Componentes para interoperabilidade entre redes sociais na Web 2.0

Carlos Leonardo Herrera Muñoz

Dissertação Apresentada
Ao
Instituto de Matemática e Estatística
Da
Universidade de São Paulo
Para
Obtenção do Título
De
Mestre em Ciências

Programa: Mestrado em Ciência da Computação

Orientador: Prof. Dr. Marco Aurélio Gerosa

Durante o desenvolvimento deste trabalho o autor recebeu auxílio financeiro do CNPq

São Paulo, Junho de 2013

Componentes para interoperabilidade entre redes sociais na Web 2.0

Carlos Leonardo Herrera Muñoz

Este exemplar corresponde à redação final da dissertação submetida à comissão julgadora apresentada por Carlos Leonardo Herrera Muñoz ao programa de Mestrado em Ciência da Computação da Universidade de São Paulo.

Comissão Julgadora

- Prof. Dr. Marco Aurélio Gerosa (orientador) – IME-USP
- Prof. Dr. Ewout ter Haar – IF-USP
- Prof. Dr. Flávio Soares Corrêa – IME-USP

Agradecimentos

Quero começar agradecendo ao Instituto de Matemática e Estatística da Universidade de São Paulo, que me possibilitou aprender tantas coisas e me ajudou a crescer profissionalmente, agradeço também ao Brasil que me deu a oportunidade de conhecer uma nova cultura e pessoas maravilhosas.

Quero agradecer de forma especial ao meu orientador, o Prof. Dr. Marco Aurélio Gerosa pela confiança, atenção e paciência que depositou em mim ao longo desse período do mestrado, por ser um excelente educador e exemplo de pessoa. Também agradeço aos professores Flávio Soares Corrêa e Ewout ter Haar membros da comissão julgadora da presente dissertação.

Agradeço ao CNPq pelo apoio financeiro por meio da bolsa de estudos que me ajudou muito a poder realizar o curso de mestrado.

Agradeço infinitamente aos meus pais, Rica e Leo, pela vida, os conselhos e o constante apoio para sempre sair na frente, e desejo muito que a minha mãe consiga-se recuperar totalmente e volte a ser a mesma de antes. Agradeço às minhas irmãs Evelyn e Piera, e aos meus primos Samuel e Christian por sempre estarem comigo nos momentos difíceis e pelas alegrias compartilhadas.

Aos meus grandes amigos Erick, Gerby, Dennis, Maritza e Ligia que são como uma segunda família para mim, agradeço por todos aqueles momentos de descontração que compartilhamos e ajudaram a aguentar o fato de estar longe de nossa querida Arequipa.

Agradeço ao Marcelo, Rogerio, José e Edith com quem compartilho o dia a dia na Kompass-RE e ajudaram me tornar um melhor profissional nestes anos, agradeço à equipe do Groupware Workbench com quem aprendi muito. Por último quero agradecer ao Bernardo, Rosário, Anyie, Carla, Fabian, Alfonso, Medardo, e Ysaac pela amizade verdadeira.

Resumo

Muñoz, C. L. H. **Componentes para interoperabilidade entre redes sociais na Web 2.0.** 2013. 102f Dissertação (Mestrado) - Instituto de Matemática e Estatística da Universidade de São Paulo, São Paulo, 2013.

Nos últimos anos, as redes sociais na Web 2.0 vêm ganhando cada vez mais importância para trabalhar e compartilhar ideias. As redes sociais armazenam informações do usuário, como preferências, experiência profissional, dados pessoais e com quem o usuário interage. Essas informações são úteis para diversos fins, como oferecer produtos e serviços personalizados. Com a aparição de cada vez mais redes sociais, surgem problemas como a duplicação de perfis de usuários. Atualmente há algumas técnicas para interoperar as redes sociais, como serviços de autenticação única ou representação padrão para compartilhamento de dados. O objetivo deste trabalho foi realizar um estudo dessas técnicas e tecnologias disponíveis, implementá-las por meio de componentes do Groupware Workbench, e implantar e avaliar os componentes desenvolvidos na rede social Arquigrafia. A avaliação dos componentes foi realizada por meio dos aspectos e questões propostos pelo projeto DataPortability. A avaliação mostrou que as questões diretamente relacionadas com a interoperabilidade técnica e semântica foram respondidas.

Palavras-chave: redes sociais, Web 2.0, interoperabilidade, desenvolvimento baseado em componentes, data portability.

Abstract

Muñoz, C. L. H. **Components for interoperability among Web 2.0 social networks.** 2013. 102f Dissertação (Mestrado) - Instituto de Matemática e Estatística da Universidade de São Paulo, São Paulo, 2013.

In recent years, the importance of social networks for working and sharing ideas increased. Social networks store information such as user preferences, professional experience, personal data, and information about with whom the user interacts. This kind of information is useful for many purposes, such as offering personalized products and services. With the appearance of more social networks, problems as duplication of user profiles arise. Currently, there are some techniques for interoperating social networks, such as single authentication services and standard representation for data sharing. The objective of this work was to study techniques and technologies available, implement them as Groupware Workbench components, and deploy and evaluate the components in the social network Arquigrafia. The evaluation of the components was based on the DataPortability project. Issues related to technical and semantic interoperability have been successfully addressed.

Keywords: social networks, Web 2.0, interoperability, component based development, data portability.

