

Janeiro de 2014

EpicWare

Social Game



EPIC WARE

Índice

Diagrama de Use-Case	3
Logical view	4
Modelo de dados	6
Estruturas de dados	6
Módulo de navegação e visualização 3D	12
Módulo de IA.....	16
Módulo de website	18
Infraestrutura computacional	22

Introdução

O projeto que aqui se explica foi desenvolvido por uma empresa fictícia – EpicWare – no âmbito da unidade curricular de Laboratório de Projeto 5 (LAPR5) do quinto semestre da Licenciatura em Engenharia Informática (LEI) do Instituto Superior de Engenharia do Porto (ISEP).

O desafio proposto consistia em desenvolver uma aplicação assente em quatro diferentes módulos – navegação e visualização 3D, inteligência artificial (IA), site e infraestrutura computacional. No primeiro pretendia-se que fosse implementado em OpenGL (com auxílio de GLUT) uma visualização a três dimensões do grafo e que também fosse possibilitada a navegação através do mesmo além de, também, permitir jogar três minijogos (enforcado, jogo do galo e labirinto). No segundo módulo era pedido para implementar com recurso à linguagem PROLOG implementar a lógica dos minijogos e também procurar caminhos entre diferentes utilizadores (mais curto ou mais pesado) e descobrir e/ou sugerir amizades com base no número de tags em comum. O terceiro ponto consiste num website onde é possível consultar estatísticas e gerir a informação dos utilizadores. O último aspecto acima abordado consiste em utilizar uma máquina Linux para monitorizar diversos aspectos (desde o número de downloads da aplicação até ao espaço em disco disponível).

Visão geral do sistema

Diagrama de Use-Case

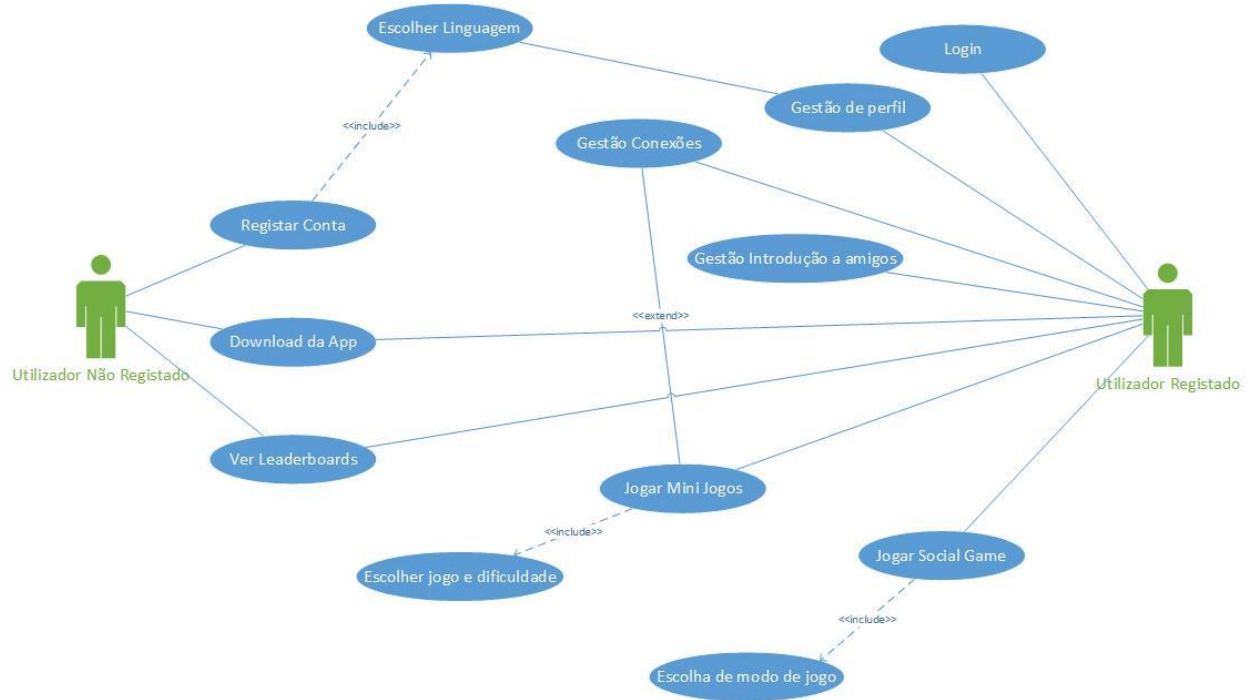


Figura 1 - Diagrama de Use-Case proposto pela nossa empresa

Como se pode ver na Figura 1 optamos por diferenciar dois tipos de utilizadores, o utilizador registado e não registado, sendo que o primeiro tem acesso a todas as funcionalidades da aplicação, enquanto que o segundo se encontra limitado nas suas acções – apenas pode observar o desenvolvimento do jogo, não o podendo jogar.

Logical view

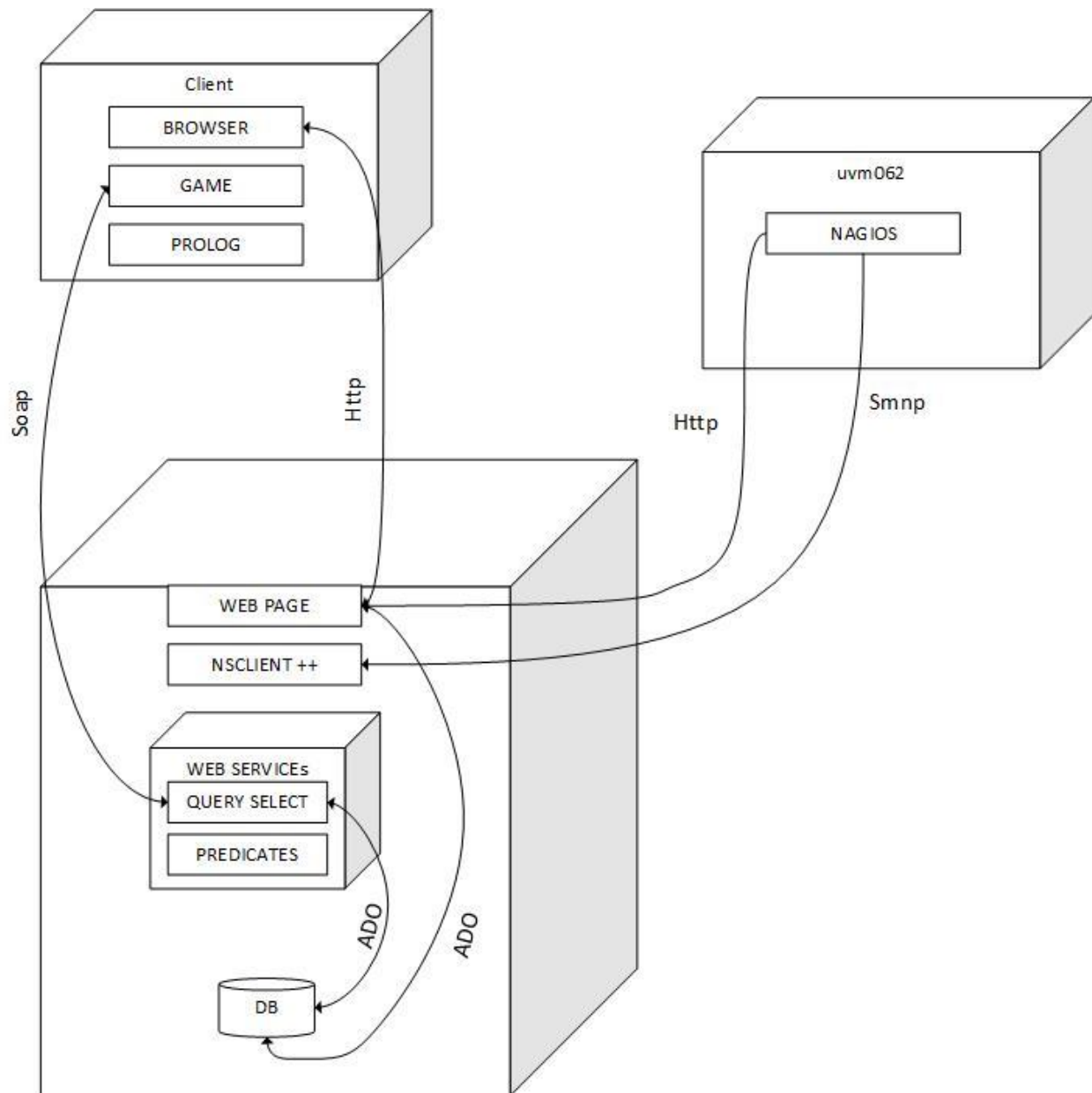


Figura 2 – Deployment view

Como é possível ver na Figura 2 existem três estruturas fundamentais, a máquina do cliente que contém a aplicação 3D do *SocialGame* instalada e também um *browser* para interagir com o *website* criado via HTTP, a máquina wvm074 onde está armazenada a base de dados do jogo (*SQLExpress*), o *website* do *SocialGame* e o webservice que o módulo de 3D utiliza para obter a

informação que necessita. Por último temos a máquina uvm062 onde se encontra instalado o NAGIOS responsável pela monitorização da wvm074 via NSClient++.

