

Social Graph

MyVa - 2013

1100554 – Bruno Cunha

1100592 – Hugo Dias

1101340 – Leniker Gomes

1100638 – Mário Queirós

1100677 – Tiago Queirós

Relatório final do projeto no âmbito da disciplina Laboratório
de projeto V

Social Graph

MyVa - 2013

1. *Introdução*

No âmbito do concurso criado pela Graphs4Social, serve o presente documento como relatório explicativo do projeto. Este consiste em criar um protótipo de uma solução de manipulação e visualização de grafos sociais baseadas em 3 módulos principais e nas suas interligações, a apresentar posteriormente.

É necessário indicar que, sendo um sistema protótipo, nem todos os requisitos estarão completamente funcionais, mas o trabalho já concluído demonstra que se trata apenas de uma questão temporal.

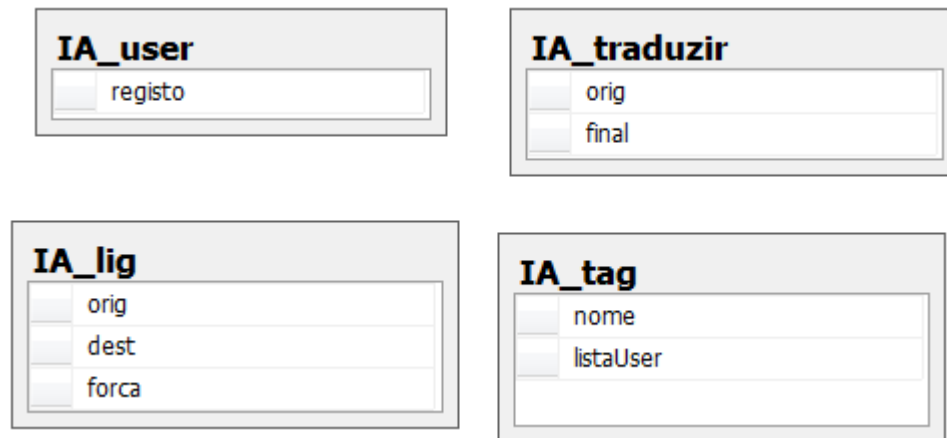
O foco do protótipo é a criação de um sistema completamente integrado, com os 3 módulos em comunicação verdadeira.

2.1 Módulo IA

O módulo de inteligência artificial (IA) tinha por como objetivo a manipulação do grafo social recorrendo a factos e predicados Prolog. Foi utilizado o Win-Prolog 4900 em todo o desenvolvimento, em conjunto com a base de dados do gandalf (mais detalhes da arquitetura no diagrama próprio).

Este módulo está concluído a 100%, ficando apenas para desenvolvimento futuro a implementação do cálculo do eco de uma mensagem (fator de valorização).

A estrutura da BD do módulo é a abaixo representada.



Análise de Complexidade temporal do Grau Médio de Separação

grauMedio(R):- (...)

findall(X,user(X),LU),

U – nº utilizadores

assert(!Users(LU)),

grauMedio(LU,V,C),

$(U*(U-1)/2)$ vezes

(...).

grauMedio([U|UR],V,C):-somaCaminhos(U,P,NC),

V^2 , sendo V o nº de vértices do grafo

grauMedio(UR,VA,RA),

(...)

Assumindo o grau média de separação como a função $g(x)$:

Estão omissas diversas atribuições e operações de impacto irrelevante dado o nível de processamento a que as operações analisadas obrigam.

O predicado somaCaminhos consiste no cálculo dos caminhos mais curtos entre um par de utilizadores, sendo V o nº de vértices do grafo a navegar.

Complexidade: $G(X)=(U*(U-1)/2)*V^2$

Neste algoritmo não se aplica o fator de caso pior e melhor, pois todos os caminhos são sempre pesquisados (não existe condição prévia de paragem).

Otimização:

A otimização que pode ser encontrada no algoritmo feito é o cálculo de apenas metade das operações feitas originalmente: inicialmente, era calculado o grau medio $U*(U-1)$ vezes, pois procurava, p .ex, o caminho de “Bruno para Sara” e posteriormente de “Sara para Bruno”. Após análise, foi possível concluir que o grafo pode ser dividido em 2 partes, sendo que o valor do GMS das 2 será exatamente igual, pelo que basta calcular uma delas, reduzindo em metade os dados processados.

2.2 *Módulo 3D*

Funcionalidades implementadas

O módulo de navegação e visualização 3D tem como objectivo apresentar de forma visual a rede social, interagindo com outros utilizadores. Na nossa aplicação é possível realizar o voo livre pelo grafo com a orientação de um mini-mapa no canto inferior esquerdo. A aplicação detecta caso a câmara esteja a colidir com o grafo. A apresentação de cada utilizador é feita através de esferas e as ligações (amigos) é feita através de cilindros a ligar as duas esferas. Ao iniciar a aplicação é possível visualizar de uma forma distinta o utilizador que fez login tendo ele e os restantes membros da rede social um avatar escolhido no módulo Site que serve para identificar o utilizador da rede social. Também é possível visualizar em cima do avatar o estado de humor do utilizador. Cada utilizador terá informação de si próprio ao passar com o rato na esfera/avatar/estado de humor (tooltip) onde se pode visualizar o seu nick, sexo e as suas tags.

