolver. Não é possível criar objetos estáticos(predefinidos) pois foge do escopo do projeto.

Com a ideia da randomização foi possível tornar o jogo infinito, garantindo que o jogo nunca chegue ao fim.

3. O jogador pode sair do computador, entrar um novo jogador e continuar o jogo

Solução. Para evitar isso foi criado o sistema de cura, onde a cada 30seg com o zumbi inativo, automaticamente o jogador perde.

4. Falta de opções

Não existem opções no jogo, nem para aumentar o som, mudar os controles ou mudar o idioma. Facilitando a disponibilidade do jogo. E que todos joguem com os mesmos problemas.

5. O jogo não possui cenas violentas

Para evitar as cenas violentas, todas as cenas que poderiam ocasionar uma fratura ou perda de membros ou que geraria sangue, resultará em fim do jogo e tela de encerramento.

6. O jogador pode falar que bateu um recorde e não bateu?

É possível, porem caso consultemos os recordes, é possível notar que tem a hora, portanto se o jogador falar q bateu o recorde, e não tiver nenhum recorde registrado nos últimos minutos, então ele não bateu.



## Filosofia

---

### **1. Por que o jogador perde o jogo se não se curar?**

*A explicação é o zombie é consciente, ou seja, um zombie bom, porem ele precisa encontrar a cura, caso perca o controle, perde-se a consciência e passa a ser um zombie mal.*

### **2. Por que o jogador perde ao cair de uma distância muito alta?**

Foi feito para dificultar um pouco pois o jogo não possui muitos obstáculos, e também para mostrar que não se deve pular de lugares altos. A queda é respeitada a lei da física, caso o jogador atinja a velocidade acima de  $-10\text{m/s}$  ( $36\text{km/h}$ ) ele perde o jogo. Isso equivale a cair de 5 metros de altura com  $-10\text{m/s}^2$  de gravidade.

Toda essa informação acima é apenas uma possível explicação, o jogador é livre para pensar o que quiser.

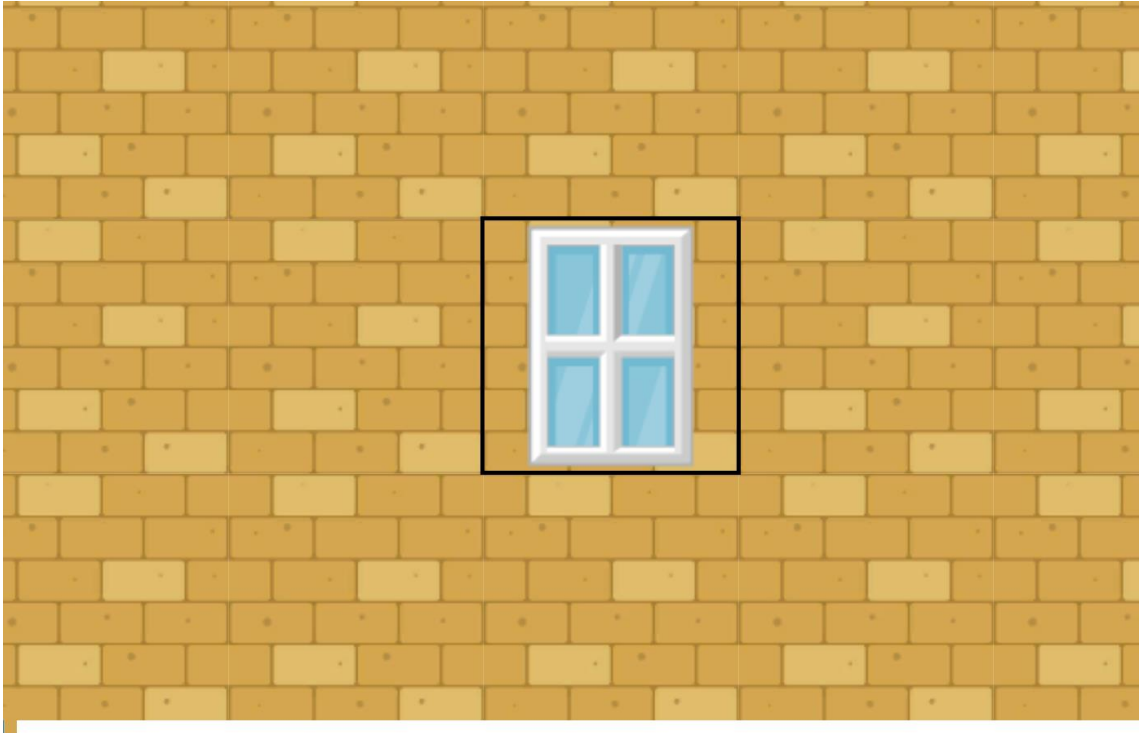
Em fase de desenvolvimento é possível mudar a estrutura visual e sons

Ou seja, considerando a seguinte imagem



É possível alterar tudo na imagem, caso a janela não esteja boa, é possível altera-la, até remover (imagem transparente), no caso ao lado do prédio e usada uma corda como porta, a mesma deverá ser trocada por uma porta.

Mas a troca deve ser feita respeitando a mecânica 2D, onde cada prédio é composto por diversos quadrados e esses quadrados aceitam cores e transparências



Olhando para o tile da janela, ela não ocupa o quadrado inteiro, existe uma transparência ao redor da janela, apesar de mostrar a parede ao fundo, na verdade a janela está sobreposta a parede.

***Para alterar é necessário trocar no código fonte do projeto.***

E não é possível adicionar novos objetos, por exemplo, digamos que houvesse a necessidade de construir 3 janelas na imagem de cima, não é possível. Pois só é possível substituir uma imagem, e não acrescentar. Considere também, que ***ao modificar uma janela, todas as janelas também serão substituídas.***