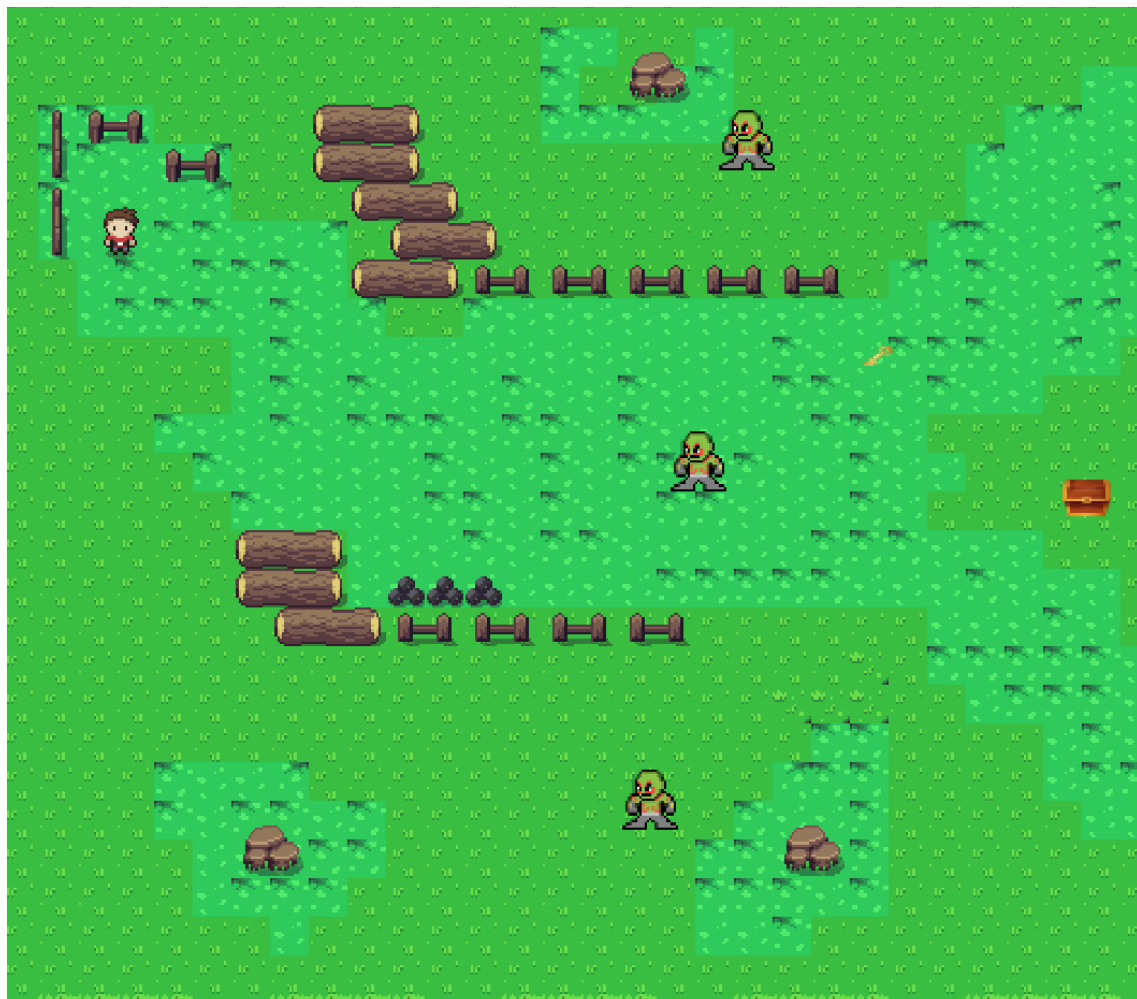


Phaser 2D Game

Unidade Curricular: Sistemas Multimédia

Escola Superior de Tecnologia e Gestão

Engenharia da Computação Gráfica e Multimédia



Introdução

Para a unidade curricular de Sistemas Multimédia, foi-me pedido para criar um jogo em Phaser utilizando todo o conhecimento que nos foi passado durante as aulas.

O trabalho tinha que ser alusivo ao ambiente e ter uma série de requisitos sobre a matéria lecionada.

