



REQUISITOS:

OBJETIVO: sobreviver o maior tempo possível, controlando uma minhoca que deve comer frutas para crescer e evoluir.

MECÂNICAS: O jogador controla uma cobra usando controles simples, como toque, teclado, para direcionar a cobra na tela. A cobra come frutas que aparecem aleatoriamente no cenário para crescer. Conforme a minhoca cresce, a dificuldade aumenta automaticamente, tornando o jogo mais desafiador.

MAPA: Consiste em uma área quadrada em 2d limitada por muros.

ARMADILHAS: Existirão dois tipos de armadilha, as fixas ao alcançar determinado número de pontos, como redução no tamanho do mapa, e algumas aleatórias que aparecerão com o passar do tempo, como tiros em um lugar aleatório.

DINÂMICAS:

PONTUACAO: O jogador ganha pontos por cada fruta consumida, e a pontuação total é usada para classificar os jogadores no placar.

POWER-UPS: Power-ups serão adicionados ao jogo para fornecer à cobra habilidades especiais temporárias, como velocidade extra e invencibilidade.

VIDAS: O jogador começa com um número limitado de vidas e o jogo acaba quando todas as vidas são perdidas. Os jogadores ganharão vidas extras ao atingir determinados objetivos, como uma pontuação específica.

BANCO DE DADOS: O banco de dados armazenará a pontuação máxima atingida pelo jogador e os recordes feito pela comunidade em geral.

LISTA DE SUBSTANTIVOS E VERBOS:


SUBSTANTIVOS	ATRIBUIÇÃO
Jogador (Player)	Classe
Fruta	Classe
Game	Classe
Armadilha	Classe
PowerUp	Classe
Tamanho (Jogador)	Atributo de Player
Velocidade (Jogador)	Atributo de Player
Pontos (Jogador)	Atributo de Player
Nome (Jogador)	Atributo de Player
Vidas (Jogador)	Atributo de Player
TipoArmadilha (Armadilha)	Atributo de Armadilha
TipoPowerUp (PowerUp)	Atributo de PowerUp
Duracao (PowerUp)	Atributo de PowerUp
Máximo de Pontos (Game)	Atributo de Game

VERBOS	ATRIBUICAO
Game cria Mundo	membro
Criar jogador	membro
Matar Jogador	membro
Aparecer armadilha	membro
Aparecer Fruta	membro
Move jogador	membro
Comer fruta	membro
Verificar colisao	membro
Atualizar frame	membro
Atualizar Pontuacao	membro

Detalhamento de requisitos e identificação de Classes


1. O Player recebera uma ação de morrer. Através desta ação os mesmos modificarão o status “vivo”, que permitirá a sua permanência no mapa.
2. O atributo “tamanho” é uma variável que se refere ao tamanho do player que aumenta a cada fruta coletada.
3. o atributo “velocidade” é a variável que armazenara a velocidade atual do player.
4. o atributo "pontos" é a variável que armazenara a pontuação atual do player.

MODELS

 Player
+ tamanho: int + velocidade: int + pontos: int + posicao: int + vivo: boolean + nickname: string + vidas: int
+ mover(): void + crescer(): void + power_up(): void

 Armadilha
+ posicao: int + tipo: TipoArmadilha
+ surgir(): void + desativar(): void

 Fruta
+ posicao: int
+ surgir(): void

 PowerUp
+ tipo: TipoPowerUp + posicao: int + duracao: int
+ ativar(): void + desativar(): void


 Game
+ cobra: Player + frutas: Fruta + armadilhas: Armadilha + power_ups: PowerUp + pontuacao_maxima: int
+ iniciar_jogo() + encerrar_jogo() + atualizar_jogo() + verificar_colisoes() + coletar_fruta() + ativar_power_up() + adicionar_armadilha()

