

Curso Técnico de Gestão e Programação de Sistemas Informáticos

Relatório PAP



“JUMPMAN”

Tiago César Nº14

2017/2018

Índice geral

Introdução	3
Objetivos	4
Tecnologias Utilizadas	5
Estado da arte	7
Fases de desenvolvimento	8
Cronograma.....	9
Conclusão	10
Melhoramentos.....	10
Webgrafia.....	11
Anexos	12

Introdução

Este trabalho surgiu no âmbito da realização da PAP, que está contemplada no desenho curricular do curso.

Este trabalho tem como principal objetivo a aprendizagem de uma nova linguagem de programação e de uma nova plataforma de criação de jogos, o Unity.

As linguagens de programação de desenvolvimento de aplicações nesta plataforma podem ser o C# ou o C++, no meu caso utilizei o C++.

A PAP consiste na realização de um jogo desenvolvido no Unity, em que o objectivo visa testar principalmente a destreza a nível motor dos utilizadores.

O jogo consiste em controlar um personagem através de vários obstáculos e só termina quando o personagem colide com um desses obstáculos, por isso, o jogo tem apenas um nível.

Objetivos

A nível de objectivos, espero conseguir complementar as minhas aprendizagens com a exploração de novos conhecimentos, como referi anteriormente, ao nível da programação. Visto que a área da informática está sempre a evoluir, torna-se necessário a formação pessoal em novos conteúdos de uma forma constante. Escolhi a realização de um jogo, porque é um tema bastante atrativo e do qual eu gosto bastante, e assim torna-se bastante mais motivante a realização da PAP.

Tecnologias Utilizadas

As tecnologias utilizadas para o desenvolvimento deste jogo foram as seguintes:

1.1 Unity

Unity, também conhecido como Unity 3D, é um motor de jogo 3D criado pela Unity Technologies. Unity é similar ao Blender, Virtools ou Torque Game Engine, em relação a sua forma primária de autoria de jogos. O Unity tem varias versões, tais como, Unity 1,2,3,4,5 e Unity 2017. A unidade é notável por sua capacidade de direcionar jogos para múltiplas plataformas. Dentro de um projeto, os desenvolvedores têm controlo sobre a entrega para dispositivos móveis, web browsers, desktops e consoles. As plataformas suportadas incluem BlackBerry 10, o Windows Phone 8, Windows, OS X, Linux (Ubuntu, principalmente), Android, iOS, Unity Web Player (incluindo Facebook, Adobe Flash, PlayStation 3, PlayStation 4, PlayStation Vita, Xbox 360, Xbox One, Wii U, e Wii.

1.2 C++

C++ é uma linguagem de programação compilada multi-paradigma e de uso geral. Desde o ano de 1990 que é umas das linguagens comerciais mais populares, sendo bastante usada também na academia por seu grande desempenho e base de utilizadores. Bjarne Stroustrup desenvolveu o C++, originalmente com o nome *C with Classes*, em 1983 no **Bell Telephone Laboratories** como um adicional à linguagem C. Novas características foram adicionadas com o tempo, como funções virtuais, sobrecarga de operadores, herança múltipla, gabaritos e tratamento de exceções. Após a padronização ISO realizada em 1998 e a posterior revisão realizada em 2003, uma nova versão da especificação da linguagem foi lançada em dezembro de 2014, conhecida informalmente como C++17.

1.3 Microsoft Visual Studio

Microsoft Visual Studio é um ambiente de desenvolvimento integrado da Microsoft para desenvolvimento de software especialmente dedicado ao .NET Framework e às linguagens Visual Basic (VB), C, C++, C# (C Sharp) e J# (J Sharp). Também é um produto de desenvolvimento na área web, usando a plataforma do ASP.NET, como websites, aplicativos web, serviços web e aplicativos móveis.

1.4 WordPress

WordPress é uma aplicação de sistema de gestão de conteúdo para *web*, escrito em PHP com banco de dados MySQL, direcionado para a criação de sites e blogs via web. O WordPress é adotado por aqueles que pretendem um site mais profissional e com maiores recursos.

1.5 Photoshop

Adobe Photoshop é um software caracterizado como editor de imagens desenvolvido pela Adobe Systems. É considerado o líder no mercado dos editores de imagem profissionais, usado para edição profissional de imagens digitais e trabalhos de pré-impressão.

Estado da arte

Apesar de nunca ter trabalhado com o Unity, fui adquirindo conhecimentos ao longo do curso, que me permitem começar a explorar esta plataforma de uma forma diferente. Antes de ter escolhido o Unity, pesquisei sobre a plataforma na internet em fóruns e no youtube, e fiquei com a noção de como trabalhar o Unity.