Objetivo do jogo: O objetivo do jogo é obter o tesouro necessário para salvar o planeta. Quando o player encontrar o tesouro juntamente coma chave o jogo termina e o planeta é salvo. Para abrir o tesouro o jogador terá que recolher a chave que está num determinado local do mapa. Existirão também inimigos no mapa que quando em contacto com o personagem fazem com que o jogo termine.

Regras do Jogo

As regras do jogo são simples. O jogador apenas se pode mover usando as teclas direcionais do computador e só pode acabar o jogo depois de obter a chave e alcançar o baú.

Durante o seu percurso o jogador vai encontrar inimigos, caso os inimigos entrem em contacto com o jogador o jogo termina e o planeta é destruído.

Implementação do jogo

Visto que o ambiente era tema do trabalho resolvi então usar um mapa bastante verde e alusivo à natureza, para além disso escolhi zombies para serem os inimigos visto que eles são conhecidos por “destruir” o planeta e os seres humanos.

Esta é a tela de jogo:

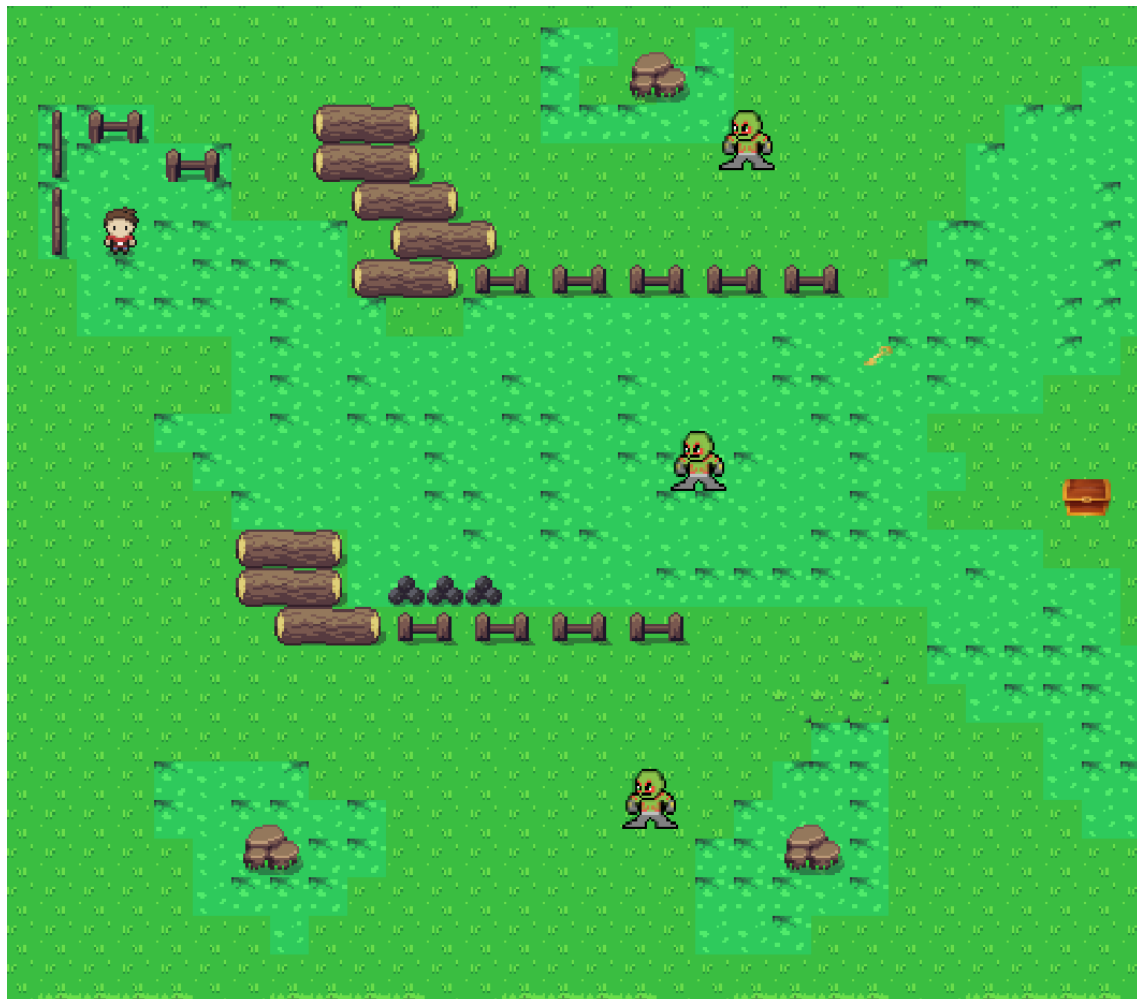


Imagem do Jogo

Para implementar esta cena utilizei um tilemap e algumas spritesheets. O mapa foi criado no software tiled e contém duas camadas, a camada base e uma camada de obstáculos onde se encontram todos os objetos que o player não pode ultrapassar.

Para fazer a animação do personagem utilizei uma spritesheet e fui alterando os frames que eram selecionados dependendo da tecla que o jogador está a clicar.

```
create : function() {
  this.som = this.sound.add('som', { loop: true });
  this.auch = this.sound.add('auch', { loop: false });
  this.som.play();
  // -- criar o "mundo" do jogo
  var mapa = this.make.tilemap({ key : 'map' });

  // -- obter o nome de um tileset
  // -- chamado "spritesheet"
  // -- disponível no ficheiro JSON
  var tiles = mapa.addTilesetImage('spritesheet', 'tiles', 16, 16);

  // -- obter a layer "Relva" e adicionar ao mapa
  var relva = mapa.createStaticLayer('Relva', tiles, 0, 0);

