

MAC0416/5855 - Tópicos Especiais em Desenvolvimento para Web

Relatório Final - Sistema Notícias

Integrantes

Alexandre Oki Takinami - 5210532
Bruno Henrique Yoshimura - 5122944

Primeira Parte : Flickr

O Flickr é um site de gerenciamento e compartilhamento de fotos. Seu principal objetivo é ajudar as pessoas a compartilhar fotos e organizar fotos. Com a possibilidade de utilização de APIs a comunidade Web conseguiu estender seus serviços e criar novas funcionalidades como: mostrar fotos baseada em geoposicionamento em mapas interativos, mostrar imagens relacionadas a páginas dinâmicas na internet, entre outros.

Além de agregar essas funcionalidades, as APIs ajudam na popularização do serviço. É possível, por exemplo, exportar imagens do programa Picasa (do Google) para o Flickr com apenas um clique.

Assim como as novas tecnologias Web 2.0, a categorização de fotos é baseada totalmente em tags. Apesar da flexibilidade da utilização de tags, o site também permite a criação de álbuns, apelidados de sets.

A grande facilidade do usuário em interagir com o conteúdo torna o Flickr um sistema tipicamente 2.0. A interação é principalmente feita através de comentários nas fotos, mas também existem os grupos. Estes grupos reúnem conteúdo de pessoas que tenham algum interesse em comum e a interação é feita através de adição de fotos e discussão em fórum.

Recursos do Flickr

1) Envio de Fotos pelo site

O recurso básico é o envio de fotos pelo próprio site

2) Envio de Fotos por APIs (Picasa, Flickr Uploadr)

Suas APIs permitem que o envio de fotos seja feito por outros programas como o Picasa do Google, por exemplo.

3) Tagueamento de fotos

Tagueamento de fotos apenas pelo criador.

4) Grupos de interesse

Estes grupos podem conter várias fotos e discussões.

5) Geotagging

Georeferencia para falar em qual local do mundo a foto foi tirada.

6) Comentários em regiões específicas da foto

É possível fazer comentários específicos em determinadas áreas (retangulares) da foto.

7) Slideshow de fotos

Slideshow das fotos de um usuário em Flash.

Segunda Parte : Sistema de Notícias

Especificações

Descrição

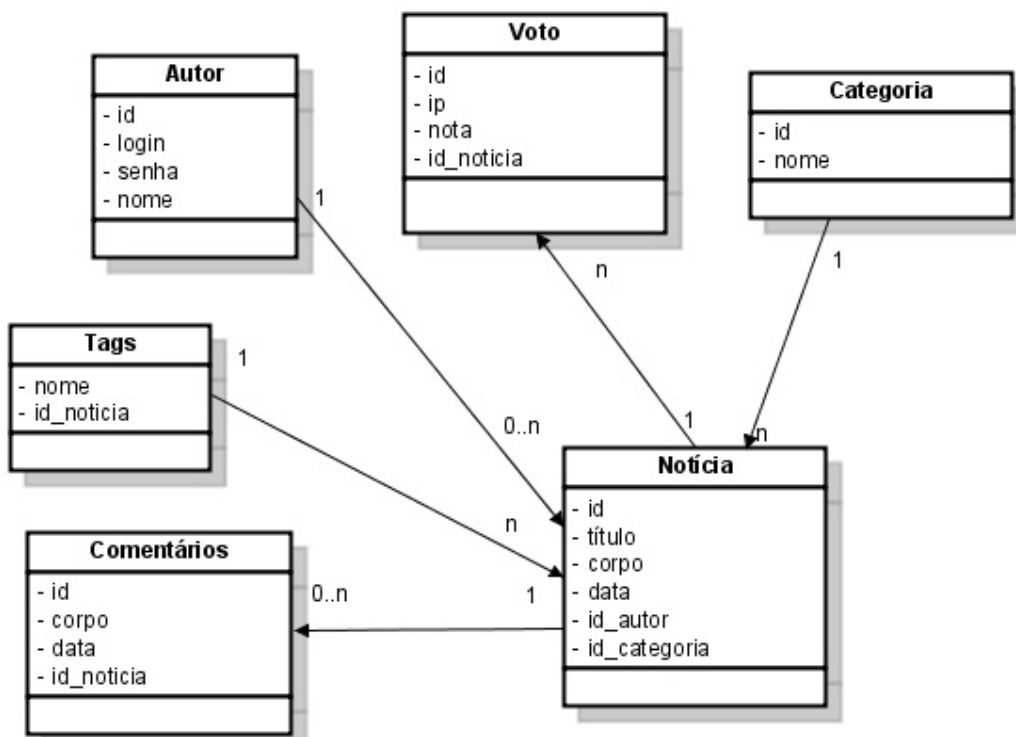
Sistema de notícias Web com múltiplos autores, sistema de avaliação pelos usuários, comentários, classificação por categoria e tags.