Sumário

Lista de Figuras xi

Lista de Tabelas xiii

Lista de Abreviaturas xiv

1 Introdução 1

1.1 Objetivos 3

1.1.1 Objetivo Principal 3

1.1.2 Objetivos secundários 3

1.2 Estrutura da dissertação 4

2 Interoperabilidade de redes sociais na Web 2.0 5

2.1 Princípios da Web 2.0 5

2.1.1 A Web como plataforma 5

2.1.2 Os dados de uma aplicação são o seu verdadeiro valor 6

2.1.3 Modelos leves de programação 6

2.1.4 Experiência rica do usuário 7

2.2 Redes Sociais na Internet 7

2.2.1 Problema das redes isoladas 8

2.3 Interoperabilidade 9

2.3.1 Dimensões da interoperabilidade 11

2.3.2 Importância dos padrões 12

3 Trabalhos Relacionados 14

4 Técnicas e Tecnologias para Interoperabilidade nas Redes Sociais 22

4.1 Sistema de autenticação padrão compartilhado entre serviços na internet 24

4.1.1 OpenID 24

4.1.2 OAuth 27

4.2 Formatos padrões para representar a informação dos usuários 34

4.2.1 RSS 34

4.2.2 RDF 34

4.2.3 vCard 35

4.2.4 Friend of a Friend 36

4.2.5 Activity Stream 37

4.3 Serviços Web RESTful e uso de conteúdo de fontes externas 39

4.3.1 Serviços Web RESTful 39

4.4 Definição de políticas de gerenciamento e segurança dos dados do usuário 41

5 Componentes do Groupware Workbench para prover interoperabilidade 43

5.1 Groupware Workbench 43

5.2 Descrição dos componentes de interoperabilidade 47

6 Avaliação no Arquigrafia 54

6.1 Rede Social Arquigrafia 54

6.2 Facebook 57

6.2.1 Graph API 57

6.2.2 Authentication 60

6.3 Orkut 60

6.4 Twitter 62

6.5 Implantação dos componentes no Arquigrafia 62

6.5.1 Single sing-on 63

6.5.2 Login Service 68

6.5.3 Add Friends 69

6.5.4 Activity Feed 70

6.5.5 ProfileExport 71

6.6 Análise com respeito ao Projeto DataPortability 73

6.7 Análise com respeito aos requisitos de interoperabilidade para redes sociais 76

7 Conclusão 78

7.1 Contribuições 81

7.2 Trabalhos relacionados 81

Referências Bibliográficas 83

Apêndice 91

Outros sistemas de componentes relacionados ao Groupware Workbench 91

1. IBM Connections 91

2. Plataforma Elgg 92

3. Portlets 93

4. WordPress 94

Lista de Figuras

Figura 2.1 Interseção entre serviços de rede sociais 8

Figura 2.2 Usuários comuns entre redes sociais na internet 9

Figura 3.1: Rede Social Tagged 20

Figura 3.2: Rede social AreYouInterested? 20

Figura 3.3: A social aggregator HootSuite 21

Figura 4.1: Funcionamento do OpenId 27

Figura 4.2: Fluxo OAuth1.0a 30

Figura 4.3: fluxo servidor web OAuth2.0 32

Figura 4.4: Fluxo usuário-agente OAuth2.0 33

Figura 4.5: Formato Activity Streams 39

Figura 4.6 Exemplo de serviços RESTful 40

Figura 4.7: API RESTful do Facebook 41

Figura 5.1: Arquitetura de Serviço do Groupware Workbench (MICHALSKY, 2012) 45

Figura 5.2: Componente CommentMgr 46

Figura 5.3: Fragmento JSTL do widget CommentMgr 46

Figura 5.4: Componente de comentário no Arquigrafia 47

Figura 5.5: Diagrama de classes do componente Single sign-on 48

Figura 5.6: Diagrama de classes do componente Login Service 49

Figura 5.7 Diagrama de classes do componente AddFriends 50

Figura 5.8: Diagrama de classes do componente Activity Feed 52

Figura 5.9: Diagrama de classes do componente ProfileExport 53

Figura 6.1: Protótipo do Arquigrafia 55

Figura 6.2 Informações da pagina do Facebook Platform 58

Figura 6.3 Diagrama de componentes Arquigrafia 63

Figura 6.4: Pagina de Login do Arquigrafia 64

Figura 6.5: Fluxo de autenticação para Facebook.....65

Figura 6.6: Fluxo de autenticação para Orkut 66

Figura 6.7: Fluxo de autenticação para Twitter.....67

Figura 6.8: Fluxo do componente Login Service.....68

Figura 6.9: Componente “Add Friends” utilizando Facebook.....69

Figura 6.10: Componente “Add Friends” utilizando Twitter 69

Figura 6.11: Utilização do componente “Activiy Feed”.....71

Figura 6.12: Funcionamento do componente “Profile Export”.....72

Figura 6.13: Documento FOAF do usuário *Carlos Herrera Muñoz* do Arquigrafia 73

Lista de Tabelas

Tabela 4.1: Questões cobertas pelas técnicas identificadas.....23

Tabela 4.2: Requisitos cobertos pelas técnicas identificadas 23

Tabela A1: Comparação do Groupware Workbench com outras tecnologias de componentes.....95

Lista de Abreviaturas

API	Application Programming Interface
AJAX	Asynchronous JavaScript and XML
BPEL4WS	Business Process Execution Language for Web Services
FOAF	Friend of a Friend
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
ISO	International Organization for Standardization
JSON	JavaScript Object Notation
JSTL	JavaServer Pages Standard Tag Library
MVC	Model View Controller
OWL	Web Ontology Language
PC	Personal Computer
PDA	Personal Digital Assistant
RDF	Resource Description Framework
REST	Representational State Transfer
RSS	Really Simple Syndication
SIOC	Semantically-Interlinked Online Communities
SOA	Service-Oriented Architecture
SOAP	Simple Object Access Protocol
SSO	Single Sign-on
TIC	Tecnologia da Informação e Comunicação
UDDI	Universal Description, Discovery and Integration
URL	Uniform Resource Locator
W3C	World Wide Web Consortium
XML	Extensible Markup Language
XMPP	Extensible Messaging and Presence Protocol

1 Introdução

Empresas baseadas na Web sobreviventes à crise de 2001 tinham algo em comum, usavam a Web como uma plataforma, desenvolvendo sítios colaborativos baseados em comunidades (O'REILLY, 2005). Essa nova concepção de desenvolvimento foi chamada de Web 2.0. Diferentemente do modelo que passou a ser conhecido como Web 1.0, em que um número relativamente pequeno de empresas e anunciantes produzem conteúdo para os usuários acessarem, a Web 2.0 envolve o usuário. Não apenas o conteúdo é frequentemente expandido por ele, mas também organizado, compartilhado, mesclado, criticado e atualizado.

A Web 2.0 aumentou a possibilidade de socialização por meio das ferramentas computacionais. Uma das formas mais expressivas são as redes sociais, que são caracterizadas por atores (pessoas, instituições ou grupos), que são os nós das redes e suas conexões (interações ou laços sociais) (RECUERO, 2009). Os serviços de redes sociais na Web guardam informações do usuário como preferências, experiência profissional, vida acadêmica, geo-localização e com quem o usuário interage. Essas informações são úteis para diversos fins, como oferecer produtos e serviços personalizados.

Com o passar do tempo apareceram vários serviços baseados em redes sociais, por exemplo, serviços focados em gosto musical (Last.fm[1]), compartilhamento de fotografias (Flicker[2]), ou simplesmente para criação de relações sociais (Facebook[3]). Segundo estudos da Compete (2007) e Anderson Analytics (2009), usuários possuem contas em mais de um desses serviços de redes sociais. Essas interseções originam o problema de replicação de

[1] http://www.lastfm.com
[2] http://www.flickr.com
[3] http://www.facebook.com

identidades, que acontece quando um usuário cria um perfil para cada serviço de rede social no qual participa, replicando assim suas informações pela internet.

Para amenizar esse problema, é preciso que os serviços de redes sociais na internet troquem informações. Segundo a ISO (2011) o conceito de interoperabilidade é "a habilidade de dois ou mais sistemas (computadores, meios de comunicação, redes, software e outros componentes de tecnologia da informação) de interagir e de intercambiar dados de acordo com um método definido, de modo que sejam obtidos resultados esperados".

Outro benefício da interoperabilidade entre serviços é o compartilhamento de usuários. Segundo Deitel (2008) o valor agregado de um serviço de rede social cresce à medida que o número de usuários também cresce, e a lei de Metcalf (METCALF, 1993) afirma que o valor da rede é proporcional ao quadrado do número de usuários. Portanto, serviços de redes sociais beneficiam-se ao compartilhar informações como o perfil dos seus usuários.

Um problema que traz consigo a interoperabilidade na internet, é o grande numero de tecnologias disponíveis e a complexidade requerida para sua implementação. Este trabalho objetivou identificar técnicas e tecnologias disponíveis, implementá-las por meio de componentes do Groupware Workbench (2010), implantar os componentes na rede social Arquigrafia e avaliar a rede social, junto com os componentes, por meio da lista de questões proposta pelo projeto Data Portability. Data Portability é um projeto patrocinado por empresas como Google, Facebook, Microsoft, Plaxo, Drupal, Netvibes, Mystrands, LinkedIn, Flickr, Six Apart, Twitter e Digg, que visa a dar a possibilidade aos usuários de reusar seus dados em qualquer aplicação ou serviço na internet. Nesse projeto é apresentada uma lista de questões que uma aplicação interoperável na internet deveria abordar.

O projeto Arquigrafia (ROZESTRATEN, MARTINEZ, *et al.*, 2010) é uma rede social online cujo conteúdo principal é baseado em imagens arquitetônicas. A rede é

orientada para o uso de alunos, professores e pessoas interessadas em estudar e compartilhar imagens de arquitetura. O projeto está sendo desenvolvido no IME/USP em parceria com a FAU/USP e ECA/USP. A rede social Arquigrafia é desenvolvida com base na plataforma Groupware Workbench (2010). O Groupware Workbench é um projeto de software livre, desenvolvido em parceria entre o IME/USP, a UFES e a PUC-Rio e está hospedado no Centro de Competência em Software Livre (CCSL) do IME/USP.

Dado que o Arquigrafia é uma rede social nova que busca atrair usuários rapidamente, a utilização de componentes que favoreçam a participação desses usuários acrescentaria valor à rede social. A maior parte do público alvo do Arquigrafia faz parte de alguma rede social na internet. Interoperar o Arquigrafia implica ganhar visibilidade na internet e aproveitar dados já existentes de usuários.

Por conta das necessidades do projeto Arquigrafia esse trabalho aborda mais os aspectos técnicos da interoperabilidade, deixando abertos outros aspectos como o semântico e o operacional para futuros trabalhos.

1.1 Objetivos

1.1.1 Objetivo Principal

Implementar em componentes de software técnicas para interoperar redes sociais na Web 2.0 e avaliar os componentes e técnicas na rede social Arquigrafia.

1.1.2 Objetivos secundários

- Identificar técnicas para interoperabilidade de redes sociais na Web 2.0.
- Fazer um estudo das tecnologias existentes que deem suporte às técnicas.
- Implementar as técnicas em componentes da plataforma Groupware Workbench.
- Implantar e avaliar as técnicas de interoperabilidade e os componentes desenvolvidos na rede social Arquigrafia.

1.2 Estrutura da dissertação

No Capítulo 2 é descrito o que é a web 2.0, o que é um serviço de rede social na internet e o problema de ter serviços de rede social sem comunicação. Também é apresentado o conceito de interoperabilidade, com suas dimensões e vantagens. No Capítulo 3, é apresentado um estudo sobre os trabalhos relacionados. No Capítulo 4 são apresentadas técnicas para interoperar redes sociais e também as tecnologias para implementá-las. No Capítulo 5 é descrito o modelo de componentes do Groupware Workbench (2010) e são definidos os componentes de interoperabilidade. No Capítulo 6 é apresentada uma prova de conceito dos componentes de interoperabilidade, implantados na rede social Arquigrafia[4], integrando-a às redes sociais Facebook[5], Twitter[6] e Orkut[7]. No Capítulo 7 é feita uma análise da prova de conceito, utilizando as diretrizes do projeto Data Portability e quatro requisitos de interoperabilidade para redes sociais. Por fim, no Capítulo 8 são apresentadas as conclusões do presente trabalho.

