PROJETO TOWER DEFENSE

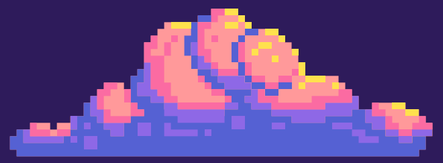
Programação para jogos

Feito por: Felipe Soares

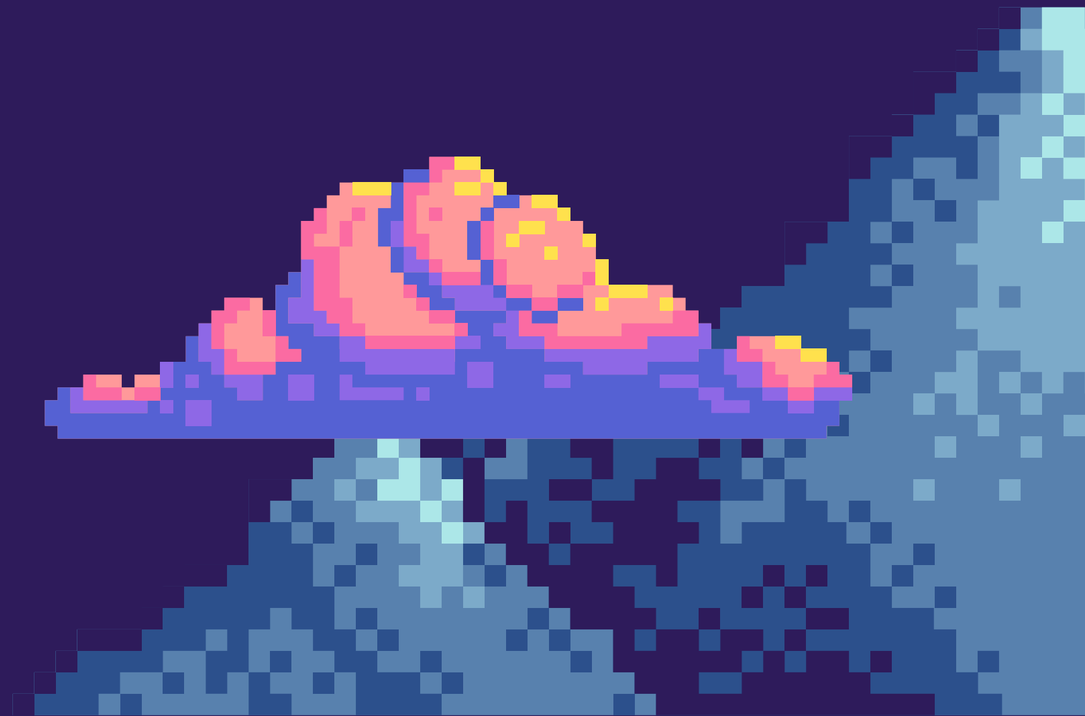
IDEIA DO JOGO

- Gastar dinheiro posicionando torres que atiram automaticamente
- Defender a base contra ondas de inimigos
- Perder se a vida chegar a zero
- Ganhar derrotando todos os inimigos ou passando da onda sem a vida chegar a zero

FERRAMENTA E PADRÕES USADOS



- HTML/Javascript
- Game loop
- Update
- Prototype



CLICK DO MOUSE

```
    }
    static mouseClick(evt){
        var rect = canvas.getBoundingClientRect();
        var root = document.documentElement;
        MouseInput.clickX = evt.clientX - rect.left - root.scrollLeft;
        MouseInput.clickY = evt.clientY - rect.top - root.scrollTop;
    }

}
```

```
window.onload = function() {
    canvas = document.getElementById('gameCanvas');
    canvasContext = canvas.getContext('2d');
    requestAnimationFrame(mainLoop);

    canvas.addEventListener('mousemove', MouseInput.updateMousePos);
    canvas.addEventListener('click', MouseInput.mouseClick);

}
```

PROTOTYPE

- Inimigos
- Torres

INIMIGO E PLAYER

```
class Tower{
    constructor (atkSpeed, damage, rangeRadius, cost) {
```

```
class Enemy {
  constructor(speed, life, color, damage, bounty, radius = 15)
```

```
class Tower1 extends Tower{
    constructor(posX = 735, posY = 200){
        super(0.2, 1, 160, 6);
```

```
class Enemy1 extends Enemy{
    constructor(posX = 475, posY = 0){
        super(0.10, 1, "Red", 1, 1);
```

INIMIGOS

```
var enemy1 = new Enemy1();
```

```
this.num_enemies = 10;
this.enemies = [];
```

```
levels(){
    if (this.wave == 1){
        for(var i=0;i<this.num_enemies;i++){
            var enemy = Object.create(enemy1);
            enemy.posY = -35*i;
            this.enemies.push(enemy);
        }
    }
}
```

```
move(deltaTime){
    this.levels();
    for(var i=0;i<this.num_enemies;i++){
        this.enemies[i].move(deltaTime);
```

TORRES

```
var tower1 = new Tower1(1000,800);
```

```
buyTower(){
    if(this.comprou == false && Player.money >= tower1.cost && MouseInput.clickX > 735 && MouseInput.clickX < 765 && MouseInput.clickY > 200 && MouseInput.clickY < 230){

        var tower = Object.create(tower1);
        entities.push(tower);
```

```
aimTarget(targetX, targetY){
    var difX = this.posX - targetX;
    var difY = this.posY - targetY;

    var angulo = Math.atan2(difY, difX);

    return angulo;
}
```

```
if(this.enemies[0].verificaColisao(entities[i])){
    entities[i].degrees = entities[i].aimTarget(this.enemies[0].posX, this.enemies[0].posY)*50;
```

DETECÇÃO DE COLISÃO

```
verificaColisao(outroCirculo) {
    const distancia = Math.sqrt((this.posX - outroCirculo.posX) ** 2 + (this.posY - outroCirculo.posY) ** 2);

    return distancia < this.radius + outroCirculo.rangeRadius;
}
```

OBRIGADO
E BOA NOITE