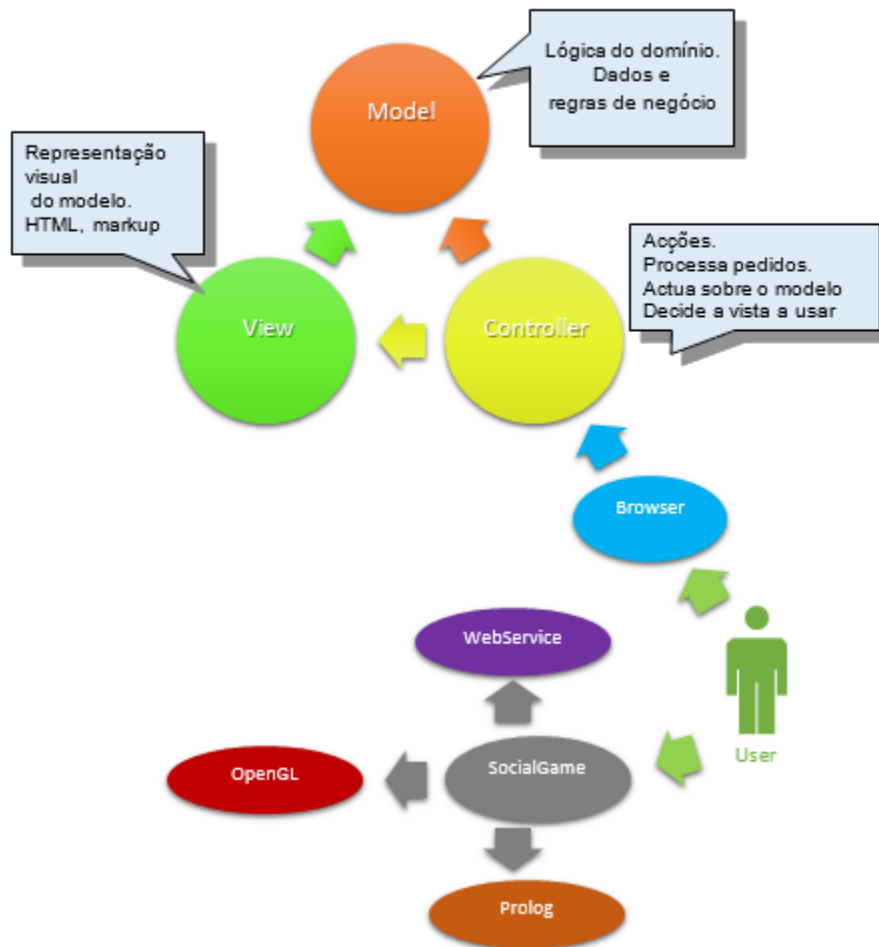


Figura 3 - Esquema simplificado do funcionamento da aplicação

Como se pode ver acima (Figura 3) o utilizador pode interagir com a aplicação de duas formas diferentes, quer através de um website (utilizando para isso o seu browser) ou através de uma aplicação 3D instalada na sua máquina (o SocialGame). No esquema é possível ver que o padrão utilizado no website foi MVC e que existem várias interações no núcleo do programa (o jogo acede à base de dados através de um webservice, utiliza de Prolog para implementar o módulo de inteligência artificial é utilizado OpenGL para a interface gráfica da aplicação). No entanto, não nos alongaremos sobre esse assunto nesta primeira abordagem ao problema, tal será feito mais abaixo quando voltarmos ao assunto de forma mais detalhada.

Modelo de dados

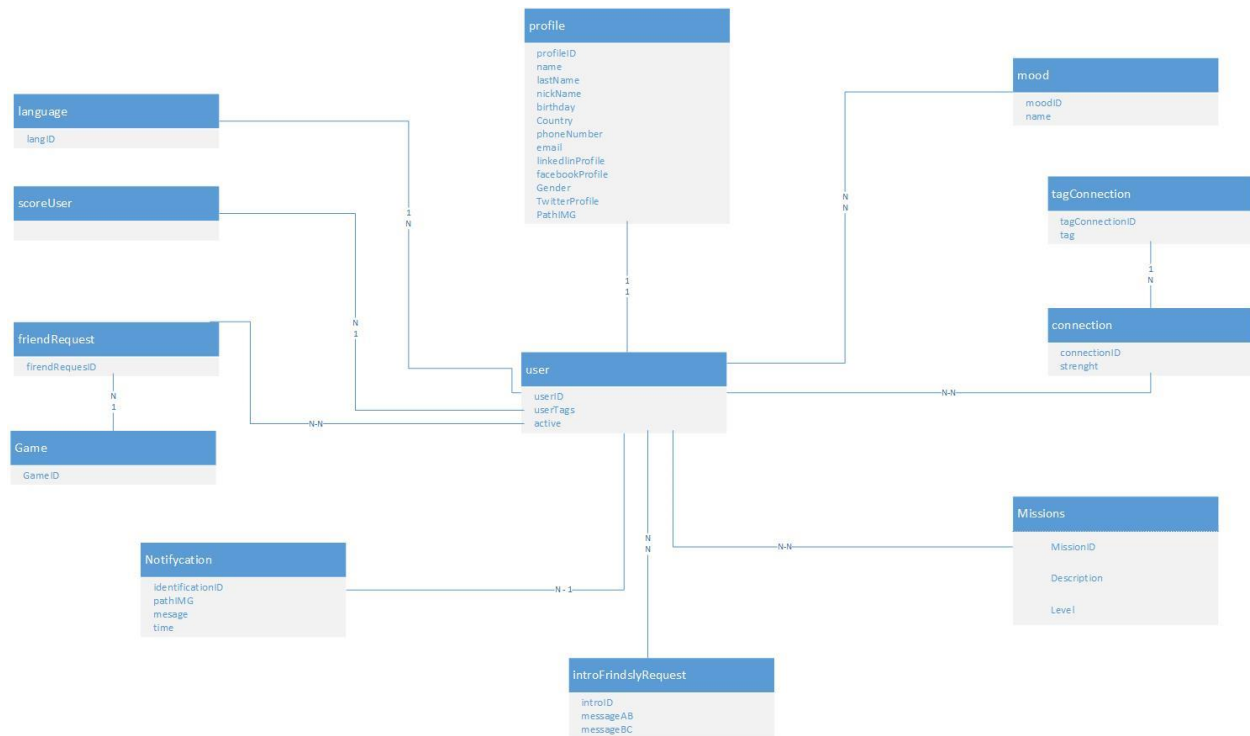


Figura 4 - Modelo de dados

Na Figura 4 acima disposta é possível observar o modelo de dados sugerido pela empresa para solucionar o problema proposto.

Estruturas de dados

Na aplicação desenvolvida foram utilizadas várias estruturas de dados. Nesta apresentação gostaríamos de salientar cinco: pickingPesquisa, Camera e Estado, Login e modelo.

pickingPesquisa: Armazena as coordenadas dos objetos que podem ser clicados (note-se que apenas para objetos 2D).