  // -- obter a layer "Obstaculos" e adicionar ao mapa
  var obstaculos = mapa.createStaticLayer('Obstaculos', tiles, 0, 0);

  // -- adição do tesouro
  this.treasure = this.physics.add.sprite(450, this.sys.game.config.height / 2, 'treasure');
  this.treasure.setScale(0.6);

  // -- adição do tesouro
  this.key = this.add.sprite(Phaser.Math.RND.between(350, 410), Phaser.Math.RND.between(0, 450), 'key');
  this.key.setScale(0.6);

  // -- obstaculos estarão disponíveis para colisão
  obstaculos.setCollisionByExclusion([-1]);
}
```

Criação do mapa e personagens

```
91 // -- animação do player
92 this.anims.create({
93   key: 'esquerdadireita',
94   frames: this.anims.generateFrameNumbers('player', { frames: [18, 19, 20, 21] }),
95   frameRate: 10,
96   repeat: -1
97 });
98
99 this.anims.create({
100   key: 'up',
101   frames: this.anims.generateFrameNumbers('player', {frames: [36, 37, 38, 39]}),
102   frameRate: 10,
103   repeat: -1
104 });
105
106 this.anims.create({
107   key: 'down',
108   frames: this.anims.generateFrameNumbers('player', {frames: [0, 1, 2, 3]}),
109   frameRate: 10,
110   repeat: -1
111 });
112
113
114 // group of enemies
115 this.enemies = this.physics.add.group({
116   key: 'inimigo',
117   repeat: 2,
118   flipY: -1,
119   setXY: {
120     x: 310
```

Animação do personagem e criação do grupo de inimigos

Como podemos ver na imagem acima, criei também um grupo de inimigos, são 3 no total e estão a andar de um lado para o outro da tela para que o player não possa passar com tanta facilidade.

Estes inimigos colidem também com os objetos e com as bordas do mapa de forma a manterem-se sempre dentro do ecrã.

```
126
127     this.physics.add.collider(obstaculos, this.enemies);
128
129     //scale enemies
130     Phaser.Actions.ScaleXY(this.enemies.getChildren(), -0.5, -0.5);
131
132     // set speeds
133     Phaser.Actions.Call(this.enemies.getChildren(), function(enemy) {
134         enemy.speed = 2 + Math.random() * 1.5;
135     }, this);
136
137
138 },
```