Desta forma passo a descrever o objectivo, a funcionalidade e a parte gráfica do jogo:

Objectivo

O objectivo do jogo é tentar conduzir o personagem principal sem colidir com os diferentes objectos que vão surgindo no mapa do jogo.

Funcionalidades

As funcionalidades do jogo são o controlo do personagem, que pode saltar e deslizar para evitar os objectos, a velocidade com que os objectos se movem e a velocidade que é percorrido o mapa do jogo, ou seja, conforme se vai progredindo no jogo assim vai aumentando a velocidade do mapa e por conseguinte a velocidade com que os objectos se movimentam.

Parte gráfica

Na parte gráfica do jogo, tentei apresentar um design simples mas ao mesmo tempo bastante atractivo.

Fases de desenvolvimento

1. Pesquisa sobre a plataforma Unity.
2. Aprendizagem da plataforma Unity.
3. Pesquisa sobre a linguagem C++.
4. Aprendizagem sobre a linguagem C++.
5. Implementação do personagem.
6. Programação da movimentação do personagem.
7. Criação dos obstáculos.
8. Programação da movimentação dos obstáculos.
9. Implementação das texturas.
10. Pós-produção do jogo.
11. Realização de um site para apresentar o jogo.

Cronograma

Descrição das etapas	Novembro	Dezembro	Janeiro	Fevereiro	Março	Abril
Pesquisa sobre a plataforma Unity						
Aprendizagem da plataforma Unity						
Pré-produção ou planeamento do jogo						
Criação do jogo						
Pós-produção do jogo						
Realização de uma aplicação para apresentar o jogo						

Conclusão

Como conclusão, gostaria de referir que espero que este projeto melhore as minhas competências nas áreas acima referidas e simultaneamente me permita concluir com êxito o meu ciclo de formação.

Este trabalho permitiu-me aprender mais sobre o grafismo usado na plataforma Unity, neste caso em 2D mas também poderá ser utilizado em algum trabalho futuro em 3D.

Também espero que este tipo de trabalho me dê motivação para trabalhos futuros no mercado de trabalho, ao seguir este tipo de profissão ou mesmo qualquer outro tipo de profissão.

Melhoramentos

Tenciono melhorar este jogo adicionando uma parte competitiva criando uma base de dados que guarde as melhores pontuações dos jogadores, tornando o jogo mais apelativo para os jogadores, podendo até haver prémios para os melhores jogadores.

Pretendo melhorar o jogo graficamente para se tornar mais atrativo para os jogadores.

Pretendo tornar o jogo multiplataforma para abranger o maior número de pessoas, com diferentes faixas etárias.

Webgrafia

<https://pt.wikipedia.org/wiki/Unity>

<https://unity3d.com/pt>

<https://pt.wikipedia.org/wiki/C%2B%2B>

<https://pt.wordpress.org/>

<https://pt.wikipedia.org/wiki/WordPress>

https://pt.wikipedia.org/wiki/Microsoft_Visual_Studio

<https://www.visualstudio.com/pt-br/?rr=https%3A%2F%2Fwww.google.pt%2F>

https://pt.wikipedia.org/wiki/Adobe_Photoshop

<https://www.adobe.com/pt/products/photoshop/free-trial-download.html>

Anexos

Design do jogo:

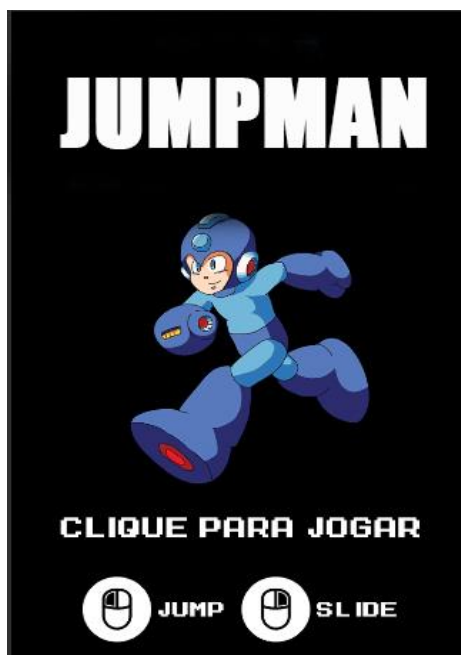


Fig.1 - Este é o menu principal no qual mostra como jogar e como começar a jogar.