Camera: Controla a posição da câmera .

Estado: Guarda toda a informação sobre o estado atual do jogo.

Login: Guarda a informação sobre o utilizador que fez o login e também do picking de login.

modelo: Guarda a informação sobre a posição da iluminação do jogo e as escalas aplicadas.

Divisão de tarefas e desenvolvimento

Para elaborar a atividade proposta elaboramos uma divisão de tarefas que procurasse ir de encontro aos “pontos fortes” de cada membro mas de forma a que todos os membros tenham, pelo menos, uma tarefa de cada um dos diferentes módulos. No entanto, existem tarefas como, por exemplo, as de planeamento que exigiram uma cooperação de todos os membros do grupo.

Decidimos dividir o nosso grupo em 3 mini grupos para ser mais ágil o trabalho e haver menos hipóteses de falhas visto que eram dois membros por cada mini grupo, desses mini grupos pode-se presumir que os commits que ambos os membros fizeram foram feitos pelos dois, visto que baseamo-nos em pair-programming.

Por último, gostaríamos de realçar que esta não foi a divisão de tarefas inicial, uma vez que houveram alterações tanto nas tarefas como permutas nas pessoas que ficaram responsáveis pela sua realização. A divisão de tarefas ficou, portanto, como a seguir se demonstra (ver Tabela 2).

	TAREFA	LUIS	MARCIO	RICARDO BRA	VÍTOR	DIOGO	RICARDO BAR
Repositório	Criar repositório			X			
	Criar Issues	X	X	X	X	X	X
Documentação	Arquitetura de Sistema	X	X	X	X	X	X
	Estruturas de dados	X	X	X	X	X	X
	Modelo de domínio	X	X	X	X	X	X
	Use Case	X	X	X	X	X	X
	Divisão de tarefas	X	X	X	X	X	X
Social Game	Sistema de Login	X				X	
	Menu		X	X			
	Picking grafo 3D	X				X	
	Representar informações de jogadores	X				X	
	Minimapa				X		X
	Desenhar nós	X				X	
	Desenhar Mundo	X				X	
	Desenhar Ligações de nós	X				X	
	Programar camara	X				X	
	Criar Iluminação	X				X	
	Estados de Humor		X	X			
	Deteção de colisões	X				X	
	Iluminar caminhos				X		X

	Notificações		X	X			
	Jogo do galo		X	X			
	Labirinto	X				X	
	Enforcado				X		X
	Música de Fundo		X	X			
	Efeitos Sonoros		X	X			
	BillBoard				X		X
	Implementação de Missões						
	Pedido de amizade no jogo	X				X	
	Criação do executável final						
	Criar SQL inicial da BD Local			X	X		
	Divisão entre modo online e Offline	X				X	
	Métodos de resposta automática no modo offline (Ex: Pedido de amizade Introdução e Aceitar Pedido)						
	Internacionalização				X	X	
	Pesquisa de users pelo nome	X				X	
	Abrir mini-jogos numa janela diferente (abrir o .exe dos mini-jogos quando necessario)						X
	Inserir estados de humor sobre o nó (para já fica só texto a dizer o estado)	X				X	
	Localização do software para abrir diferentes linguagens.						
	Animações	X	X	X	X	X	X
IA	Pesquisar todos os amigos	X				X	
	Determinar tamanho até ao 3º nível				X		X
	Cálculo de amigos com X tags iguais		X	X			
	Sugestão de conexões tendo por base tags e conexões até 3º nível				X		X
	Caminho mais forte				X		X

	Caminho mais curto				X		X
NAGIOS	Configurar Nsclient na WVM				X		X
	Limpar NAGIOS de ASIST			X			
	Controle WVM por HTTP			X			
	Verificar espaço em disco			X			
	Taxa de utilização de CPU			X			
	Controlar trafego da interface de rede					X	
	Número de utilizadores registados		X				
	Taxa de downloads do jogo	X					
Website	Criação do modelo e persistência		X	X			
	Criar perfil		X	X			
	Editar perfil		X	X			
	Remover conta (*)		X	X			
	Pedido de Amizade		X	X			
	Visualizar Pedidos de amizade		X	X			
	Responder a pedidos de amizade		X	X			
	Catalogar ligações (Tag/Força)		X	X			
	Visualizar Sugestões de amigos		X	X			
	Calculo cota perfil	X				X	
	Calculo dimensão de perfil				X		X
	Pedido de introdução a amigo		X	X			
	Analisar pedidos de introdução		X	X			
	Criar introdução		X	X			
	Visualizar introduções sugeridas		X	X			
	Responder a introdução		X	X			
	Ver Notificações		X	X			
	Funcionalidade de pesquisa de perfis		X	X			
	Download do jogo		X	X			

	Página de administrador		X	X			
	Semântica entre Tag's						
	ScoreBoards		X	X			
	Script SQL de inicialização			X	X		
	Fornecer Dados a utilizadores não autenticados: LeaderBoard; Número de utilizadores; TagCloud – Todos Utilizadores; TagCloud – Conexões		X	X			
	Sugestão de amigos a utilizador		X	X			
	Utilizador ver tamanho da sua rede (Até 3º nível)		X	X			
	TagCloud do próprio Utilizador		X	X			
	TagCloud das relações do utilizador		X	X			
	Internacionalização	X	X	X	X	X	X
Valorizações	Conectar ao Facebook		X				
	Desconectar do Facebook						
	Comentar perfil de utilizador						
	Enviar Mensagem a utilizador						
	Conectar ao Twitter						
	Desconectar do Twitter						
	Controlar WVM por https		X	X			
WebService	Vários métodos necessários	X	X	X	X	X	X
Apresentação	Elaborar apresentação	X	X	X	X	X	X

A vermelho estão representadas as tarefas que estavam pensadas mas que por motivos de falta de tempo ou por escolhas internas no grupo não foram realizadas.

Funcionalidades implementadas

Módulo de navegação e visualização 3D

SocialGame

Inicialmente implementou-se um menu de acesso à aplicação (ver **Figura x**). Dessa forma, mesmo que um utilizador tenha a aplicação instalada no seu computador, vai precisar dos seus dados de acesso ao website para poder jogar.