[4] www.arquigrafia.org.br
[5] www.facebook.com
[6] www.twitter.com
[7] www.orkut.com

2 Interoperabilidade de redes sociais na Web 2.0

Muito se tem dito sobre uma nova geração de serviços e aplicativos da Web e sobre tecnologias e conceitos que possibilitam um maior grau de interatividade e colaboração na internet. Para designar esse fenômeno, Tim O'Reilly cunhou o termo "Web 2.0" que se popularizou rapidamente a partir da publicação do artigo intitulado "*What is Web 2.0: Design Patterns and Business Models for the Next Generation of Software*" (O'REILLY, 2005). Embora possa parecer que esse termo esteja associado a uma grande mudança tecnológica, a Web 2.0 refere-se a uma nova forma com que a Web vem sendo encarada por seus usuários e desenvolvedores. A maior parte das bases tecnológicas envolvidas existe há alguns anos, sendo utilizadas de uma forma diferente. A novidade é um conjunto de princípios e práticas comuns. É a esse triângulo composto de princípios, tecnologias e aplicações, que comumente se chama de Web 2.0 (BOMFIM e SAMPAIO, 2008). A Web 2.0 é caracterizada pela descentralização, pelo compartilhamento e pelo reúso, sendo valorizado o conteúdo gerado de forma coletiva.

2.1 Princípios da Web 2.0

Segundo O'Reilly (2005), a Web 2.0 é baseada em seis princípios, dos quais os mais relevantes para o presente trabalho são: a Web como plataforma, os dados de uma aplicação são o seu verdadeiro valor, modelos leves de programação e a experiência rica do usuário. Os outros dois princípios são: aproveitar a inteligência coletiva e os usuários agregarem valor.

2.1.1 A Web como plataforma

A Web é uma fonte de conteúdo e funcionalidades. Um software é apresentado como um serviço e não como um produto, acarretando mudanças fundamentais no modelo

de negócio das empresas. Como não existe uma versão definitiva a ser distribuída, as ferramentas ficam em produção contínua, sendo continuamente testadas por um grande número de usuários. Diferentemente do que acontece com sistemas tradicionais, que possuem instaladores e são dependentes de um sistema operacional, aplicativos Web são atualizados de forma transparente aos usuários. Além disso, as aplicações Web não se limitam mais à plataforma PC. Torna-se possível utilizar diversos equipamentos conectados, como computadores de bolso, celulares e *tablets*.

2.1.2 Os dados de uma aplicação são o seu verdadeiro valor

Os aplicativos da Web são, cada vez mais, apoiados por grandes bancos de dados especializados. Há bancos de dados que indexam as referências para as páginas da internet, bancos de imagens capturadas por satélites, bancos de produtos e consumidores das grandes lojas virtuais, entre outros. O maior valor das empresas está nos dados que elas mantêm e não no software propriamente dito.

2.1.3 Modelos leves de programação

A Web 2.0 prega a construção de sistemas fracamente acoplados[8] em que as funcionalidades são construídas e agregadas. Entre as principais estratégias utilizadas para atingir esse modelo está a definição de APIs abertas baseadas em tecnologias simples, como RESTful (RODRIGUEZ, 2008) e JSON (1999), e a agregação de conteúdo.

A agregação de conteúdo possibilita a utilização da informação em sites de terceiros ou em outros ambientes. Baseadas em formatos padronizados, como RSS ou ATOM[9], o conteúdo é publicado sem o conhecimento sobre o que será feito com ele em seu destino. Cabe à aplicação requisitante decodificar o arquivo recebido, apresentando os dados ao usuário.

---

[8]São aqueles sistemas cujos módulos possuem um baixo nível de interdependência.

[9] Atom é um protocolo para ler e escrever informações na Web (http://bitworking.org/projects/atom/rfc5023.html)

O uso de APIs abertas possibilita que os aplicativos Web 2.0 incorporem novos serviços e forneçam funcionalidades para serem agregadas por terceiros. As APIs da Web 2.0 geralmente fornecem dados em formato XML ou JSON via HTTP por intermédio de requisições aderentes ao estilo arquitetural REST. Esse fato possibilita que os aplicativos sejam construídos a partir de uma rede cooperativa de serviços.

2.1.4 Experiência rica do usuário

As tecnologias utilizadas na Web 2.0 favorecem a criação de aplicações Web com interfaces mais ricas do que as das aplicações Web tradicionais. O AJAX[10] é uma combinação de algumas tecnologias, capaz de buscar dados assincronamente no servidor, exibindo as páginas Web na tela do usuário sem recarregá-las completamente, representando uma mudança no paradigma de funcionamento das aplicações Web por meio de requisições e respostas síncronas.

2.2 Redes Sociais na Internet

Redes sociais se popularizaram na Web 2.0. Segundo Duarte (DUARTE, QUANDT e SOUZA, 2008), "uma rede social é uma estrutura social composta por pessoas ou organizações, conectadas por um ou vários tipos de relações, que compartilham valores e objetivos comuns". Uma das características fundamentais das redes sociais é que possibilitam relacionamentos horizontais não hierárquicos entre os participantes.

De acordo com Recuero (2009), uma rede social é definida como:

> Um conjunto de dois elementos característicos que servem de base para que a rede seja percebida e suas informações compreendidas: atores, que podem ser pessoas, instituições ou grupos; e suas conexões (interações ou laços sociais).

[10] Asynchronous JavaScript and XML

No seu livro "Redes Sociais na Internet", Recuero (2009) indica caminhos possíveis para mapear, compreender e discutir as redes sociais.

De acordo com Cardozo (2008), a questão central das redes sociais é a valorização dos elos informais e das relações, em detrimento das estruturas hierárquicas. As redes sociais representam um conjunto de participantes autônomos, unindo ideias e recursos em torno de valores e interesses compartilhados. Os sistemas de redes sociais na internet constituem as relações entre os indivíduos na comunicação mediada por computador. Esses sistemas funcionam por meio da interação social, tendo como principal objetivo conectar pessoas e favorecer sua comunicação.

2.2.1 Problema das redes isoladas

Num serviço de rede social são armazenadas informações recorrentes. Essas informações são o perfil, composto das informações básicas da pessoa como nome, correio eletrônico, data de aniversário e o grafo social composto pelas relações de amizade com outras pessoas.

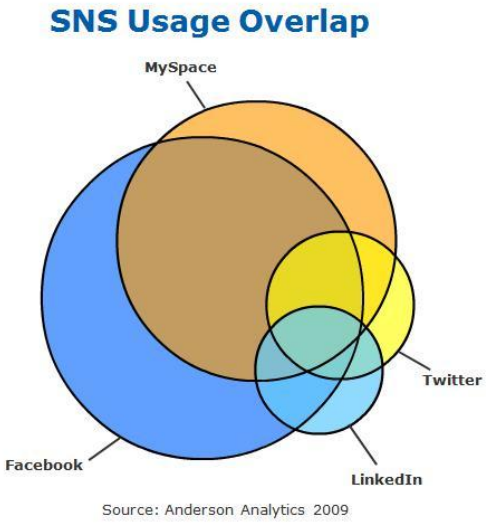

Figura 2.1 Interseção entre serviços de rede sociais[11]

[11] Imagem tomada de http://www.tomhcanderson.com/2009/07/09/overlap-among-major-social-network-services

Segundo estudos da Compete (2007) e Anderson Analytics (2009), há pessoas que possuem contas em mais de um serviço baseado em rede social, causando interseções, como pode ser observado na Figura 2.1. Essas interseções originam o problema da replicação de identidades, que acontece quando um usuário cria perfil e grafo social para cada serviço de rede social no qual quer participar. Uma comparação entre redes sociais e seus usuários em comum é apresentada na Figura 2.2

users of these sites...

... are also users of these sites	LinkedIn	Facebook	Twitter	MySpace
LinkedIn	-	12%	21%	6%
Facebook	82%	-	91%	57%
Twitter	31%	20%	-	17%
MySpace	36%	49%	70%	-

Figura 2.2 Usuários comuns entre redes sociais na internet[12]

Os serviços de redes sociais poderiam interoperar de tal modo que os usuários de um serviço pudessem ver as informações de outro usuário em outro serviço, respeitando as políticas de privacidade.