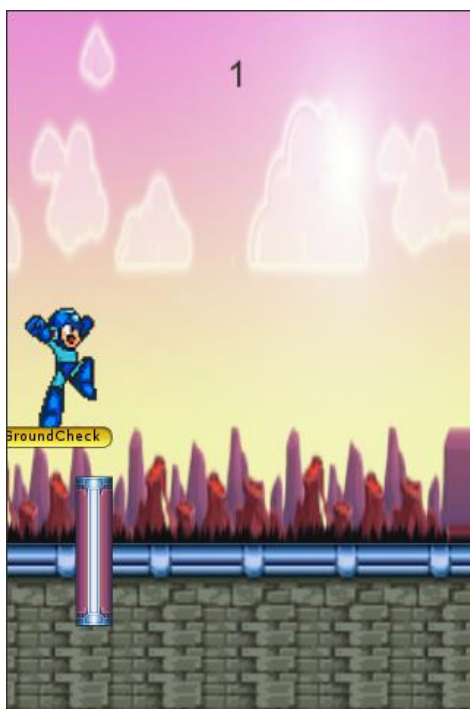


Fig.2 - Este print screen mostra o personagem a saltar.

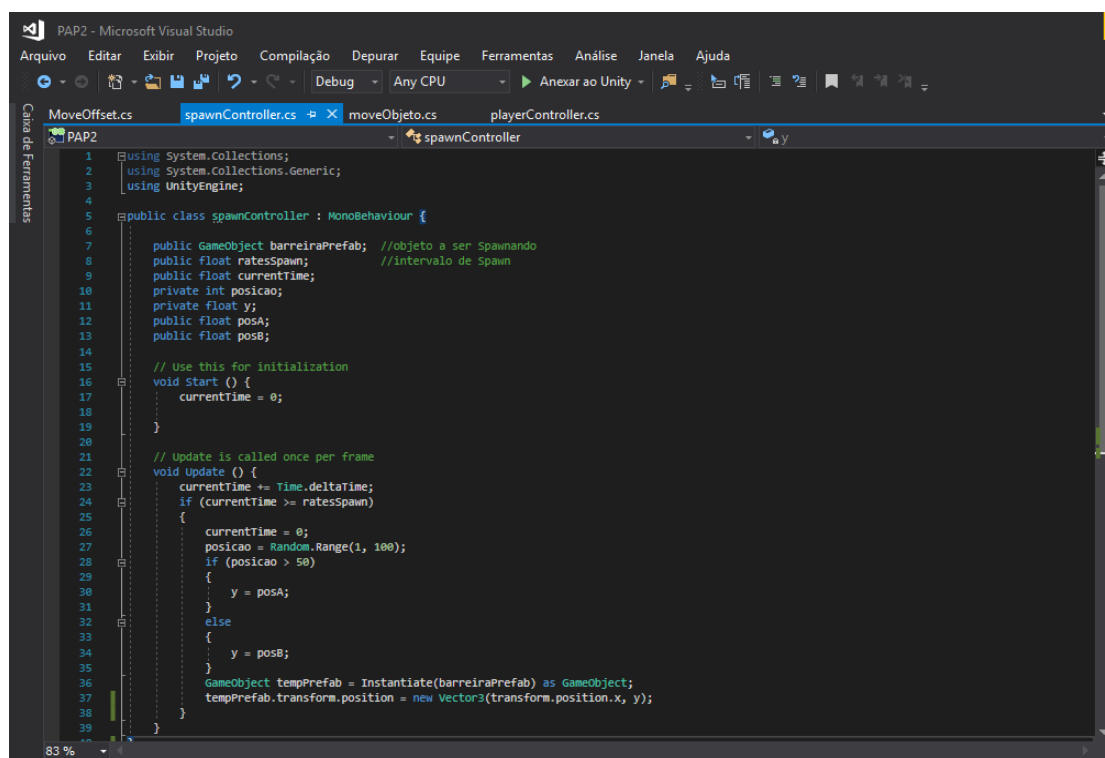


Fig.3 - Este print screen mostra o personagem a deslizar.