Escala e velocidade dos inimigos

```
170     //enemy movement
171     let enemies = this.enemies.getChildren();
172     let numEnemies = enemies.length;
173
174     for(let i=0; i < numEnemies; i++){
175
176         //move enemies
177         enemies[i].y += enemies[i].speed;
178
179         // reverse movement if reached the edges
180         if(enemies[i].y >= 462 && enemies[i].speed > 0){
181             enemies[i].speed *= -1;
182         }else if(enemies[i].y <= 16 && enemies[i].speed < 0){
183             enemies[i].speed *= -1;
184         }
185
186         // inverter as direções dos inimigos ao colidir com os obstaculos
187         if(enemies[0].y >= 0 && enemies[0].y < 102){
188             enemies[0].speed *= 1;
189         }else{
190             enemies[0].speed *= -1;
191         }
192         if(enemies[1].y >= 140 && enemies[1].y < 245){
193             enemies[1].speed *= 1;
194         }else{
195             enemies[1].speed *= -1;
196         }
197         if(enemies[2].y >= 283 && enemies[2].y < 470){
198             enemies[2].speed *= 1;
199         }else{
200             enemies[2].speed *= -1;
201         }
202     }
```

Movimento dos inimigos

No jogo está também implementada a colisão do player com a chave e com o baú. Caso o player tenha a chave então é alterado um boolean para verdadeiro e a próxima colisão do player com o baú vai terminar o jogo em vitória.

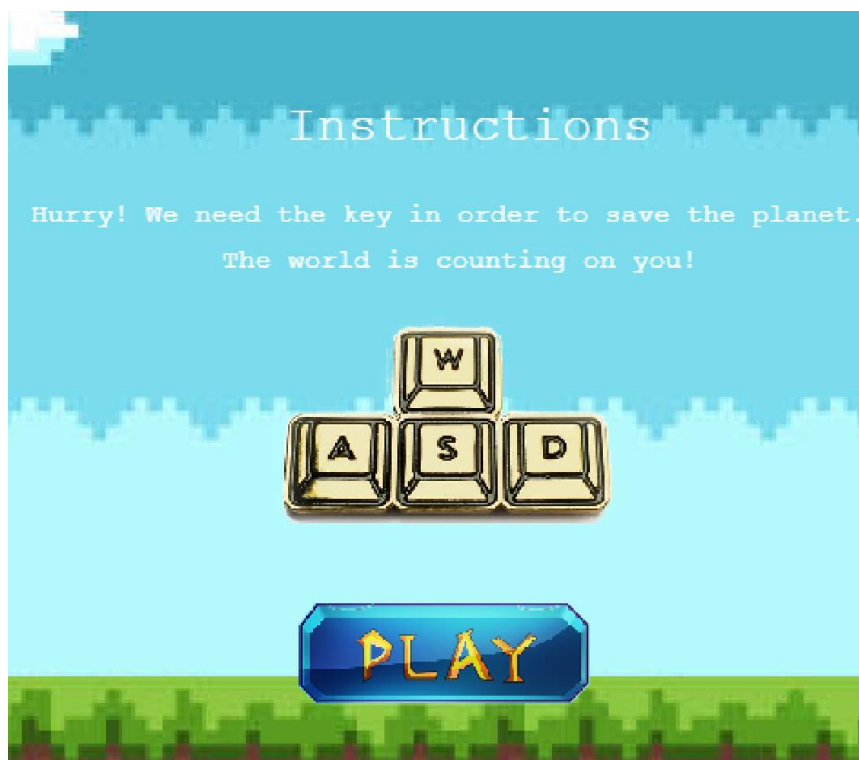
```
205 //colisao com a chave
206 if (Phaser.Geom.Intersects.RectangleToRectangle(this.player.getBounds(), this.key.getBounds()) && keyCount
207     this.key.destroy();
208     keyCount = 1;
209     console.log(keyCount);
210 }
211
212 // colisao do player com o tesouro
213 this.physics.add.overlap(this.player, this.treasure, this.colisaoTesouro);
214
215 //colisao do player com os inimigos
216 this.physics.add.overlap(this.player, this.enemies, this.gameOver);
217
218 // verifica se o jogo terminou
219 if(acabado == true){
220     this.auch.play();
221     acabado = false;
222     this.som.stop();
223     this.scene.start('CenaOverLost');
224 };
225 if(acabado1 == true){
226     acabado1 = false;
227     this.som.stop();
228     this.scene.start('CenaOver');
229 };
230
```