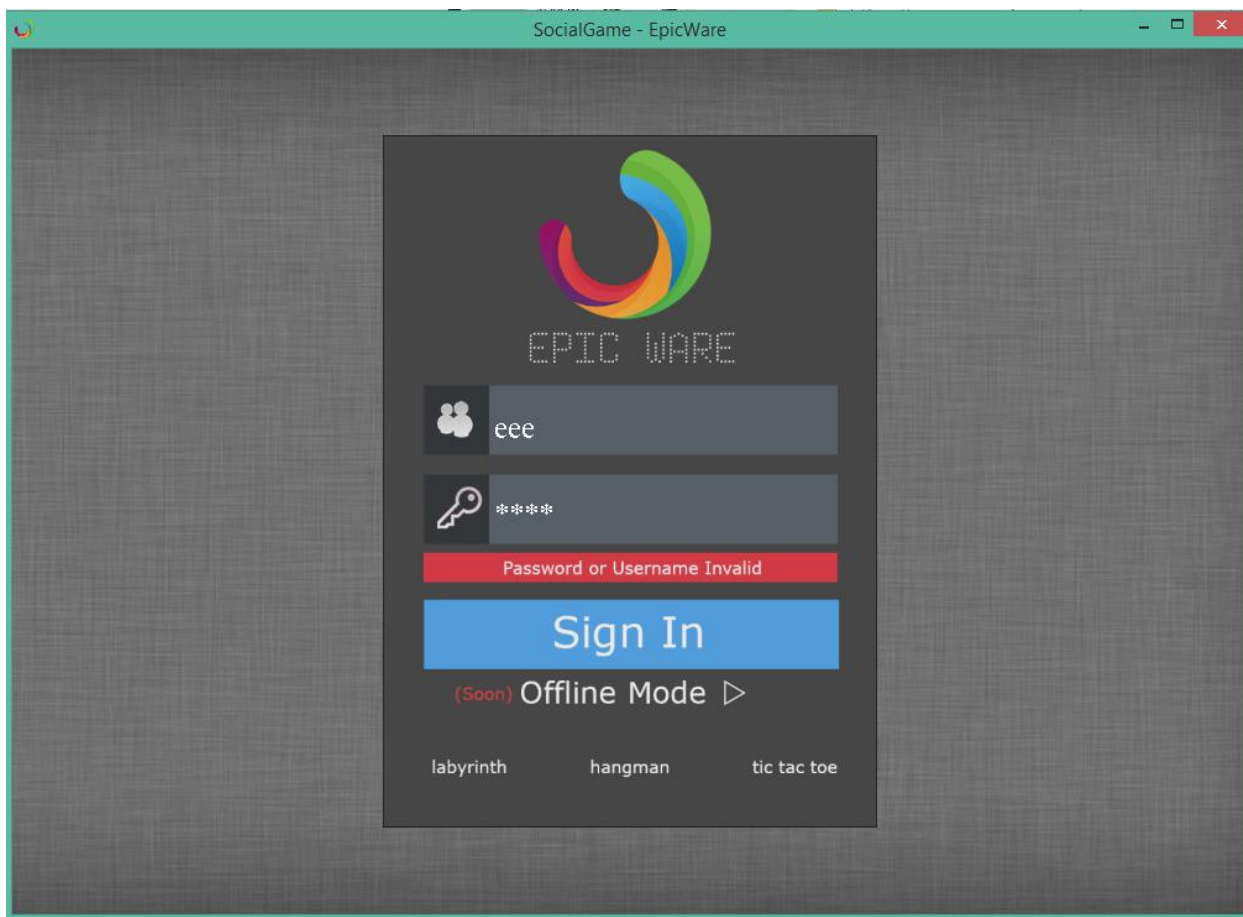


Figura 5 - Menu de login da aplicação

Após o login bem sucedido, terá acesso à vista da sua rede até ao terceiro grau (ver Figura 5), podendo distinguir aqueles que são os seus amigos (ligações a vermelho) dos que são amigos dos seus amigos (ligações cinzentas).

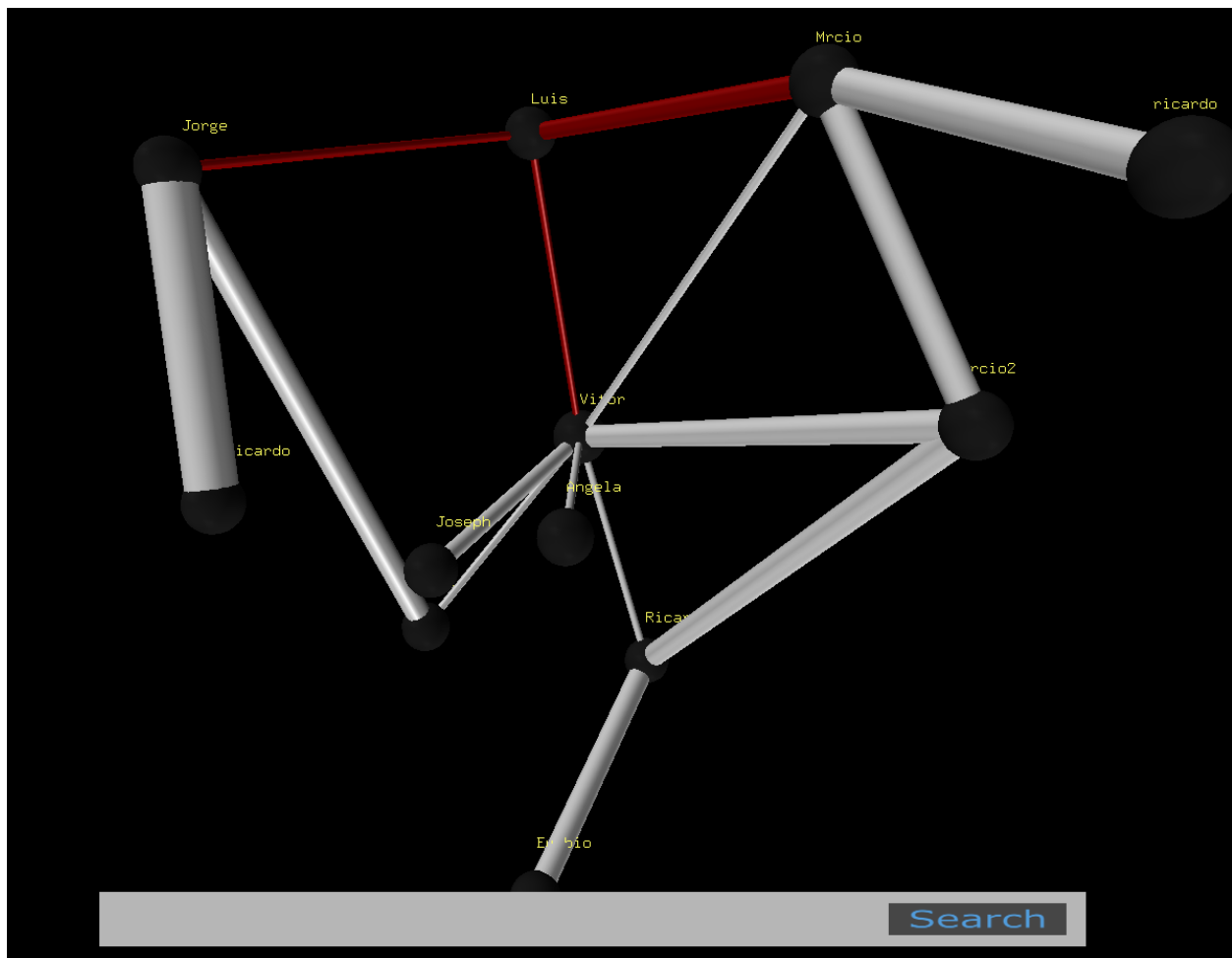


Figura 6 - Vista do grafo da rede social até ao terceiro nível.

Clicando num nó, através de picking 3D aparece a informação disponível do utilizador (nome, estado de humor e nível em relação ao utilizador que fez login).

Mini jogos

Labirinto: O jogo inicia num menu com 4 opções (Figura 7) na qual se utilizou picking 2D para identificar qual a opção selecionada pelo utilizador. Depois de escolhida a opção, é carregado de forma dinâmica (é carregado um ficheiro de texto externo) com o formato do labirinto. O conteúdo deste ficheiro é carregado para uma matriz que será usada para desenhar os vários “quadrados” que unidos formam o labirinto (Figura 8).

Este jogo importa também texturas que são aplicadas aos quadrados que formam o labirinto criando assim uma interface mais apelativa para o utilizador.

Em último lugar, as colisões estão implementadas através das teclas, isto é, são calculadas com base na posição atual do jogador. Se a próxima posição estiver dentro do caminho valido é permitida a jogada, caso contrário será impedido o movimento.