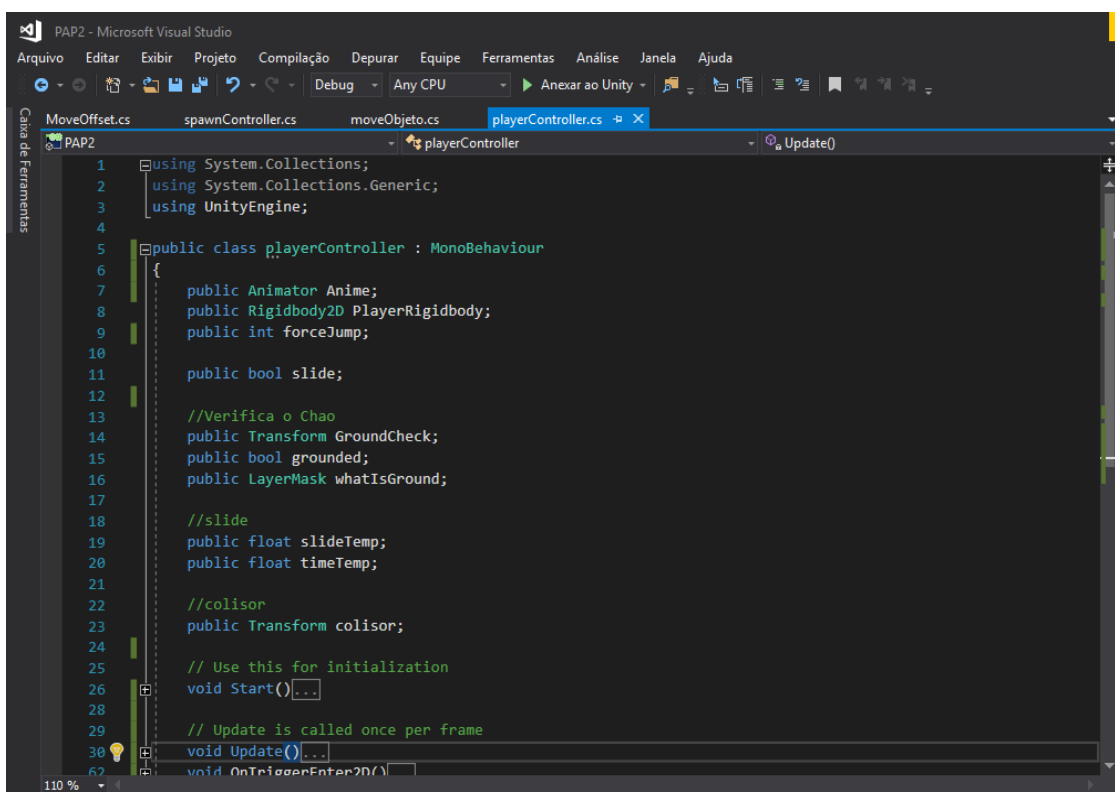
Fig.4 - Este print screen mostra a pontuação feito caso se bata num obstáculo.

Estes prints screens mostram algum dos códigos de programação do jogo:



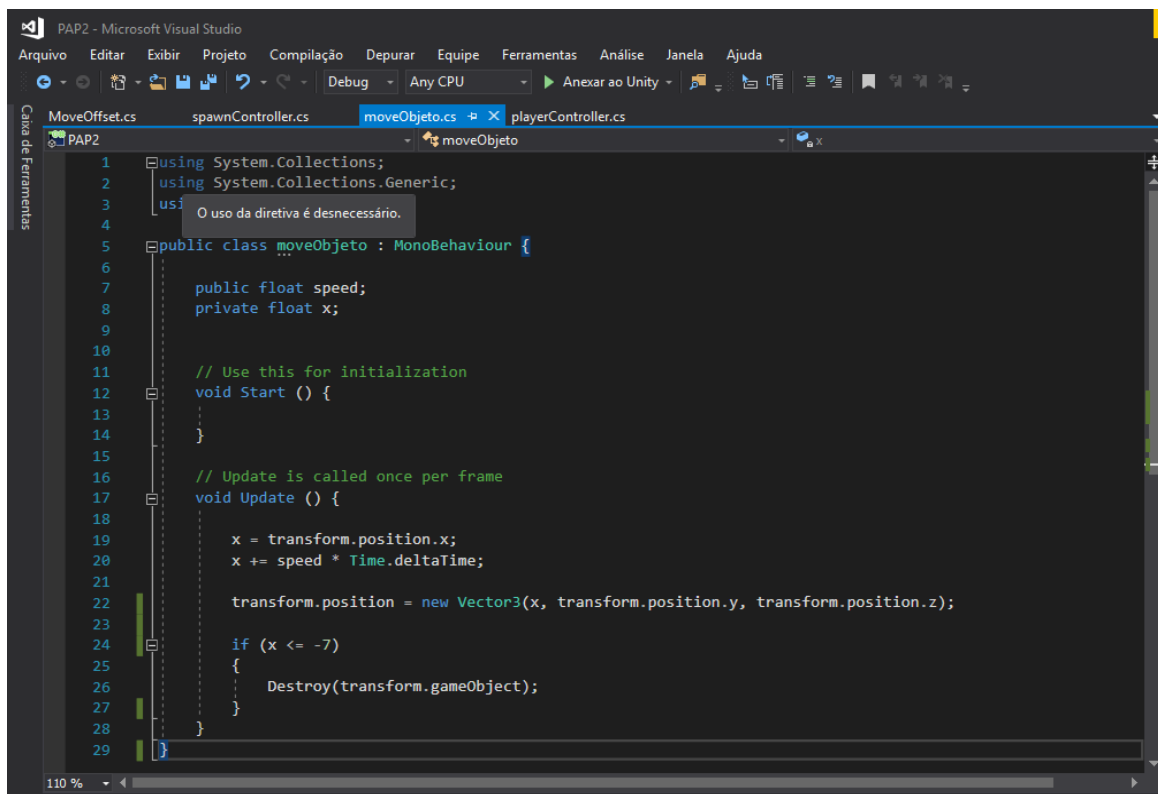
```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class spawnController : MonoBehaviour {
6
7     public GameObject barreiraPrefab; //objeto a ser Spawnando
8     public float ratesSpawn; //intervalo de spawn
9
10    private int posicao;
11    private float y;
12    public float posA;
13    public float posB;
14
15    // Use this for initialization
16    void Start () {
17        currentTime = 0;
18    }
19
20
21    // Update is called once per frame
22    void Update () {
23        currentTime += Time.deltaTime;
24        if (currentTime >= ratesSpawn)
25        {
26            currentTime = 0;
27            posicao = Random.Range(1, 100);
28            if (posicao > 50)
29            {
30                y = posA;
31            }
32            else
33            {
34                y = posB;
35            }
36            GameObject tempPrefab = Instantiate(barreiraPrefab) as GameObject;
37            tempPrefab.transform.position = new Vector3(transform.position.x, y);
38        }
39    }
40 }
```

Fig.5 – Código que cria os obstáculos automaticamente.



```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class playerController : MonoBehaviour
6 {
7     public Animator Anime;
8     public Rigidbody2D PlayerRigidbody;
9     public int forceJump;
10
11     public bool slide;
12
13     //Verifica o Chao
14     public Transform GroundCheck;
15     public bool grounded;
16     public LayerMask whatIsGround;
17
18     //slide
19     public float slideTemp;
20     public float timeTemp;
21
22     //colisor
23     public Transform colisor;
24
25     // Use this for initialization
26     void Start() { ... }
27
28
29     // Update is called once per frame
30     void Update() { ... }
31
32     void OnTriggerEnter2D() { ... }
33 }
```

Fig.6 - Código que faz com que o personagem reaja aos comandos do utilizador.



```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class moveObjeto : MonoBehaviour {
6
7     public float speed;
8     private float x;
9
10
11     // Use this for initialization
12     void Start () {
13
14     }
15
16     // Update is called once per frame
17     void Update () {
18
19         x = transform.position.x;
20         x += speed * Time.deltaTime;
21
22         transform.position = new Vector3(x, transform.position.y, transform.position.z);
23
24         if (x <= -7)
25         {
26             Destroy(transform.gameObject);
27         }
28     }
29 }
```

Fig.7 – Código que movimenta os obstáculos.

Site para partilha do jogo (WordPress):

Este site foi criado no intuito de partilhar o jogo para o maior número de pessoas.