Colisões e gameOver

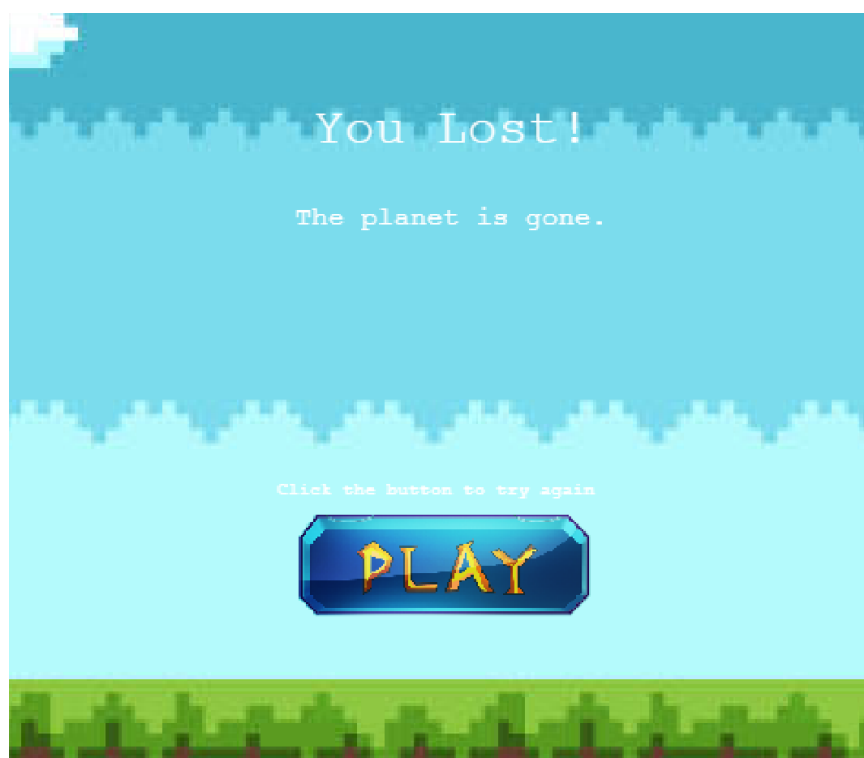
```
233
234 // colisao do tesouro
235 colisaoTesouro: function(player, tesouro){
236     if(keyCount == 1){
237         acabado1 = true;
238     }else if(keyCount == 0){
239         console.log("key missing");
240     }
241 },
242
243 // gameOver
244 gameOver : function(){
245     acabado = true;
246 },
247 });
248
```

Função para colidir e terminar o jogo

Para além disto temos também outras 3 cenas criadas, uma cena para começar o jogo, uma para terminar com vitória e outra para terminar com a derrota.



Tela de início



Tela de jogo perdido/ganho

```

initialize :

function CenaInicial(){
    Phaser.Scene.call(this, { key : 'CenaInicial' });
},

preload : function() {
    this.load.image('background', 'assets/background.jpg');
    this.load.image('play', 'assets/play1.png');
    this.load.image('instrucao', 'assets/instrucao.png');
    this.load.image('spider', 'assets/spider.png');
},

create : function(){
    console.log("olaInicial");

    this.add.image(this.game.renderer.width / 2, this.game.renderer.height / 2, 'background').setDepth(0).setVisible(true);

    let playButton = this.add.image(this.game.renderer.width / 2 , this.game.renderer.height / 2 + 150, 'play');

    this.add.text(this.game.renderer.width / 2 - 85, this.game.renderer.height / 4 - 50, "Instructions", {fontSize: 24, color: 'white'});
    this.add.text(this.game.renderer.width / 2 - 225, this.game.renderer.height / 2 - 100, "Hurry! We need", {fontSize: 18, color: 'white'});
    this.add.text(this.game.renderer.width / 2 - 120, this.game.renderer.height / 2 - 75, "The world is counting on you!", {fontSize: 18, color: 'white'});
    this.add.image(this.game.renderer.width / 2, this.game.renderer.height / 2 + 40, 'instrucao').setDepth(1);

    playButton.setInteractive();

    playButton.on('pointerdown', () => this.scene.start('CenaMundo') );
}