2.3 Interoperabilidade

Uma das definições mais citadas (EICTA, 2006) sobre interoperabilidade é a proposta pela IEEE (1990): "interoperabilidade é a habilidade de dois ou mais sistemas, redes de comunicação, aplicações ou componentes trocarem informações entre eles". Por meio da Interoperability Technical Framework (ITF) (AGIMO, 2003), o governo australiano define interoperabilidade como "a capacidade de transferir e utilizar informações de uma maneira uniforme e eficiente por múltiplas organizações e sistemas de tecnologias de informação".

[12] Imagem tomada de http://www.tomhcanderson.com/2009/07/09/overlap-among-major-social-network-services

Nessa mesma linha, o European Interoperability Framework for Pan-European eGoverment Services (IDABC, 2004), que é o documento de referência para o desenvolvimento de padrões e infraestruturas comuns necessárias para a implementação da interoperabilidade na área da administração eletrônica europeia, considera que "a interoperabilidade é a capacidade dos sistemas de tecnologias da informação e comunicação (TICs) e os processos de negócio, para trocar dados e ter a capacidade de compartilhar informações e conhecimento".

O comitê CEN-TC251 do Comitê Europeu da Normalização (THEODOROU, 2007) define a interoperabilidade não como capacidade ou habilidade como os anteriores, mas como um estado "que existe entre duas entidades de aplicações, em relação a uma tarefa específica, quando uma aplicação aceita dados de outra e realiza uma tarefa de forma apropriada e bem sucedida sem necessidade de intervenção externa" (tradução livre).

Concluindo, a interoperabilidade faz referencia à relação entre dois ou mais recursos ou sistemas de tecnologias da informação e comunicação, que precisam trabalhar conjuntamente de forma automática. As definições coincidem ao indicar duas capacidades com as quais devem contar os sistemas interoperáveis: comunicar-se (para poder transferir informações) e entender a estrutura das informações que são transferidas entre as entidades (para poder utilizá-las).

A interoperabilidade traz diversos benefícios, como, por exemplo, maior efetividade (interconexão em vez de soluções isoladas), eficiência (redução dos custos de transação e aumento da participação dos agentes envolvidos) e receptividade (melhor acesso a mais informações, possibilitando a resolução mais rápida dos problemas) (LANDSBERGEN e WOLKEN, 2001).

Por outro lado, existem barreiras significativas para alcançar a interoperabilidade de forma efetiva e ampla. Essas barreiras são classificadas como políticas, organizacionais, econômicas e técnicas (ANDERSEEN e DAWES, 1991):

- políticas – definição das diretrizes das políticas adotadas; conflitos nas definições dos níveis de privacidade nos acessos às informações; cultura organizacional predominante; ambiguidade da autoridade na coleta e uso das informações; descontinuidade administrativa;

- organizacionais – falta de experiência e ausência da predisposição de compartilhar; nível de qualificação do pessoal envolvido nos processos; cultura organizacional;

- econômicas – falta de recursos para disponibilização das informações para outros órgãos; forma de aquisição dos recursos (normalmente adquiridos pelo menor preço e não pelo melhor valor agregado);

- técnicas – incompatibilidade de hardware e software adotados; direitos de propriedade; desconhecimento dos dados gerados e armazenados pelos sistemas; múltiplas definições de dados.

2.3.1 Dimensões da interoperabilidade

Arms (2000) foi um dos primeiros a classificar formalmente os níveis da interoperabilidade. No memorando *Thoughts about Interoperabilidy in the NSDL*, o autor discorre sobre três dimensões da interoperabilidade: técnica, de conteúdo e organizacional. Segundo ele, somente a compreensão dessas três dimensões leva a uma melhoria da interoperabilidade nas organizações. O guia de interoperabilidade do governo brasileiro na versão 1.0A (2011) reconhece três dimensões da interoperabilidade: a dimensão técnica, semântica e organizacional.

A Interoperabilidade Técnica cobre as questões associadas à construção de sistemas computacionais interoperáveis. Inclui preocupações com arquiteturas de software, uso de padrões e tecnologias para o projeto e desenvolvimento de sistemas, além do uso correto dos meios de comunicação de dados (redes de computadores). A Interoperabilidade Semântica trata do significado da informação que se deseja transmitir ou utilizar. Assim, questões relacionadas à integridade, representação, formatação e interpretação dos dados são abordados pela interoperabilidade semântica. Por fim, a Interoperabilidade Organizacional ocupa-se dos processos de trabalho das organizações, o que envolve uso dos recursos materiais e humanos, além da definição de políticas adequadas para a promoção da cooperação e da coordenação dos serviços de governo.

2.3.2 Importância dos padrões

Segundo Tassey (2000), de uma forma ampla, um padrão é definido como um conjunto de especificações para o qual todos os elementos de produto, processos, formatos ou procedimentos sob sua jurisdição têm que estar de acordo. Para David e Greenstein (1990), um padrão é compreendido como um conjunto de especificações técnicas aderido por um grupo de fornecedores, tacitamente ou como resultado de um acordo formal.

David e Steinmueller (1994) classificam os padrões em quatro categorias: referência, qualidade mínima, interface e compatibilidade. Os padrões de compatibilidade possuem um papel relevante dentro das TIC, pois são os facilitadores do intercâmbio de dados entre componentes de um sistema em particular ou entre diferentes sistemas interorganizacionais.

Os padrões também são classificados de acordo com os processos por meio dos quais eles surgem. Uma distinção é feita frequentemente entre formal, de facto e de jure. Padrões formais são criados por entidades de padronização; os de facto são tecnologias

unificadas por seu uso no mercado; e os de jure são os impostos por lei (HANSETH e MONTEIRO, 1998).

De acordo com Graham et al. (GRAHAM, I. ET AL, 1995), o processo de padronização representa uma tentativa de alinhar interesses, práticas de negócios e expectativas de um grupo de pessoas com interesse em desenvolver e usar o sistema que será padronizado. Então, padronizar não é apenas prover uma solução utilizável, mas, principalmente, articular e alinhar expectativas e interesses (WILLIAMS, 1997).

Em relação às Tecnologias da Informação (TI), a padronização é definida como o processo pelo qual dois ou mais agentes concordam e aderem a um conjunto de especificações técnicas de um sistema, suas partes ou sua funcionalidade, tacitamente ou como resultado de um acordo formal (DAVID e GREENSTEIN, 1990). Consequentemente, esses padrões habilitam e restringem concorrentemente o comportamento de vários agentes no futuro (GARUD, JAIN e KUMARASWAMY, 2000). Esses efeitos sobre as ações futuras dos agentes envolvidos devem ser levados em conta pelos seus especificadores.

3 Trabalhos Relacionados

Para ter um melhor entendimento e panorama da interoperabilidade entre redes sociais na Web, o primeiro passo é saber quais informações são intercambiáveis pelos sistemas de redes sociais e como essas informações devem ser trocadas. Data Portability (2010) é um projeto formado por grandes empresas de internet (p. ex.: Facebook, Google, Plaxo, LinkedIn, entre outros). O projeto tem como objetivo reduzir o problema dos *walled gardens*[13] identificando e promovendo interoperabilidade e reúso de dados eletrônicos entre diferentes fornecedores de tecnologias.

Data portability (2010) visa a possibilitar aos usuários utilizar vários serviços na internet, reusando seus dados, tendo domínio da sua privacidade e respeitando a privacidade de terceiros. Também possibilita aos usuários ter controle sobre suas identidades, amigos, conversações e arquivos, sem ter que manualmente adicioná-los em cada novo serviço que utilize. Cada um dos serviços podem se beneficiar dessa informação, usando-a de acordo com seu contexto. À medida que o usuário vai alterando seus dados num serviço, esses dados alterados ficam disponíveis para os outros serviços assinados pelo usuário.

Data Portability (2010) propõe uma lista de dez aspectos descritos por meio de questões que orientam as aplicações na internet a intercambiar dados livremente, são elas:

1. **APIs e formato de dados**: As APIs e os formatos de dados estão documentados?
2. **Identidade e autenticação**: As pessoas precisam criar uma nova identidade para utilizar o produto? Ou elas podem reusar uma identidade existente?

[13] "Walled gardens" refere-se a um conjunto fechado e exclusivo de serviços fornecidos a um usuário, impossibilitando utilizar serviços e conteúdos de outras fontes.