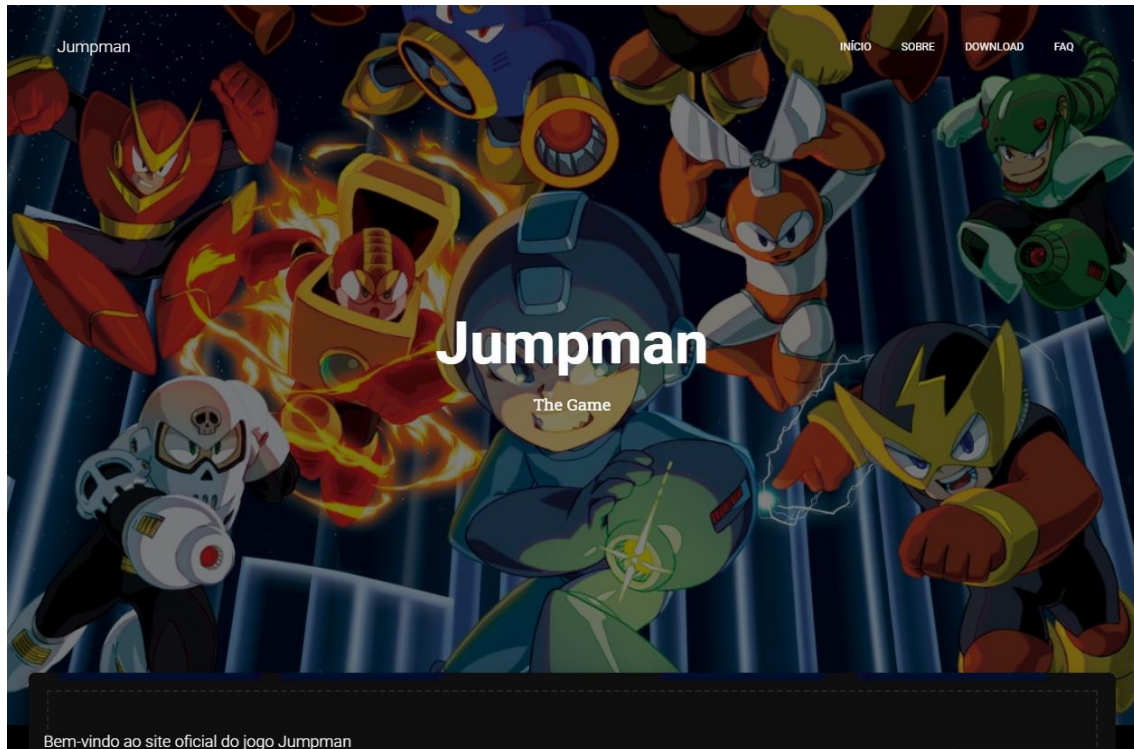


Fig.8 - Página Inicial.

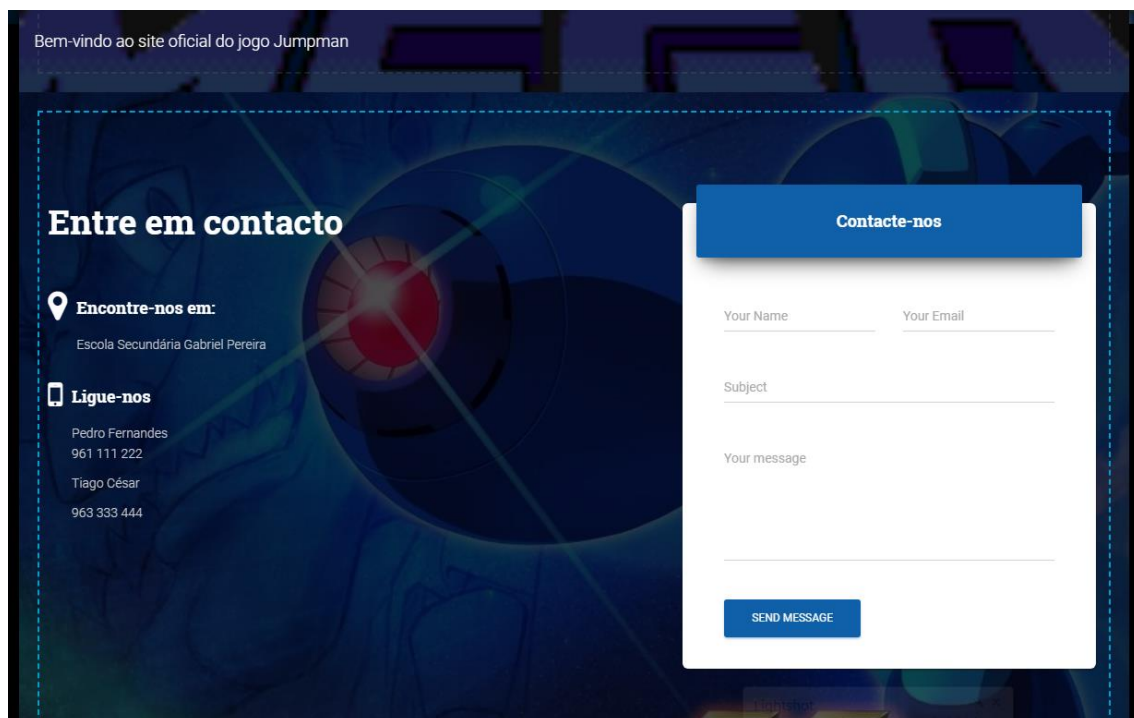


Fig.9 - Página Inicial (parte contactos).