Objetivos

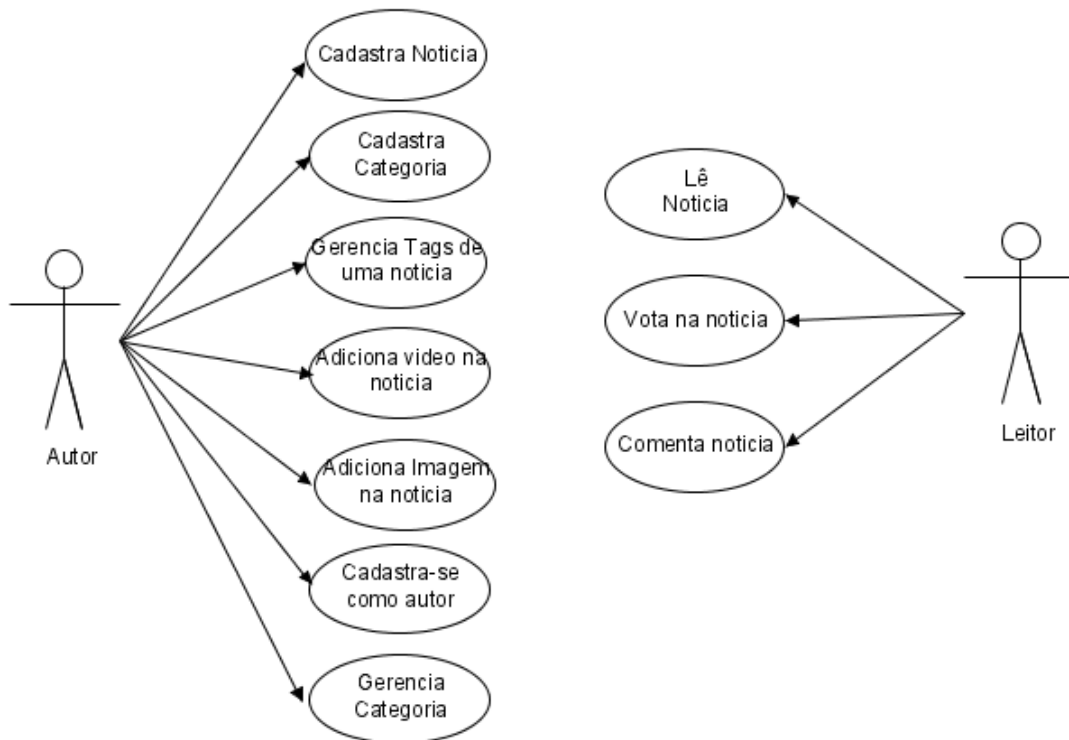
- Permitir inclusão de notícias.
- Permitir inclusão de comentários por visitantes.
- Permitir avaliação de notícias por nota.
- Permitir inclusão de vídeos e fotos da web (através da URL dos mesmos). - Permitir adição de tags livres pelo autor.