3. **Trabalhar com conteúdo armazenado em outro lugar**: As pessoas devem importar conteúdos no seu produto, ou seu produto pode referenciar conteúdos armazenados em algum outro lugar? Seu produto pode trabalhar com objetos e informações em que a autorização seja gerenciada por outro produto, ou seu produto trabalha só com conteúdos armazenados por ele diretamente?
4. **Ver atualizações**: Seu site pode monitorar atualizações que as pessoas fizeram em outros sites? No caso em que seu produto mantenha rastro ou gerencie conteúdos que as pessoas tenham armazenado em outros produtos, pode seu produto automaticamente manter-se atualizado?
5. **Difundir alterações feitas aqui**: Se uma pessoa atualiza algum conteúdo, essa mudança é armazenada só aqui, ou pode ser notificada para outro produto? Seu produto fornece uma forma para que outros acessem a atualizações dos seus usuários?
6. **Acessar desde outros produtos**: As pessoas podem permitir que outros sites usem os conteúdos que criaram ou atualizaram no seu site? Seu produto fornece uma forma para que serviços de terceiros autentiquem uma pessoa, acesse e modifique o conteúdo?
7. **Cópia de segurança**: Pode uma pessoa recuperar uma cópia de tudo o que ela forneceu para seu site? Como parte do uso padrão de muitos produtos, as pessoas importam e criam conteúdos. Seu produto fornece uma forma aberta e livre de administração de direitos digitais, para outras pessoas recuperarem ou acessarem todos os conteúdos que elas têm criado ou fornecido?
8. **Dados públicos**: Uma pessoa pode baixar ou remotamente acessar a informação que outros têm fornecido para seu produto? No caso em que o produto permita baixar ou acessar remotamente, pode uma pessoa exportar todos os dados aos quais ela tem acesso ou só os dados que foram criados diretamente por ela?

9. **Fechar uma conta**: Seu site excluirá uma conta e todos os dados associados a pedido de um usuário? Se o usuário cria uma senha ou conta para uso do seu produto, o produto fornece uma forma de cancelar a conta e apagar todos os dados associados a ela?
10. **Onde são armazenados os conteúdos**: Você divulga onde os dados do usuário estão sendo armazenados no mundo real? Se sim, como o usuário pode saber onde os seus dados são armazenados? Ele pode solicitar que seus dados sejam armazenados em uma ou outra jurisdição?

Tiwari (2008) apresenta também uma série de requisitos de interoperabilidade para redes sociais, os quais um serviço de rede social deve reunir para interoperar com outro serviço:

1. Possibilidade de adicionar amigos entre redes sociais.
2. Possibilidade de acessar o relatório de atividades dos amigos de outras redes sociais.
3. Possibilidade de acessar a lista de amigos dos amigos de outras redes sociais (navegar o grafo social dos amigos).
4. Possibilidade de enviar mensagens aos amigos sem se importar a que rede social pertença.

Vários trabalhos tentam abordar algumas das questões do Data Portability de diferentes perspectivas. Um primeiro ponto de vista, descrito no trabalho de Curry (CURRY, KIDDLE e SIMMONDS, 2009), é feito sobre a rede social GeoChronos[14] (uma rede social vertical[15]), para realização de experiências baseadas na observação da Terra. A proposta é incluir toda a rede social vertical dentro de outra horizontal, e, depois, criar as

[14] http://geochronos.org/

[15] São aquelas que estão centradas num nicho de mercado, nestas redes os utilizadores ligam-se com a finalidade de debater um determinado assunto ou tema de interesse comum.

aplicações específicas que ofereçam suporte às experiências. Essa abordagem consegue cumprir os requisitos de interoperabilidade, mas o trabalho também indica que tiveram problemas de direito autoral, com as informações geradas pelas experiências.

Os trabalhos de Tapiador Del Dujo et al. (2008) e Mostarda et al. (2009) tomam como base a existência de duas redes sociais diferentes, as quais utilizam um autenticador centralizado baseado em OpenID. Desse modo, um usuário cria um perfil no servidor de OpenID (*service provider*) que é usado em cada uma das redes sociais que o usuário deseje participar. O trabalho de Mostarda et al. (2009) estende mais essa abordagem e agrega uma aplicação chamada de "adaptador" para cada uma das redes sociais, conseguindo assim publicar uma mesma informação em várias redes. Atualmente esse tipo de abordagem é a mais comum na internet, alguns exemplos são as redes sociais Tagged[16], que é de relacionamentos, e Photobucket[17], que é para compartilhar imagens.

Também há outros trabalhos que, por meio de ontologias, representam o perfil e o grafo social dos usuários. Desse modo, as informações de uma rede social são compartilhadas com outras compatíveis com aquela ontologia. Um exemplo disso é apresentado no trabalho de Golbeck e Rothstein (2008) em que os perfis e grafos sociais de usuários em várias redes sociais são representados com base na ontologia *Friend of a Friend* (FOAF PROJECT, 2000). O resultado da análise mostrou que os diversos perfis e grafos sociais de uma pessoa na internet podem ser misturados para formar um só perfil e grafo social global.

O trabalho de Schwagereit et al. (2010) buscou estender a ontologia FOAF para representar grupos dentro do conjunto de amigos. Essa extensão é chamada de dgFOAF e o objetivo é atacar o problema de um usuário participar de várias redes sociais em que os

[16] http://www.tagged.com/
[17] http://photobucket.com/

grafos sociais são diferentes. Cada um desses grafos sociais está representado por um grupo dentro do dgFOAF.

O trabalho de Bojars et al. (2008) não só utiliza FOAF para representar o perfil e o grafo social, mas também outra ontologia, chamada SIOC para representar o relatório de atividades de um usuário num serviço de rede social.

Outro trabalho importante na área de interoperabilidade é o projeto brasileiro e-Ping (2010), que apresenta orientações para o desenvolvimento de soluções de TIC (Tecnologia da Informação e Comunicação) como forma de incentivar a interoperabilidade governamental entre os entes da Federação. O e-Ping oferece um guia de interoperabilidade em dois volumes: um para professionais técnicos e outro para gerentes de TI. O e-Ping aborda a interoperabilidade de cinco formas: interconexão, segurança, meios de acesso, organização e intercambio de informações e áreas de integração para governo eletrônico. Um ponto importante para o presente trabalho é o estudo feito sobre as diferentes tecnologias para interoperar sistemas heterogêneos.

Na internet há redes sociais que conseguem interoperar com outras ou com serviços na Web. A interoperação é atingida por meio de APIs baseadas em serviços Web RESTFul, (p. e. a Graph API do Facebook). Algumas dessas redes sociais são:

- **Elgg[18]:** é uma plataforma de serviços de rede social de código aberto que oferece blogging, comunidades e troca de arquivos. Elgg fornece suporte para descrição de pessoas por meio de FOAF, sendo que na página do perfil do usuário existe uma ligação a um arquivo RDF com essa descrição. Também oferece suporte para interoperabilidade com Facebook e outras redes sociais por meio de plug-ins. Oferece, ainda, uma API baseada em REST. Um exemplo de rede social construída

[18] http://www.elgg.org/

com essa plataforma é a rede social Planet Red[19], a rede social da Universidade de Nebraska-Lincoln.

- **Tagged**[20]: É uma rede social de relacionamentos, que sugere o adicionamento de novas pessoas para o usuário, baseado nas suas preferências. Essa rede social interage com Facebook, Twitter e MySpace (Figura 3.1) de onde extrai dados do perfil por meio de OpenID e OAuth, e também publica informações das ações realizadas pelos usuários nas páginas das outras redes sociais.
- **Delicious**[21]: É uma rede social que objetiva compartilhar endereços de páginas na internet, agrupados por meio de tags. Delicious possibilita usar uma conta do serviço de OpenID do Yahoo, mas não é possível importar ou enviar convites a amigos de outras redes sociais.
- **Google+**[22]: É uma rede social desenvolvida pelo Google, com capacidade de importar os contatos e reusar uma conta do Google. Também está integrada no serviço de correio eletrônico do Google.
- **AreYouInterested?**[23]: É uma rede de relacionamentos construída sobre a plataforma de desenvolvimento do Facebook. AreYouInterested? (Figura 3.2) reutiliza a conta, o grafo social e as atividades do perfil do Facebook do usuário. Essa rede social funciona dentro do próprio Facebook ou como uma aplicação Web independente.