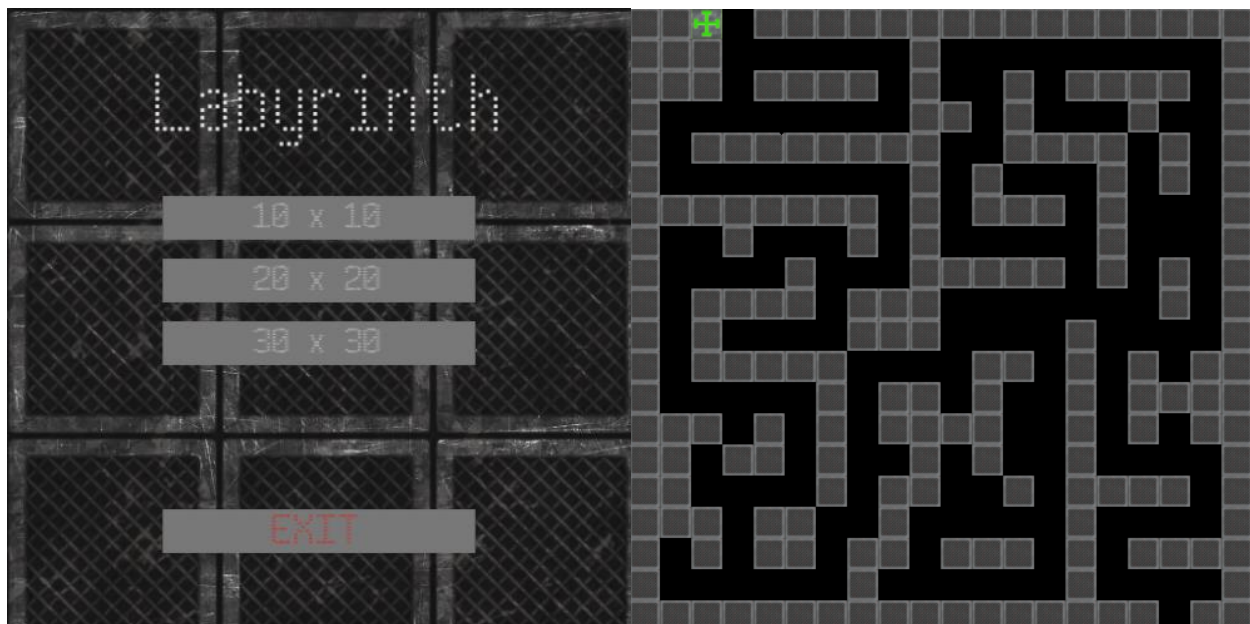


Figura 7 e 8 - Menu inicial e ecrã de jogo do labirinto

Jogo da força: Neste jogo apenas foi utilizado picking em 2D. Inicialmente, quando o jogo arranca são desenhados botões com as categorias passíveis de ser seleccionadas (note-se que são polígonos com texto sobreposto) – ver Figura 9. Uma vez seleccionada uma categoria é carregada uma palavra e é desenhada no ecrã a estrutura da forca (ainda sem o enforcado) – feita através de polígonos com translações de forma a “encaixarem” corretamente uns nos outros –, é desenhado o teclado – 26 polígonos com uma textura com as letras aplicada sobre eles – e é desenhado o texto com a categoria e a palavra escondida. Sempre que o utilizador clica numa letra é validado o palpite e a letra desaparece. Caso o utilizador acerte no palpite é colocada na(s) posição(ões) correta(s) a letra correspondente. Caso contrário é adicionado mais um membro ao enforcado (ver figura 10).

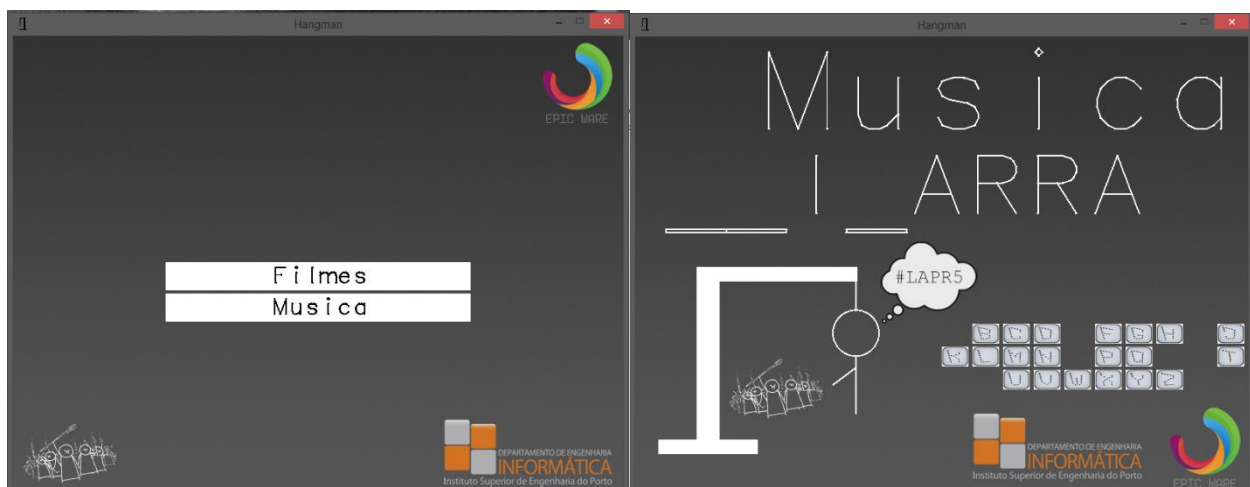


Figura 9 e 10 - Jogo do enforcado (menu de seleção de categoria e jogo).

Jogo do galo: O jogo do galo usa uma matriz para ser desenhado, essa matriz contém 3 colunas por 3 linhas, em cada uma dessas estamos a guardar qual é a textura e jogada que está lá guardada (Figura 11).

Ao início do programa carregamos todas as texturas para desenhar quer o menu, quer o jogo, assim que o jogador e computador jogam ele actualiza o quadrado em que eles jogam com a textura devida e com a jogada devida (Figura 12).

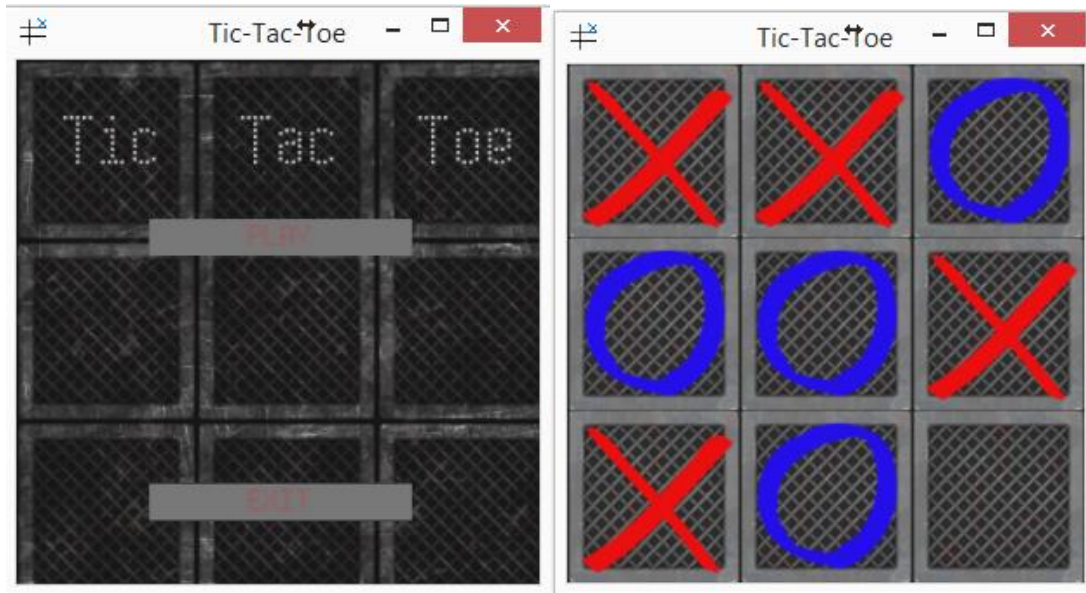


Figura 11 e 12 - Jogo do galo (menu inicial e ecrã de jogo).

Módulo de IA

Determinar o tamanho da rede de um utilizador até ao 3º nível

É calculado o tamanho da rede sem pessoas repetidas até ao terceiro nível. É carregada a base de conhecimento pelo webservice de um utilizador e são percorridos todos os utilizadores anexados para ir buscar as suas conexões

Obter os amigos que tenham “X” tags em comum

Para obter os amigos com um determinado número de tags em comum com outro é feita uma pesquisa pelos amigos do utilizador, de seguida são contabilizadas as tags que tem em comum com cada um deles e os que preencherem o requisito serão adicionados à lista de resultados.

Determinar o caminho mais curto e o mais pesado

Para encontrar o caminho mais pesado foi adaptado o método de pesquisa A*. Para o nosso caso em particular, em vez de guardar o caminho cujo custo é menor, guarda aquele cujo custo é maior, com um ligeiro defeito – o caminho mais pesado está sempre na última resposta do predicado. Para implementar esta funcionalidade são usados os ids do utilizador de origem e de destino e é retornado o peso e a lista de utilizadores com o caminho percorrido.