DIAGRAMA DE CLASSE

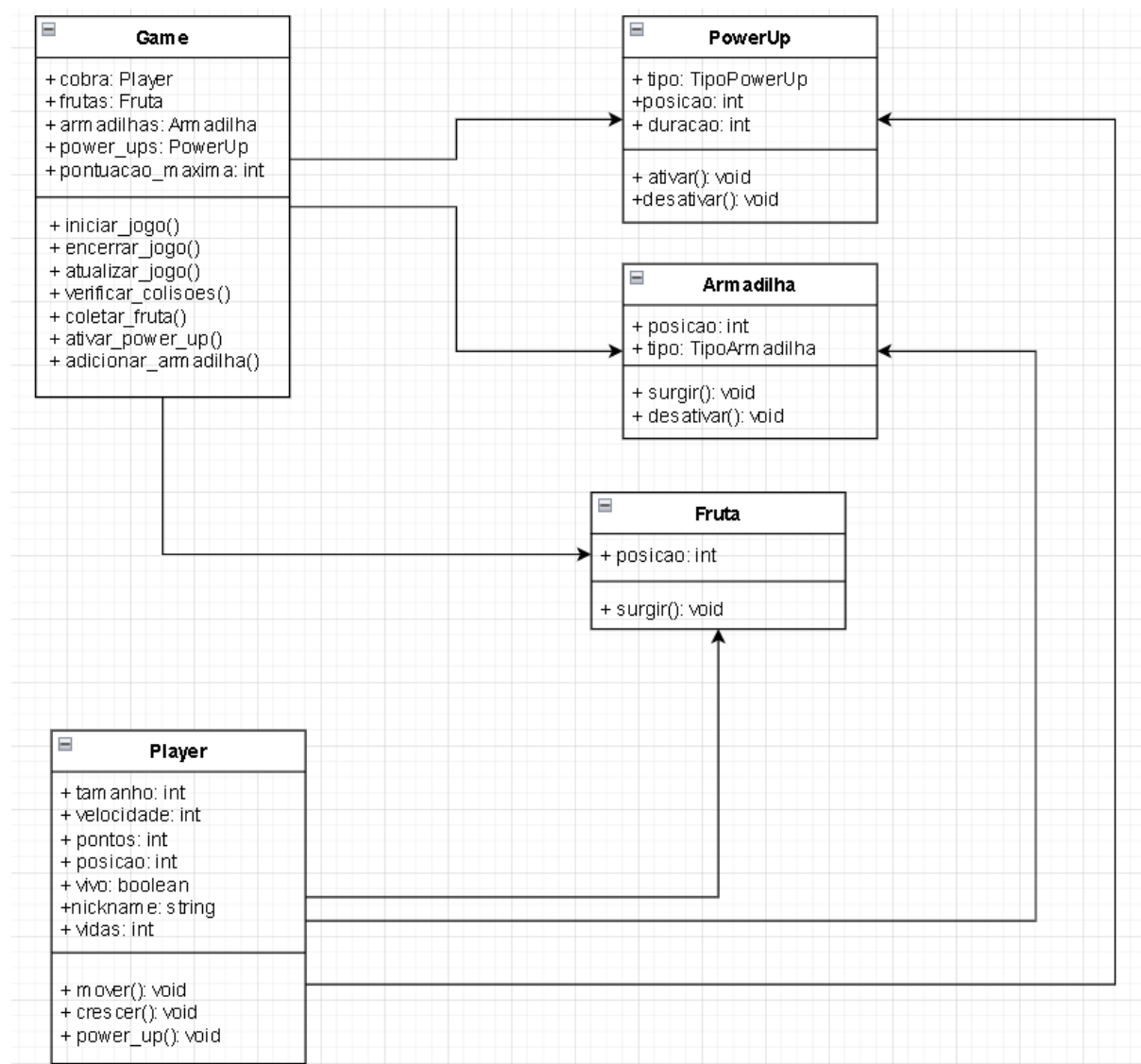
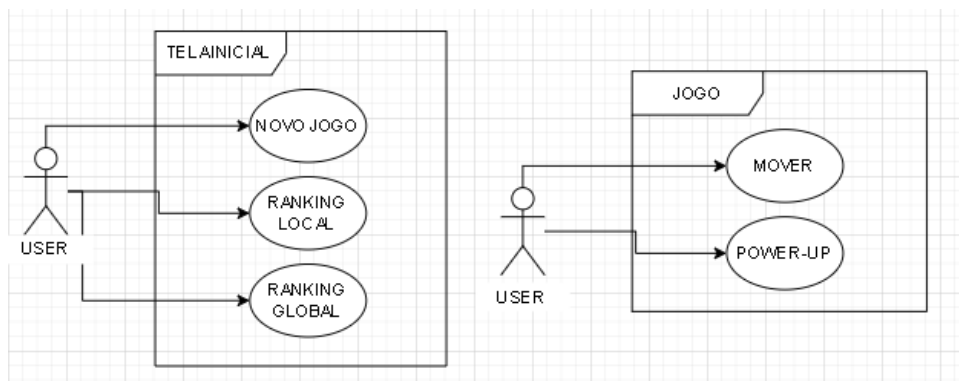


TABELA DE CONSTRUTORES

CLASSE	CONSTRUTOR
Game	vazio
Player	(nickname, resources)
PowerUp	vazio
Armadilha	vazio
Fruta	vazio

DIAGRAMA DE CASO DE USO



O diagrama acima demonstra as utilidades de cada tela do jogo. Na tela inicial, temos as opções de (1) iniciar novo jogo, (2) ver ranking local, (3) ver ranking global. Na tela de Jogo, temos as ações do jogador, (1) movimentar ou (2) usar power up. No início da partida, os usuários podem definir o nome (nickname) que será usado para compor o ranking geral.