[19] http://planetred.unl.edu/
[20] http://www.tagged.com/
[21] http://www.delicious.com/
[22] http://www.google.com/buzz
[23] http://www.areyouinterested.com/

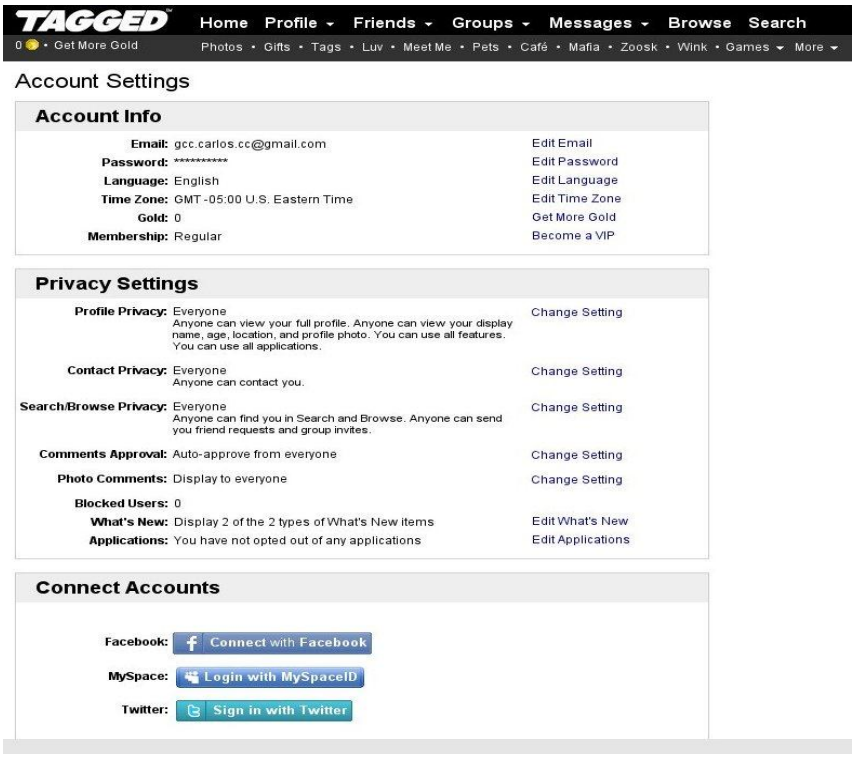

Figura 3.1: Rede Social Tagged

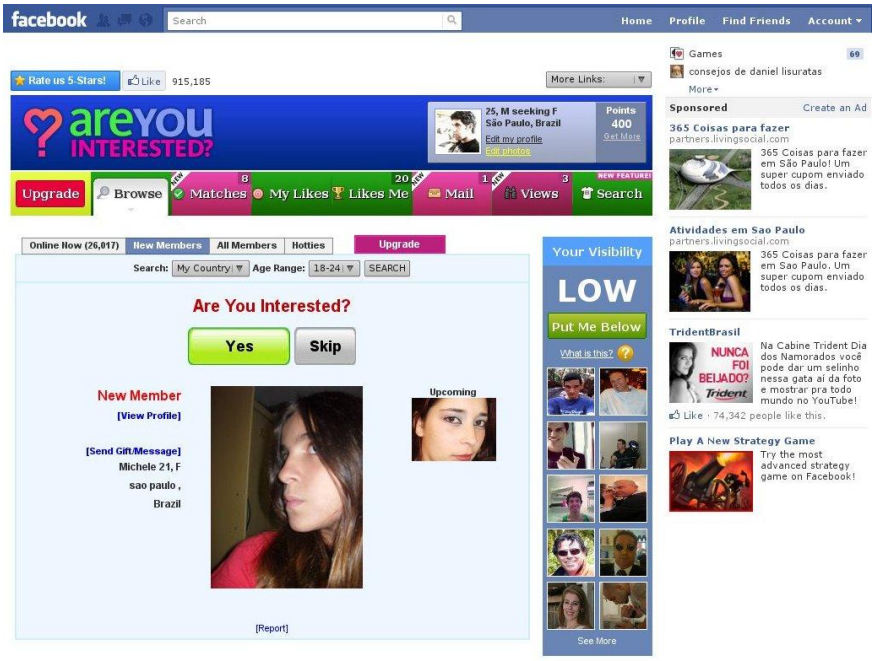

Figura 3.2: Rede social AreYouInterested?

Por último, há outra abordagem popular na internet chamada de *social aggregator*, que consiste em uma aplicação (Web, desktop ou móvel), que adiciona as contas das redes sociais das quais o usuário participa, apresentando-as como uma única rede social. Nesse tipo de abordagem, as redes sociais não interagem, o *social aggregator* envia a mesma informação para todas elas. Exemplos de *social aggregators* são a Profilactic[24], a HootSuite[25] (Figura 3.3) e a SocialHub[26].

Figura 3.3: A social aggregator HootSuite

[24] http://www.profilactic.com/
[25] http://hootsuite.com/
[26] http://socialhub.samsungmobile.com/

4 Técnicas e Tecnologias para Interoperabilidade nas Redes Sociais

Nos trabalhos encontrados na literatura e nas redes sociais exploradas, foram identificadas quatro técnicas que interoperam redes sociais recorrentemente utilizadas. São elas:

T1. Utilização de um sistema de autenticação padrão compartilhado entre serviços de redes sociais na internet. Essa técnica possibilita ao usuário utilizar vários serviços sem ter que ingressar seus dados em todos eles, bastando definir a identidade do usuário em um dos serviços para que os demais a reutilizem.

T2. Utilização de um formato padrão para representar a informação dos usuários. Possibilita que os dados trocados pelos serviços na internet possuam o mesmo significado entre todos eles.

T3. Utilização de serviços Web RESTful e uso de conteúdo de fontes externas. Essa técnica fornece a plataforma de comunicação por meio da qual as informações dos serviços de redes sociais na internet são trocadas.

T4. Definição de políticas de gerenciamento dos dados do usuário por parte do serviço de rede social. Essas políticas definem o que podem fazer os serviços de rede social com os dados fornecidos por seus usuários.

Essas técnicas estão relacionadas com as questões propostas pelo projeto Data Portability da seguinte forma (Tabela 4.1):

- As questões Q2 e Q6 estão relacionadas com a primeira técnica dado que elas fazem ênfase à utilização de um único ponto de acesso a um serviço na internet.
- As questões Q1, Q4, Q5 e Q7 estão relacionadas com a segunda técnica dado que seu objetivo é a representação e significado dos dados.
- As questões Q3, Q4 e Q5 estão relacionadas com a terceira técnica dado que tais questões têm a ver com a comunicação entre sistemas da internet.
- Finalmente as questões relacionadas com o gerenciamento de políticas sobre os dados da quarta técnica são as questões Q7, Q8, Q9 e Q10.

Que. / Técnica	T1	T2	T3	T4
Q1		X		
Q2	X			
Q3			X	
Q4		X	X	
Q5		X	X	
Q6	X			
Q7		X		X
Q8				X
Q9				X
Q10				X

Tabela 4.1: Questões cobertas pelas técnicas identificadas

As quatro técnicas também cumprem os quatro requisitos apresentados por Tiwari (2008). Os relacionamentos de cada requisito e técnica são apresentados na Tabela 4.2.

Req. / Técnica	T1	T2	T3	T4
R1	X	X	X	
R2		X	X	X
R3		X	X	X
R4		X	X	

Tabela 4.2: Requisitos cobertos pelas técnicas identificadas

Com base nessas técnicas foi realizado um estudo das tecnologias da Web 2.0 presentes nas redes sociais, visando a classificar as tecnologias segundo o suporte que oferecem.

4.1 Sistema de autenticação padrão compartilhado entre serviços na internet

Os sistemas de autenticação e autorização padrão, comumente referenciados como Single Sign-On (SSO), ou assinatura única, são definidos como um único ponto de entrada na aplicação, ou seja, um usuário apenas necessita se autenticar ou autorizar uma única vez, para, posteriormente, ter acesso automaticamente às diversas aplicações externas. O serviço de Single Sign-On centraliza a autenticação e autorização de diversas aplicações, de tal forma que os usuários não tenham que entrar mais de uma vez com suas credenciais. Na Web 2.0 e redes sociais, as tecnologias de Single Sign-On mais usadas são OpendID e OAuth, tecnologias utilizadas pelo Facebook, Twitter e Google.

4.1.1 OpenID

OpenID (2006) é um sistema de identificação desenvolvido por Brad Fitzpatrick do LiveJournal. Trata-se de uma rede distribuída na qual a identidade do utilizador é dada por uma URL ou XRI que pode ser verificada por um provedor de identidade. Em sites que dão suporte a OpenID, utilizadores não necessitam de criar uma nova conta, só é necessário autenticação por um site que dê suporte ao OpenID.

4.1.1.1 Funcionamento

Antes de descrever o processo de autenticação, vamos definir os seguintes termos:

- **Usuário final** é a entidade cuja identidade precisa ser comprovada.

- **Identificador OpenID** é a URL ou XRI escolhida pelo usuário final como nome de sua identidade.
- **Provedor de identidade ou provedor OpenID** é o serviço que se especializa em registrar URLs ou XRIs de autenticação de OpenID.
- **Parte dependente** é o site que quer verificar a identidade do usuário final.
- **Agente do usuário** é o programa utilizado pelo usuário final para se comunicar com a parte dependente e o provedor de identidade (comumente é o navegador Web).

O processo de autenticação começa quando o usuário final interage com a parte dependente, como, por exemplo, uma página Web que fornece um meio para especificar um identificador de OpenID para fazer a autenticação. Um usuário final normalmente já tem registrado um identificador OpenID (p. ex. alice.openid.example.org) em um provedor OpenID (p. ex. openid.example.org).