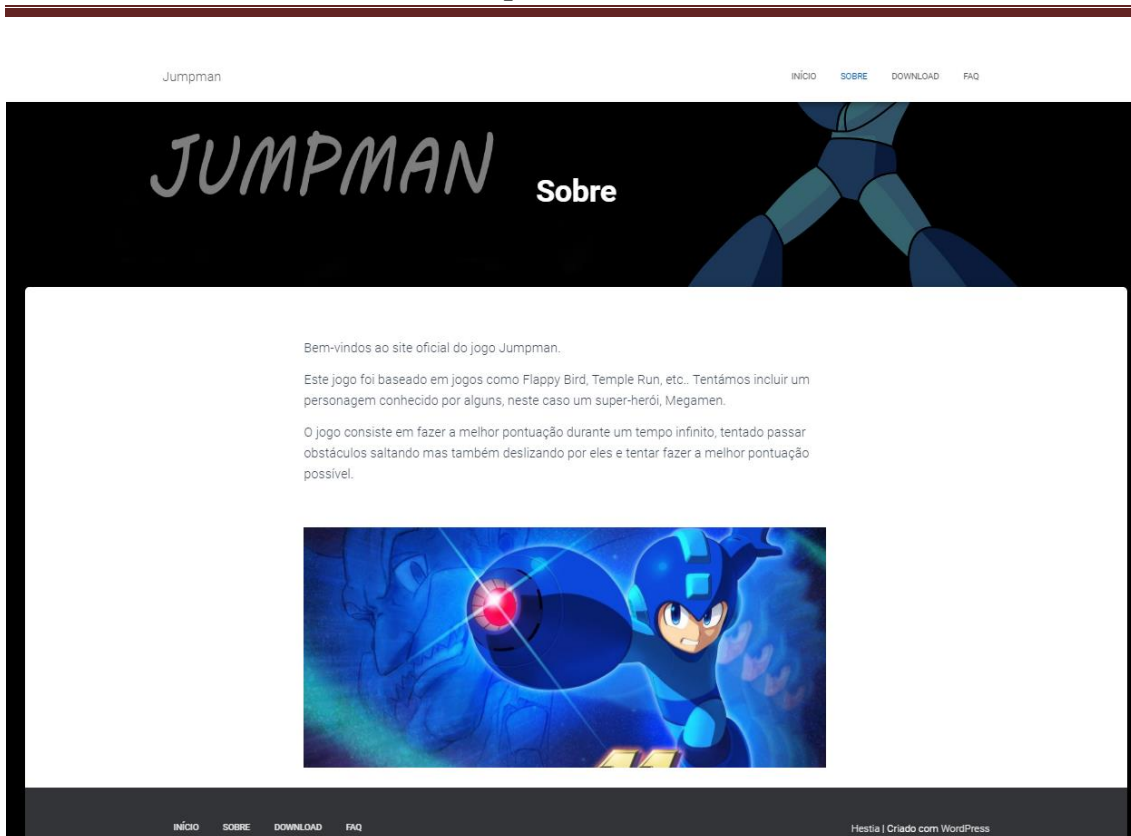


Fig.10 - Página Sobre.