```

Criação da cena inicial

```

initialize :

function CenaOver(){
    Phaser.Scene.call(this, { key : 'CenaOver' });
},

preload : function() {
    this.load.image('background', 'assets/background.jpg');
    this.load.image('play', 'assets/play1.png');
    this.load.image('instrucao', 'assets/instrucao.png');
    this.load.image('spider', 'assets/spider.png');
},

create : function(){
    this.add.image(this.game.renderer.width / 2, this.game.renderer.height / 2, 'background').setDepth(0).setVisible(true);

    let playButton = this.add.image(this.game.renderer.width / 2 , this.game.renderer.height / 2 + 100, 'play');

    this.add.text(this.game.renderer.width / 2 - 50, this.game.renderer.height / 4 - 50, "You Won", {fontSize: 24, color: 'white'});
    this.add.text(this.game.renderer.width / 2 - 150, this.game.renderer.height / 2 - 100, "The planet is saved!", {fontSize: 18, color: 'white'});
    this.add.text(this.game.renderer.width / 2 - 90, this.game.renderer.height / 2 + 50, "Click the button to play again", {fontSize: 18, color: 'white'});

    playButton.setInteractive();

    playButton.on('pointerdown', () => this.scene.start('CenaInicial') );
}

```

Criação da cena de jogo ganho


```

function CenaOverLost(){
    Phaser.Scene.call(this, { key : 'CenaOverLost' });
},

preload : function() {
    this.load.image('background', 'assets/background.jpg');
    this.load.image('play', 'assets/play1.png');
    this.load.image('instrucao', 'assets/instrucao.png');
},

create : function(){

    this.add.image(this.game.renderer.width / 2, this.game.renderer.height / 2, 'background').setDepth(0).setVisible(true);

    let playButton = this.add.image(this.game.renderer.width / 2 , this.game.renderer.height / 2 + 100, 'play');

    this.add.text(this.game.renderer.width / 2 - 70, this.game.renderer.height / 4 - 50, "You Lost!", {font: 'Arial', size: 24, color: 'red'});
    this.add.text(this.game.renderer.width / 2 - 80, this.game.renderer.height / 2 - 100, "The planet is gone", {font: 'Arial', size: 18, color: 'black'});
    this.add.text(this.game.renderer.width / 2 - 90, this.game.renderer.height / 2 + 50, "Click the button to restart", {font: 'Arial', size: 18, color: 'black'});

    playButton.setInteractive();

    playButton.on('pointerdown', () => this.scene.start('CenaInicial') );

}

```

Criação da cena de jogo perdido

Por fim adicionei alguns sons para tornar o jogo mais divertido. Mal o jogo inicia podemos ouvir uma música de fundo e sempre que o jogador colide com um inimigo podemos ouvir um “ouch” de fundo.

Quanto à conclusão do que foi pedido, o jogo cumpre todos os requisitos mínimos e alguns adicionais. Tem diversas Scenes, no caso são 5, o personagem está animado utilizando spritesheets que vão alterando o frame ao clicar nas teclas de direção. Tem diversas colisões entre objetos, inimigos, chave, baú e player. Contém dois efeitos sonoros diferentes e está claro tudo estruturado em classes e devidamente comentado. Tem também um nível, infelizmente não consegui fazer mais do que um. E tem a inclusão de tilemaps que foram feitos no software Tiled.

Conclusão

Embora tenha conseguido concluir o que era pretendido houveram algumas coisas que eu gostaria de mudar e que poderiam estar melhores. Uma delas é o facto de o jogo só ter um nível, inicialmente estava com ideias de fazer 3 níveis mas não consigo concluir essa tarefa a tempo. Para além disso, um mapa maior e com mais inimigos seria mais interessante pois dava para o jogador estar mais tempo durante o jogo e investigar aquilo que há no mapa.

Quanto a problemas do jogo, deparei com um que não consegui resolver, que é o facto da cena que contém o load dos elementos não estar a funcionar corretamente, tentei de várias formas corrigir o erro mas não consegui e portanto tive que implementar esta parte do código nas próprias cenas. E também acontece um pequeno bug quando o player colide com o baú sem ter a chave. Não é um bug grave mas se não existisse era melhor.

Tirando estes fatores acho que o jogo está bem conseguido, penso que consegui abordar as partes principais da matéria e acho que o jogo está fácil de jogar, fácil de entender e está bastante agradável visualmente.