A parte dependente transforma o identificador OpenID em sua URL padrão (p. ex. http://alice.openid.example.org/). Com a versão OpenID 1.0, a parte dependente, em seguida, solicita o recurso HTML identificado pela URL e lê a etiqueta link do HTML para descobrir a URL do provedor de OpenID (p. ex. http://openid.example.org/openid-auth.php). Com a versão OpenID 2.0, a parte dependente descobre a URL do provedor de OpenID solicitando um documento XRDS (também chamado de documento Yadis) com o tipo de conteúdo "application/xrds+xml".

Há dois modos em que a parte dependente se comunica com o provedor de OpenID:

- **checkid_immediate**, em que a parte dependente solicita que o provedor de OpenID não interaja com o usuário final; toda a comunicação é retransmitida ao agente do usuário sem explicitamente notificar o usuário final.

- **checkid_setup**, em que o usuário final se comunica com o provedor de OpenID por meio do agente do usuário utilizado para acessar a parte dependente.

Depois, opcionalmente, a parte dependente e o provedor de OpenID estabelecem um "segredo compartilhado" (*shared secret*) referenciado por um gerenciador associado, que a parte dependente guarda. Se está sendo utilizado o modo "checkid_setup" a parte dependente redireciona o agente do usuário ao provedor de OpenID para que o usuário final possa se autenticar.

O método de autenticação varia, porém, tipicamente, um provedor de OpenID solicita ao usuário final sua senha, e depois pergunta se o usuário final confia na parte dependente para receber os detalhes de sua identidade. Se o usuário final não aceita a solicitação de confiança da parte dependente, o agente do usuário é redirecionado para a parte dependente com uma mensagem indicando que a autenticação foi rejeitada. Se o usuário final aceita a solicitação de confiança, o agente do usuário é redirecionado à parte dependente junto com as credenciais do usuário final. Então a parte dependente confirma se as credenciais realmente vêm do provedor de OpenID. Há duas formas de confirmar isso: a primeira chamada de *statefull,* que é quando a parte dependente e o provedor de OpenID previamente compartilharam um *shared secret.* A parte dependente compara sua cópia do *shared secret* com a recebida nas credenciais do usuário. A segunda é chamada de *stateless*¸ na qual a parte dependente faz mais uma requisição (*check_autentication*) para validar que os dados recebidos vieram do provedor OpenID.

Depois que o identificador de OpenID é verificado, a autenticação é considerada satisfatória e o usuário final é autenticado na parte dependente sob o identificador OpenID (Figura 4.1).

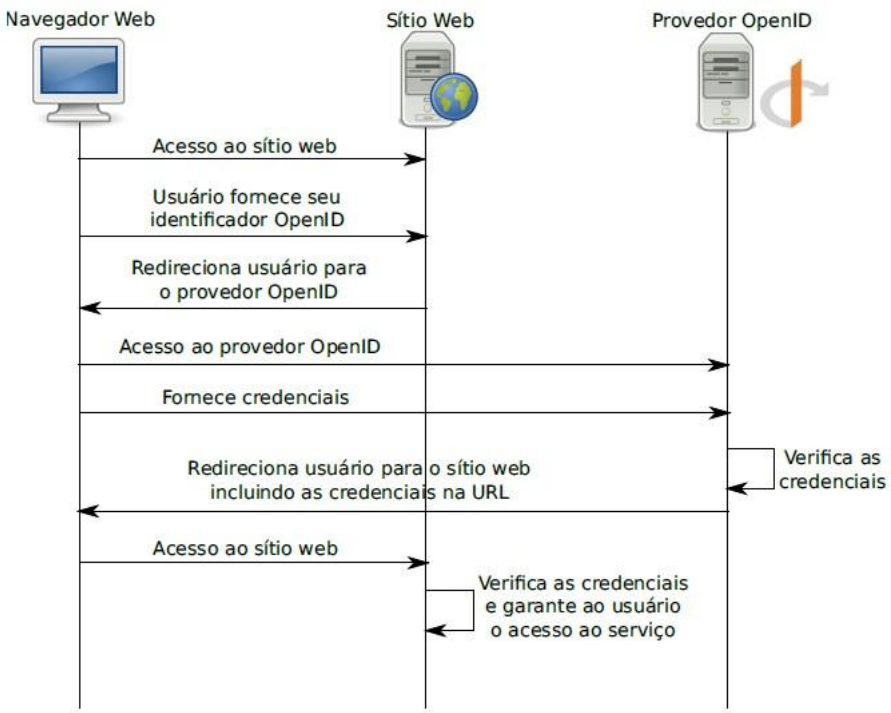

Figura 4.1: Funcionamento do OpenId

4.1.2 OAuth

OAuth (2007) é um padrão aberto para autorização. Possibilita aos usuários compartilhar recursos como fotos, vídeos e contatos, de um site para outro, sem ter que entregar suas credenciais (comumente usuário e senha). Por meio de um provedor de serviço (*service provider*), OAuth entrega *tokens* em vez de credenciais para acessar os recursos e as fontes de dados. Cada *token* possibilita acessar um conjunto de recursos de um site sem ter que compartilhar as permissões de acesso de outros conjuntos de recursos do site.

OAuth possui duas versões por enquanto, OAuth1.0a e OAuth2.0. A primeira com a especificação concluída e madura, ainda é usada no momento da escrita deste texto por muitos sites na web, como por exemplo Twitter[27], LinkedIn[28] e Flickr[29]. A segunda tem

[27] https://dev.twitter.com/docs/auth/oauth
[28] https://developer.linkedin.com/documents/authentication
[29] http://www.flickr.com/services/api/auth.oauth.html

a especificação ainda em desenvolvimento pelo *Internet Engineering Task Force* (IETF)[30], sendo o último rascunho da especificação o oauth-v2-31. OAuth2.0 é usado em vários sites como Facebook[31], Google[32] e GitHub[33]. Na sequência é apresentado o funcionamento dessas duas versões de OAuth.

4.1.2.1 OAuth 1.0a

O fluxo de utilização está dividido em duas partes. A primeira descreve como obter um *access token* e a segunda mostra como utilizar o *access token* para acessar os recursos. O fluxo apresenta os seguintes elementos:

- **Provedor do serviço** é a aplicação em que ficam os dados do usuário e gerencia a autorização sobre eles, também chamado de servidor OAuth.
- **Usuário** é a pessoa que possui uma conta e seus dados no provedor do serviço.
- **Aplicação externa** é a aplicação que solicita acesso aos dados do usuário que ficam no provedor do serviço.
- **Cliente** é um programa capaz de fazer requisições HTTP, por exemplo, um navegador web.

A primeira parte do fluxo possui os seguintes passos: (Figura 4.2).

1. O usuário por meio do cliente faz uma requisição ao provedor do serviço enviando como parâmetros, a chave de acesso da aplicação cliente (*oauth_consumer_key*), uma assinatura digital (*oauth_signature*), o método utilizado para criar essa assinatura (*oauth_signature_method*), uma marca de tempo (oauth_timestamp), e uma URL (oauth_callback) da aplicação externa, para qual o provedor de serviço redireciona depois de concluída a fase de autorização.

[30] http://www.ietf.org/
[31] https://developers.facebook.com/docs/authentication/
[32] https://developers.google.com/accounts/docs/OAuth2
[33] http://developer.github.com/v3/oauth/

2. O provedor de serviço recebe essa informação e cria um *token* sem acesso e um *token secret*. Esse *token secret* junto com o *oauth_consumer_secret* servem para assinar digitalmente as requisições durante o processo de autorização.
3. O cliente redireciona o usuário para a página de autorização do provedor do serviço, e caso o usuário não tenha uma sessão ativa, ele inicia uma sessão e é direcionando para a página de autorização.
4. Depois que o usuário autorizou, o provedor do serviço redireciona o usuário para a URL da aplicação externa (*oauth_callback*) enviando como parâmetros o *token* (*oauth_token*) e um verificador (*oauth_verifier*).
5. A aplicação realiza uma requisição ao provedor do serviço para verificar a origem e conceder as permissões do *token*, para o qual envia como parâmetros: *oauth_verifier, oauth_token*, *oauth_consumer_key*, *oauth_signature*, *oauth_signature_method*, *oauth_timestamp* e *oauth_nonce.*
6. Depois de verificar e autorizar o *token*, o provedor do serviço responde à requisição da aplicação externa com o *token* autorizado (*oauth_token*) e um novo *token secret (oauth_token_secret).* O *token secret* e o *oauth_consumer_secret* servem para assinar digitalmente todas as requisições na segunda parte do fluxo.