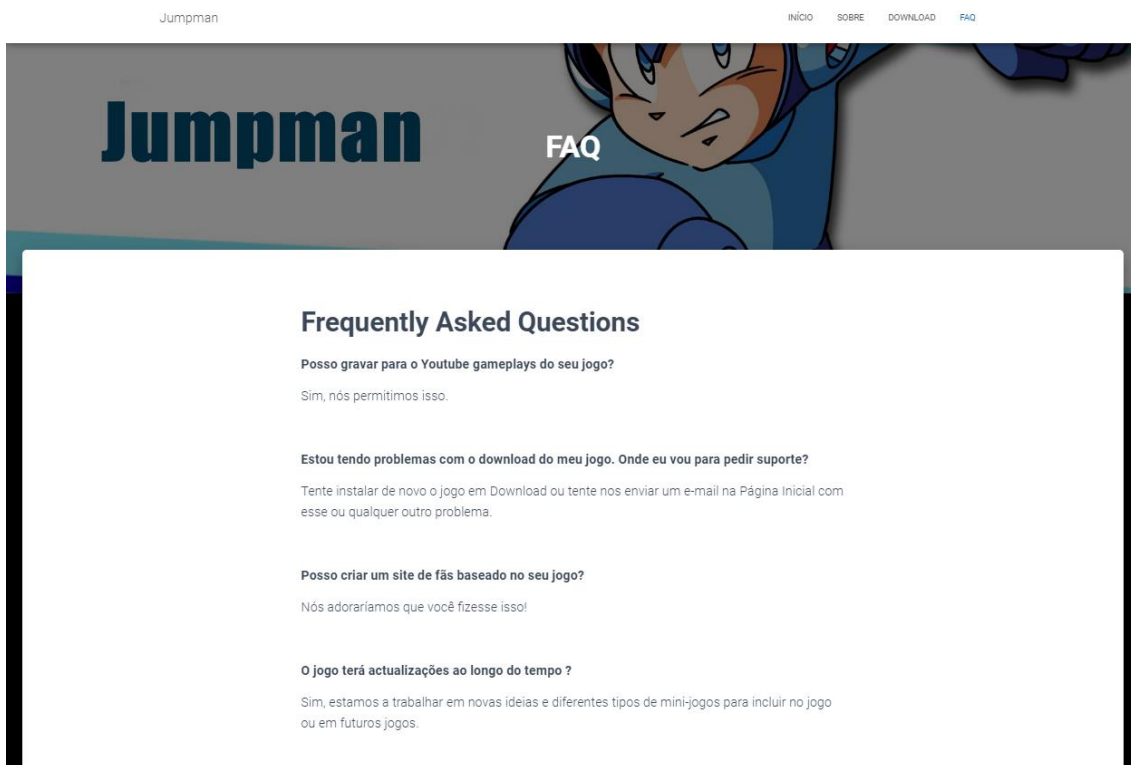


Fig.11 - Página FAQ.

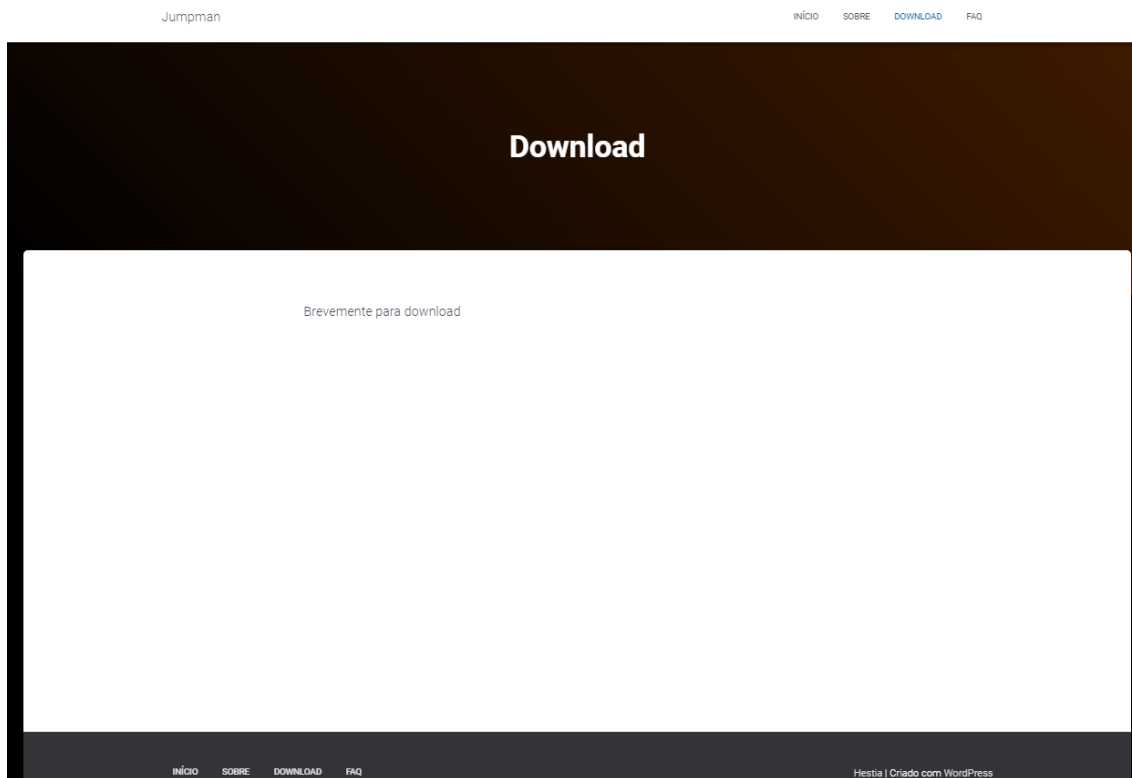


Fig.12 – Pagina de Download.