Arquitetura

Arquitetura do Banco de Dados :



Estudo de Casos



Justificativa da escolha do tecnologia

- Ajax:** Usamos o AJAX para criar um sistema mais iterativo com o usuário, possibilita atualizar dados sem recarregar a página toda, o AJAX roda no próprio navegador web. No nosso projeto utilizamos o AJAX na busca das últimas notícias enquanto o usuário vai digitando a palavra de busca o AJAX manda requisições para o servidor que busca pela notícia e usando o AJAX o resultado da busca é alterado.
- Comet:** Decidimos usar o comet para gerar atualização na página assim que uma notícia é adicionada, ou seja, o cliente não precisa fazer uma requisição para o servidor pedindo a atualização da página. Usando o comet o servidor identifica que houve uma nova notícia e empurra essa informação para a página sem uma requisição do cliente. Para o site implementamos uma seccao que mostra as últimas notícias do site usando o comet. Assim o usuário não precisa atualizar o navegador para ter as últimas notícias.
- JSON:** O JSON foi usado no nosso sistema para fazer o parse de uma requisição feita a um webservice. Criamos para o sistema notícia uma busca nas notícias da New York Times usando a api disponibilizada. O usuario faz uma busca, o sistema utilizando a api da New York Times faz uma requisição das notícias buscadas o retorno é um arquivo JSON. O arquivo é tratado e as informações é disponibilizada para o usuário.
- JQuery:** Facilitou muito no desenvolvimento e no entendimento de códigos JavaScript. Usamos JQuery em quase todas as páginas. Gostamos muito dessa biblioteca e sua utilização é de fácil aprendizado. O JQuery foi usando nas ferramentas implementadas em AJAX e Comet.

- **Hibernate e JPA:** Usamos o Hibernate mapeando algumas das classes para entender o seu funcionamento. Identificamos que a vantagem do mapeamento Objeto-Relacional é que os desenvolvedores não precisou escrever query em SQL. Bastava apenas mapear os objetos a ser persistido e salva-lo no banco. Gostamos muito desse framework sua utilização facilita a manipulação do banco de dados.
- **Spring:** Usamos o Spring apenas para entender o seu funcionamento e suas vantagens. Notamos que o framework é de difícil entendimento mas que a injeção de dependência e a programação orientada a aspectos ajuda muito em um sistema complexo.
- **JUnit e Selenium:** implementamos alguns testes unitários para as nossas classes usando o JUnit. E com o Selenium testamos a utilização do site chamado de teste de aceitação. Identificamos que testes automatizados tem muitas vantagens para manter a consistencia do sistema, segurança, além disso ajuda na documentação. Os testes implementados foram bons em situações de refatoração de certo trecho de código. Se temos testes cobrindo a funcionalidade do código a ser alterado pudemos alterar o código sem se preocupar se a funcionalidade será mantida pois os teste implementados garantiram a funcionalidade.
- **Estilo arquitetural REST e Mashups:** Utilizamos o padrão REST nas chamadas das nossas páginas um detalhe importante o estilo REST na url é que o um dos critérios para o ranqueamento da páginas são url legível na url. No sistema de notícia criamos um mashup que utiliza uma API do jornal New York Times e consulta suas notícias. O Mashup na concepção web2.0 é importante pois utiliza conteúdo disponibilizado por outros sites para agregar conhecimento ao seu sistema isso é importante para algoritmos de inteligência coletiva.
- **Ruby on Rails:** R&R foi o framework estudado pelo grupo. Percebemos que sua simplicidade e agilidade são suas principais características e isso nos incentivou a escolha da ferramenta.

Parte 3 - Conclusões sobre a disciplina

A disciplina foi bem prática foi possível aprender todos muitos frameworks e novos conceitos da Web2.0. Sentimos que o curso de Ciência da Computação na USP tem poucas disciplinas práticas relacionadas à Web e suas tecnologia essa disciplina nos deu conhecimento prático e conceitual de como criar um ótimo sistema Web.

Seria interessante se, em uma turma futura, os alunos pudessem escolher a tecnologia principal para ser trabalhada durante o curso. Particularmente, nós estamos mais familiarizados com Ruby on Rails e CakePHP e sentiríamos a vontade para desenvolver um projeto mais completo.

Por estarmos cursando no título de graduando (e não pós-graduando), achamos um pouco pesada a carga horária de trabalho. Este é um caso típico de disciplina que deveria ser 4+2 créditos, sendo que os 2 adicionais são de trabalho.

Enfim, estamos felizes por finalmente ter uma disciplina diretamente relacionada com o nosso dia-a-dia no trabalho. É uma pena que nossos colegas de turma de 2004 não terem aproveitado essa oportunidade.