Para determinar o caminho mais curto aplicou-se uma simples pesquisa em largura. Para esse efeito são usados os ids de dois utilizadores diferentes (o de origem e o de destino).

Minijogos

Labirinto: Para pesquisar o caminho mais curto até ao final do labirinto foi utilizado um algoritmo de pesquisa em largura. Preferimos esta em relação à pesquisa em profundidade porque, apesar de a segunda ser mais rápida, não oferece garantias de devolução do caminho mais curto.

Jogo do galo: Optamos por utilizar o algoritmo de pesquisa “Minimax” para decidir qual a jogada a efectuar pelo computador. Para validar as jogadas do utilizador é utilizado um predicado que verifica se o local que o utilizador escolheu não foi já utilizado e também para verificar se ganhou/perdeu o jogo.

Enforcado: É utilizada inteligência artificial para escolher uma palavra aleatória dentro da categoria escolhida previamente pelo utilizador. Posteriormente é validado o palpite do utilizador (verifica em que posições se encontra uma letra passada – caso não exista retorna uma lista vazia) e, caso ganhe o jogo é informado disso mesmo.

Integração com C++

Devido à falta de tempo, apenas os predicados de pesquisa de caminhos foram integrados com a aplicação. No entanto, tal como referido acima, houveram outros predicados que ficaram operacionais e prontos a ser integrados com o resto do projeto.

Módulo de website

Descrição da interface IEpicService

Para facilitar a comunicação entre a base de dados e a aplicação 3D foi implementado um *webService* em C# que está explicado na tabela abaixo.

Assinatura do método	Descrição
<code>int getNumberUsers();</code>	Devolve o número total de utilizadores registados na aplicação.
<code>int getNumberDownloads();</code>	Devolve o número total de <i>downloads</i> da aplicação já feitos.
<code>User getUserById(int id, string user, string pass);</code>	Devolve toda a informação do utilizador cujo id é igual ao passado por parâmetro.
<code>IList<int> getUserFriendsByUserId(int id, string user, string pass);</code>	Devolve uma lista com os ids dos amigos do utilizador cujo id é o passado por parâmetro.
<code>User getUserByAutetication(string user, string password);</code>	Devolve a informação do utilizador atualmente autenticado.
<code>IList<Word> GetEveryWord();</code>	Devolve uma lista com todas as palavras existentes na base de dados (note-se que uma variável do tipo Word contém uma palavra, a sua categoria e a respetiva língua).
<code>IList<int> getAllUsersID();</code>	Devolve uma lista com os ids de todos os utilizadores .
<code>int getConnectioStrenght(int id1, int id2);</code>	Devolve a força da ligação entre dois utilizadores cujos ids são passados por parâmetro.

O website foi implementado através da Framework 4.5 .NET (ASP.NET e C#). O padrão utilizado para o implementar foi o MVC4¹ e também recorremos a funcionalidades fornecidas por JQuery, CSS (Bootstrap) e AJAX – mais concretamente ASP.NET AJAX.

¹ Model View Controller

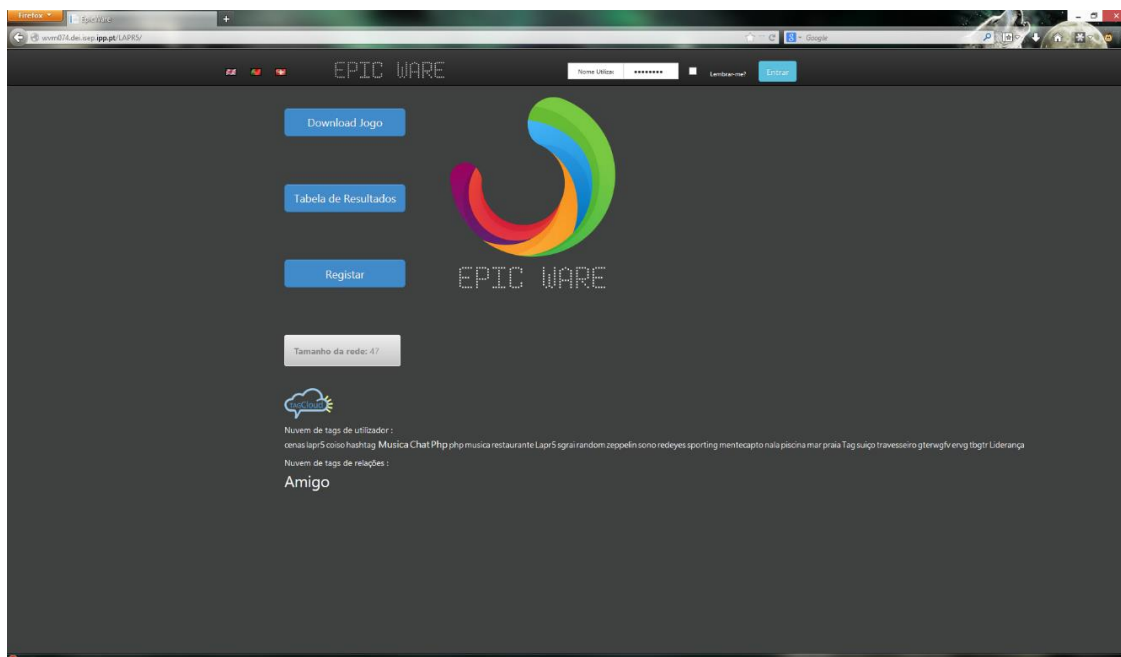


Figura 13 - Menu inicial do website.

Como se pode observar na Figura 13 na página principal da aplicação (index) é possível fazer download do jogo, consultar a tabela de resultados, registar-se na rede social e (caso ainda não o tenha feito) fazer login no website. Nesta página também é possível verificar as tag cloud gerais.

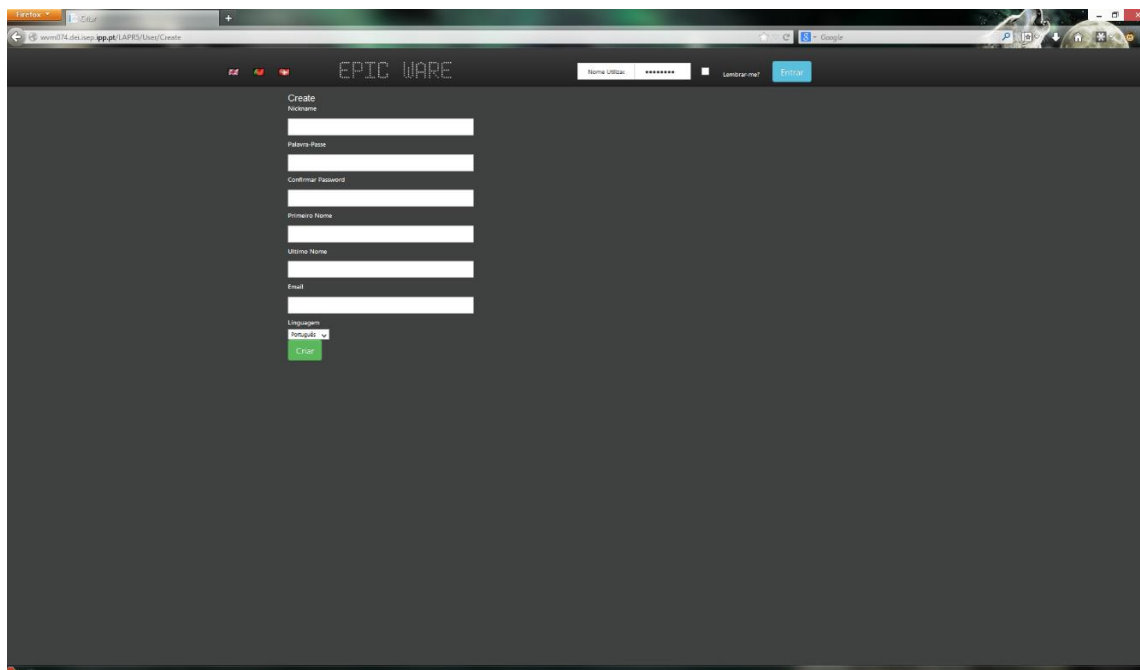


Figura 14 - View de criação de utilizador.