O “mundo” está representado com texturas no chão, à volta do grafo (paredes) na tooltip e nos estados de humor. O posicionamento de cada utilizador no grafo é definido aleatoriamente, tendo atenção caso dois utilizadores estejam muito pertos ou a colidir, a aplicação automaticamente reposiciona os utilizadores. A sua altura no grafo é definida consoante o número de amigos (ligações), a força de ligação entre dois utilizadores é definida através da grossura das ligações entre amigos. A aplicação também tem uma música de fundo.

É possível também visualizar de forma diferente o caminho entre o utilizador autenticado e outro utilizador através de um clique no rato em cima da esfera de um utilizador.

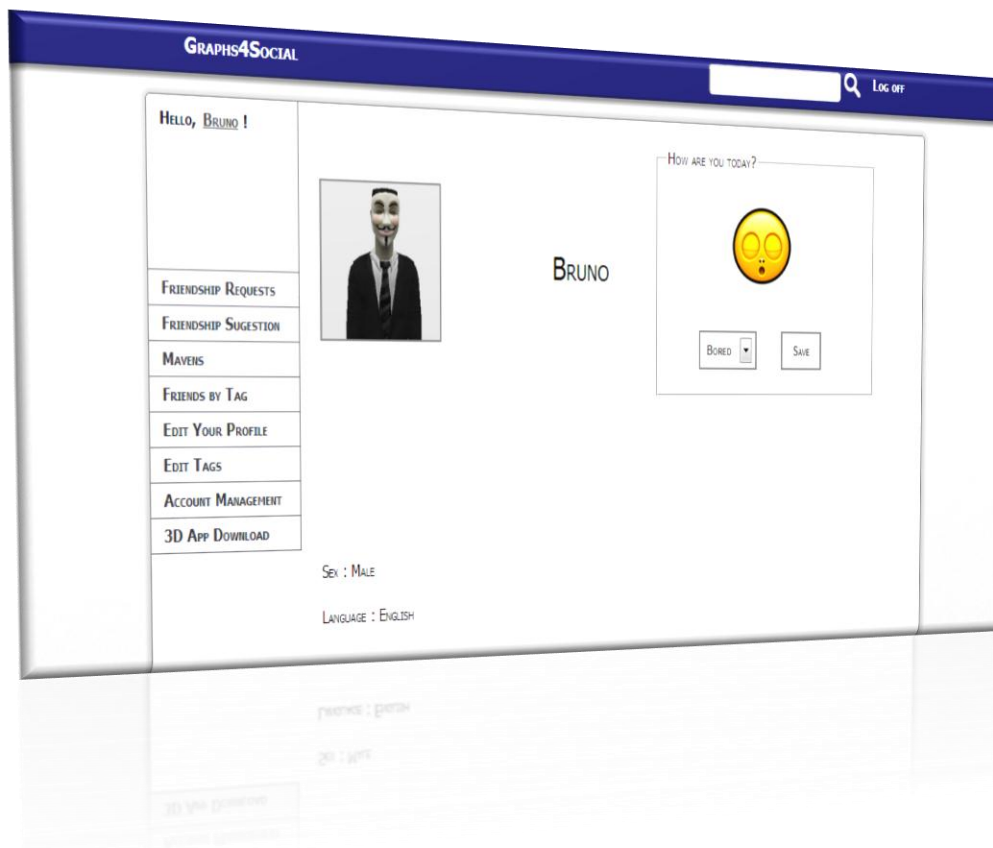
Não foi possível realizar aos nós e ligações para onde “se está a olhar” com as fontes de iluminação. Também não foi possível realizar as valorizações devido ao facto de o tempo não ter sido suficiente.

2.3 Módulo Site

Funcionalidades implementadas:

O módulo site foi construído numa base de ASP.NET com Active Record e toma um aspecto jovial e característico do fenómeno que são as redes sociais. Neste modulo é possível a uma pessoa se registar e conectar-se com outras pessoas. Para além disso, existe a possibilidade de alterar o perfil e adicionar tags, podendo estas representar os seus gostos, isto é, tags de interesse ou as suas relações com outros utilizadores, através de tags de relação.

Em termos de design, foi utilizado uma página em branco e foi construído recorrendo as funcionalidades do ASP.NET. Em termos de informação, fornecemos os dados básicos inseridos pelo utilizador, imagem, tag cloud (na qual foi utilizada uma classe externa criada por terceiros) e estatísticas da rede social do utilizador, calculadas pelo módulo de Inteligência Artificial. Também foi inserido, através da mesma referida, as sugestões de amigos e o maven de todas as tags.



3.1 *Processo de desenvolvimento*

Durante o processo tentámos aplicar a metodologia RAD “*Rapid Application Development*”. O início do projeto foi utilizado para definir os “templates” e as modelos de dados a utilizar. Na fase seguinte, criámos protótipos do sistema para verificar que alterações seriam necessárias fazer nos modelos definidos na fase inicial. Estas 2 fases consistem em várias iterações consecutivas, até atingir um sistema estável. Neste ponto, seguimos um modelo orientado a objetivos muito específicos, de forma a poder dividir e planear entre vários elementos “rápidas” tarefas que visam atingir os objetivos definidos nos testes, embora não tenha sido cumprido “à letra”, fugindo para um modelo “*code and fix*”.

Foi, no panorama geral, um método que teve as suas vantagens e serviu o seu propósito – melhorar o processo de desenvolvimento do sistema.

3.2 *Documentos em anexos*

Divisão de tarefas: Atribuição de tarefas entre cada elemento

Relatório Nagios: relatório específico do software de monitorização

Formulário de avaliação: Autoavaliação da equipa em cada componente e do trabalho

Arquiteturas e Modelo de dados: contêm uma breve descrição em cada documento