Na Figura 14 é possível ver uma view para a criação de um novo utilizador do site. Na imagem seguinte (Figura 15) podemos observar a página para o preenchimento dos campos do perfil.

EPIC WARE

Nome | Sobrenome | Tag

Notificação 0

Ingresso resistência

Perfil

Criar Perfil

Data de Nascimento (YYYYMMDD)

País

Nome

Número de Telefone

Link do Perfil do LinkedIn

Link do Perfil do Facebook

Link do Perfil do Twitter

Email

Pesquisar

Nenhum ficheiro seleccionado

Figura 15 - View de inserção de dados no perfil de utilizador.

Uma vez feito o login, a página que aparecerá terá disponível vários botões para ver todos os pedidos que o utilizador tem pendentes (Figura 16), além do mais, aparece também um botão a informar sobre as notificações que este ainda não viu. Aparece também um botão com a foto de perfil do utilizador com o atalho para a página com informação pessoal do utilizador (ver Figura 17)

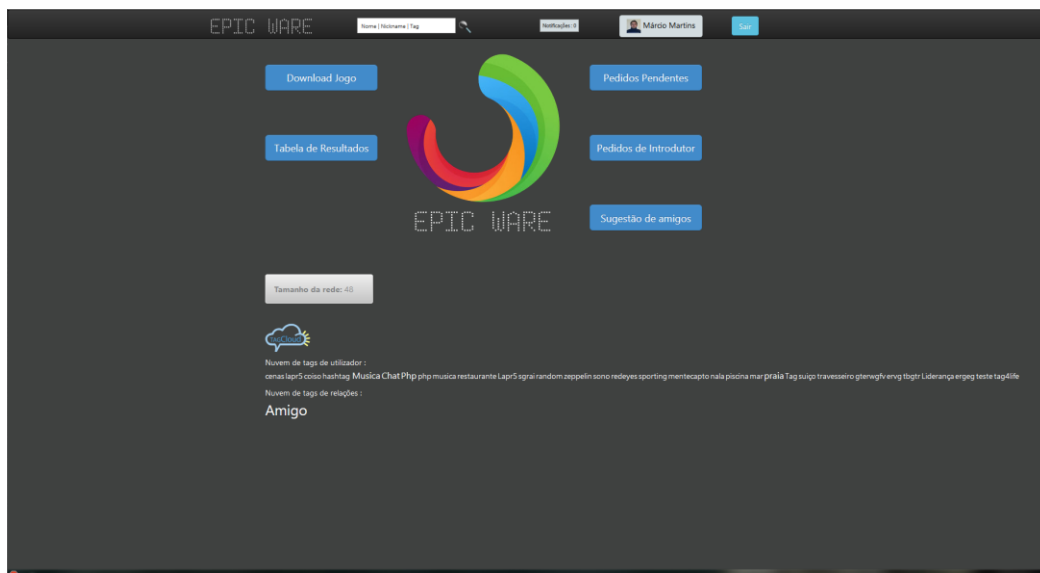


Figura 16 - View da página inicial do utilizador

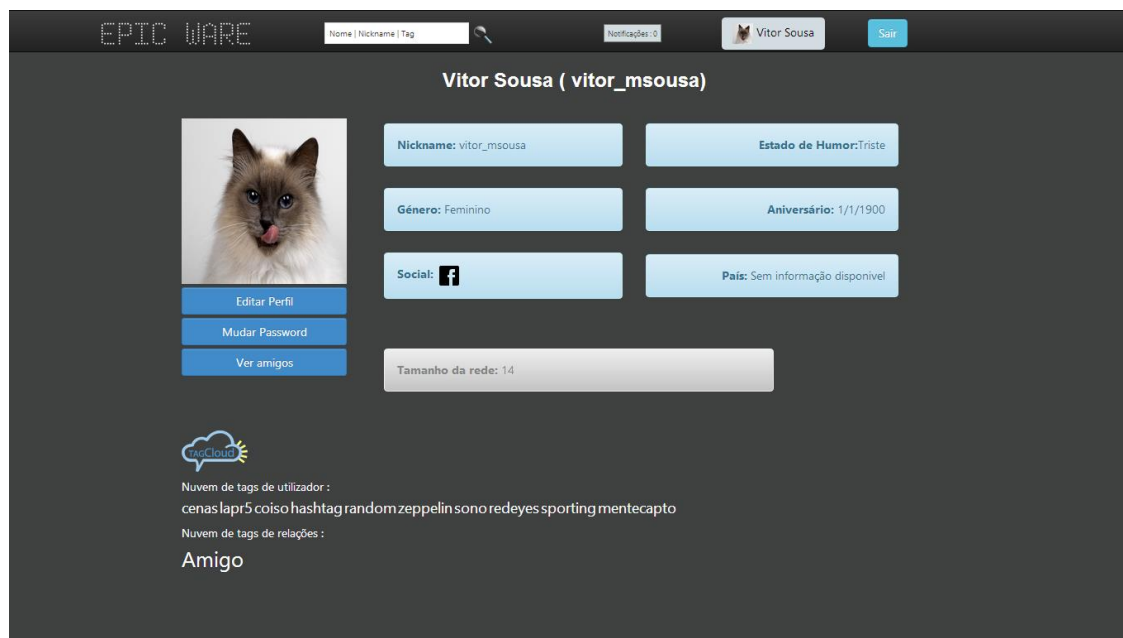


Figura 17 - View da página pessoal do utilizador

Nesta página (Figura 17), além da informação pessoal, é também possível gerir o perfil do utilizador (através dos botões de editar perfil e mudar a password). Na opção de editar perfil, entre outras coisas, é possível adicionar ou remover tags ao perfil, editar/acrescentar informação e a foto de perfil (ver Figura 18).

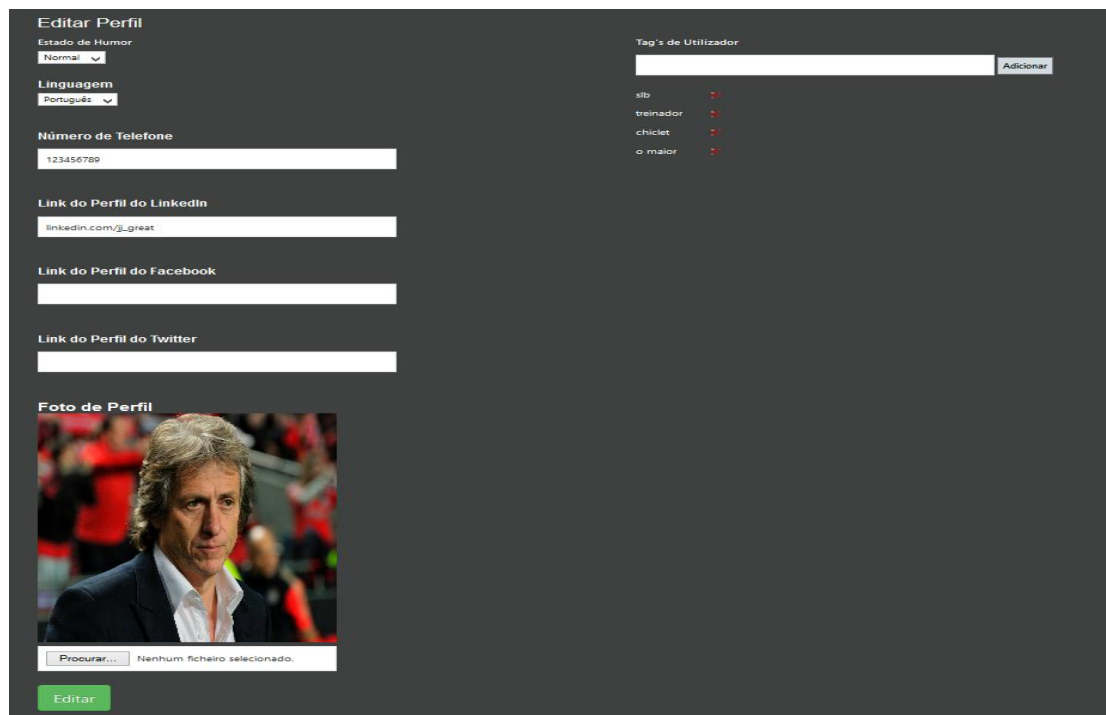


Figura 8 - View de alteração de dados pessoais do utilizador

A parte de login com o facebook começou a ser implementada mas não foi concluída.

Infraestrutura computacional

NAGIOS

Nagios foi elaborado de acordo como era pedido no enunciado, havendo alguns problemas com o controle de tráfego na interface de rede mas que foram facilmente sanados.

Fizemos também a valorização de https, tendo apenas o problema de o nosso certificado não ter uma assinatura por uma entidade certificadora.

Os scripts de Nagios e documentação encontram-se em anexo a este relatório ou na pasta de documentação do projeto.

Conclusão

Após estas semanas de desenvolvimento do desafio proposto foi possível concluir uma parte significativa dos requisitos propostos tendo até conseguido fazer algumas valorizações.

Foram semanas enriquecedoras e que nos permitiram evoluir imenso enquanto equipa e ajudou-nos a perceber a importância de boas práticas de comunicação e documentação ao longo da realização de um projeto e também a aprofundar algo que vínhamos a explorar anteriormente: o *pair-programming*, além claro, de nos ajudar a perceber quais as nossas limitações e ajudou-nos também a perceber qual as áreas com que mais nos identificamos neste enorme mundo que é o da engenharia de software.

Gostaríamos também de salientar a importância do desenvolvimento desta aplicação na nossa compreensão da importância da estruturação das equipas de desenvolvimento e na gestão do tempo, uma vez que, apesar de sermos uma equipa numerosa, esta envolvia empenho e um rigoroso aproveitamento do tempo que dispomos.

Por último, gostaríamos também de agradecer a todos os professores sempre disponíveis a tirar dúvidas e que foram também parte importante no sucesso desta aplicação.

Bibliografia

<http://www.lighthouse3d.com/tutorials/glut-tutorial/preparing-the-window-for-a-reshape/>
<http://stackoverflow.com/questions/5467218/opengl-2d-hud-over-3d>
<http://www.dei.isep.ipp.pt/~matos/cg/docs/manual/glutBitmapCharacter.3GLUT.html>
<http://stackoverflow.com/questions/14370/glut-pop-up-menus>
<http://electricbacon.wordpress.com/2010/09/08/calc-example-using-swi-prolog-and-c/>
<http://electricbacon.wordpress.com/2010/09/08/setting-up-swi-prolog-with-c/>
<http://www.swi-prolog.org/pldoc/package/pl2cpp.html>
<http://www.swi-prolog.org/pldoc/package/pl2cpp.html#class:PIFrame>
<http://cboard.cprogramming.com/cplusplus-programming/124728-swi-prolog-embedded-application-vs2008-cplusplus.html>
<http://open.gl/textures>
http://nccastaff.bournemouth.ac.uk/jmacey/RobTheBloke/www/opengl_programming.html
<http://www.glprogramming.com/red/chapter13.html>
<http://content.gpwiki.org/index.php/OpenGL:Tutorials:Picking>
<http://jerome.jouvie.free.fr/opengl-tutorials/Tutorial27.php>
<http://www.swi-prolog.org/pldoc/man?section=pltail>
<http://www.microsoft.com/betaexperience/pd/SQL2012EvalCTA/enus/default.aspx>
http://paginas.fe.up.pt/~eol/IA/IA0809/APONTAMENTOS/Alunos_MiniMax.pdf
<https://app.box.com/s/a2154e1db18b400d0677>
<http://net.tutsplus.com/articles/general/team-collaboration-with-github/>
<http://www.hanselman.com/blog/GlobalizationInternationalizationAndLocalizationInASPNETMVC3JavaScriptAndjQueryPart1.aspx>
<http://www.youtube.com/watch?v=3YUzLXepfsE>
<http://www.code52.org/aspnet-internationalization/tutorial.html>
<http://ajdotnet.wordpress.com/2013/11/23/asp-net-mvc-i18n-part-2-detect-browser-settings/>

<http://ajdotnet.wordpress.com/2013/12/08/asp-net-mvc-i18n-part-4-css-styles/>

<http://ajdotnet.wordpress.com/2013/12/14/asp-net-mvc-i18n-part-5-imperative-view-localization/>

<http://ajdotnet.wordpress.com/2013/11/15/asp-net-mvc-i18n-basics/>

<http://blogs.msdn.com/b/sqlexpress/archive/2005/05/05/415084.aspx>

<http://www.cooltext.com/Logo-Design-Still-Cool-Button>

<http://forums.asp.net/t/1584359.aspx>

http://diretonocodigo.blogspot.pt/2011/07/helper-para-calendario-com-jquery-ui_24.html

<http://www.jqueryui.com/datepicker/>

<http://www.jqueryui.com/show/>

http://www.w3schools.com/js/js_functions.asp

<http://www.youtube.com/watch?v=5xYjiFjrj8>

<http://automagical.rationalmind.net/2011/03/14/basic-ajax-with-mvc3-razor/>

<http://www.youtube.com/watch?v=3v4-NQYpS7M>

<http://www.youtube.com/watch?v=0WMFixqPLg4>

<http://20fingers2brains.blogspot.pt/2013/02/social-networking-website-in-aspnet.html>

<http://nareshkamuni.blogspot.pt/2012/02/how-to-change-image-on-mouseover-and.html>

<http://www.getbootstrap.com/components/#alerts-examples>

<http://www.daniweb.com/web-development/aspnet/threads/266584/asp.net-mvc-chat>

www.asp.net/signalr/overview/signalr-20/getting-started-with-signalr-20/tutorial-getting-started-with-signalr-20

<http://www.stackoverflow.com/questions/596444/html-actionlink-as-a-button-or-an-image-not-a-link>

<http://www.tutorialrepublic.com/twitter-bootstrap-tutorial/bootstrap-dropdowns.php>

<http://www.markchouanard.com/post/25542393378/using-web-fonts-in-your-asp-net-mvc-application>

<http://www.robertpinchbeck.com/college/work/prolog/>

<http://www.cplusplus.com/forum/general/8510/>

<http://www.daniweb.com/software-development/cpp/threads/15252/sound-in-c>
<http://www.stackoverflow.com/questions/2017736/importing-models-into-a-opengl-project>
<http://www.sourceforge.net/projects/objloader/>
<http://en.serverklan.net/index.php?file=FichierDownload&typeFichier=Models&AfficheFichier=true&IDfichier=430343>
<http://forums.asp.net/t/1927823.aspx?Tag+Cloud>
<http://www.mikesdotnetting.com/Article/107/Creating-a-Tag-Cloud-using-ASP.NET-MVC-and-the-Entity-Framework>
<http://aspnet.4guysfromrolla.com/articles/102506-1.aspx>
<http://www.codecapers.com/post/aspnet-creating-custom-tag-cloud.aspx>
<http://www.afana.me/post/aspnet-mvc-internationalization.aspx>
www.code-inside.de/blog-in/2010/10/19/howto-create-and-remove-cookies-with-asp-net-mvc/
<http://www.stackoverflow.com/questions/20509299/jquery-datepicker-beforeshowday-ajax-return-array-of-unavailable-dates>
<http://technet.microsoft.com/en-us/library/cc731183.aspx>
<https://www.youtube.com/watch?v=3QEatBgURiE>
<https://www.youtube.com/watch?v=NzTYx6Yview>
<http://www.iis.net/learn/publish/using-the-ftp-service/using-ftp-over-ssl-in-iis-7>
<http://www.sanuja.com/blog/resolving-iis-403-forbidden-access-is-denied>
<http://www.stackoverflow.com/questions/14695180/online-certification-authority-cannot-select>
<http://www.devcurry.com/2012/06/aspnet-mvc-handling-exceptions-and-404.html>

Nota: Todos os endereços acima falados encontram-se online no dia 15/10/2014