A segunda parte do fluxo mostra como acessar um recurso, e é formada pelos seguintes dois passos:

1. O cliente realiza uma requisição a um recurso protegido do provedor do serviço enviando como parâmetros: *oauth_consumer_key*, *oauth_token*, *oauth_signature_method*, *oauth_signature*, *oauth_timestamp* e *oauth_nonce.*
2. O provedor do serviço verifica se a assinatura digital (*oauth_signature*), o *nonce* (*oauth_nonce*) e o *timpestamp* (*oauth_timestamp*) da requisição são válidos, e em caso positivo, o provedor do serviço verifica as permissões do *token* sobre o recurso solicitado. Caso o *token* possua permissões suficientes o provedor do serviço

responde enviando o recurso solicitado, caso contrário é enviado o código HTTP 401 de acesso negado.

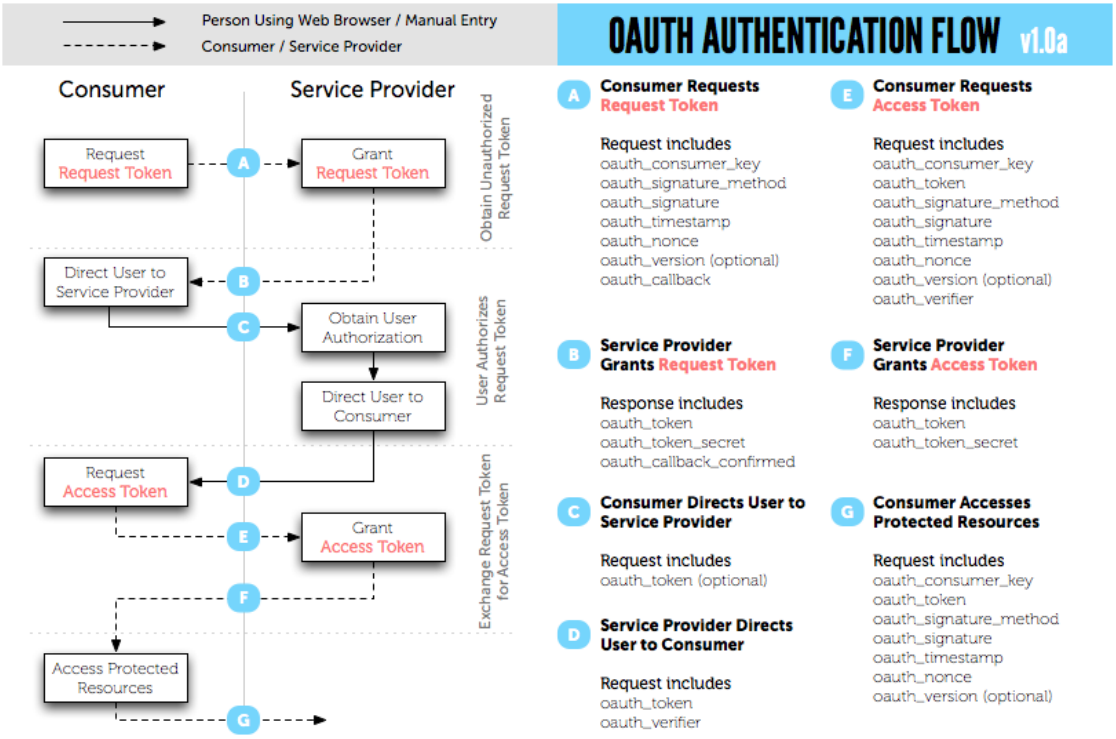

Figura 4.2: Fluxo OAuth1.0a[34]

4.1.2.2 OAuth 2.0

Diferentemente do OAuth1.0a, o OAuth 2.0 apresenta seis fluxos dependendo do tipo do cliente que o usuário está utilizando: fluxo de usuário-agente, fluxo de servidor web, fluxo de dispositivo, fluxo de usuário-senha, fluxo de credenciais do cliente e fluxo de asserção. Todos esses fluxos são obrigatoriamente montados em um canal seguro (HTTPS). Os fluxos mais importantes para o presente trabalho são o fluxo de servidor web e o fluxo usuário-agente.

[34] Imagem tomada de https://wiki.fitbit.com/display/API/OAuth+Authentication+in+the+Fitbit+API

- **Fluxo de servidor web**: É utilizado por aplicações web nas quais o servidor da aplicação cliente participa do fluxo de autenticação. Os elementos que participam do fluxo são os seguintes:
    - **Aplicação cliente** é uma aplicação web que executa num servidor capaz de atender requisições HTTP.
    - **Provedor do serviço** é a aplicação na qual ficam os dados do usuário e que gerencia a autorização sobre eles, também chamado de servidor OAuth.
    - **Usuário** é a pessoa que possui uma conta no provedor do serviço.

    O fluxo apresenta os seguintes passos: (Figura 4.3).

1. O usuário por meio da aplicação cliente realiza uma requisição com os seguintes parâmetros: *cliente_id*, que é o identificador da aplicação cliente no provedor do serviço, *redirect_uri* é a URL encarregada de processar o redirecionamento do provedor do serviço depois da autorização, e por último o parâmetro *scope*, que indica que permissões a aplicação cliente está solicitando.

2. O provedor do serviço mostra uma tela de login se o usuário não tem uma sessão ativa, e depois mostra uma tela confirmando as permissões solicitadas.

3. Se usuário aprova a solicitação das permissões, o provedor do serviço redireciona o usuário para a URL especificada pelo parâmetro *redirect_uri* da aplicação cliente contendo um código de verificação (*code*).

4. A aplicação cliente faz uma requisição enviando como parâmetros o código obtido no passo anterior, o *cliente_id*, o *client_secret* e a *redirect_uri*.

5. O provedor do serviço verifica o código enviado (*code*), o *cliente_id* e o *cliente_secret*, e redireciona para a URL especificada no *redirect_uri*, acrescentando como parâmetro o access token (acess_token).

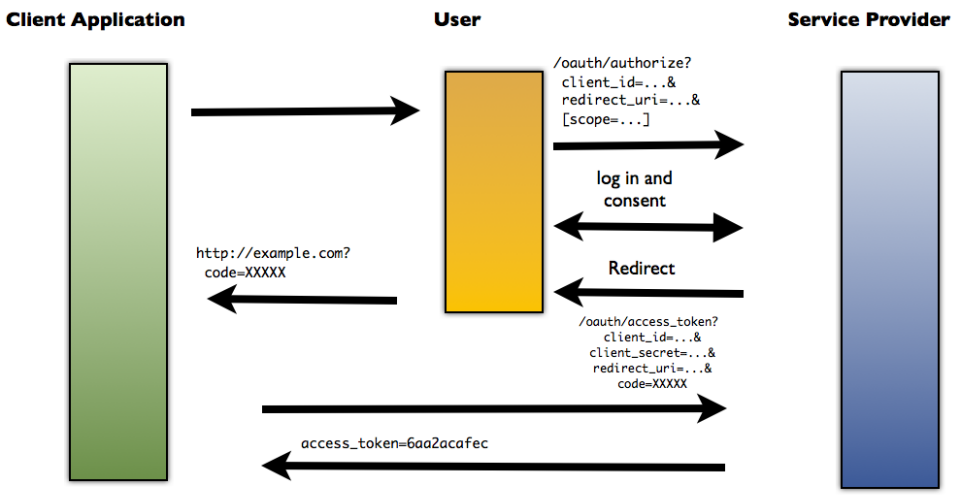

Figura 4.3: fluxo servidor web OAuth2.0[35]

- **Fluxo de usuário-agente**: É utilizado em aplicações desktop que interpretam código HTML, como um navegador web ou a aplicação TweetDeck[36]. Os elementos que participam do fluxo são os seguintes:
  - **Aplicação cliente** é uma aplicação desktop capaz de realizar requisições HTTP e interpretar HTML.
  - **Provedor do serviço** é a aplicação na qual ficam os dados do usuário e que gerencia a autorização sobre eles, também chamado de servidor OAuth.
  - **Usuário** é a pessoa que possui uma conta no provedor do serviço.

  O fluxo apresenta os seguintes passos: (Figura 4.4)

[35] Imagen tomada de http://www.sociallipstick.com/?p=239
[36] http://tweetdeck.com

1. O usuário por meio da aplicação cliente (aplicação desktop) realiza uma requisição ao provedor do serviço tendo como parâmetros o *cliente_id*, o *redirect_uri* e o *scope*, descritos no fluxo servidor web.

2. O provedor do serviço apresenta a sua tela de login para que o usuário ingresse suas credenciais, e logo apresenta a tela de autorização dos recursos.

3. Logo que o usuário concede a autorização o provedor do serviço redireciona ao usuário para a aplicação cliente colocando como parâmetro o *access token.*

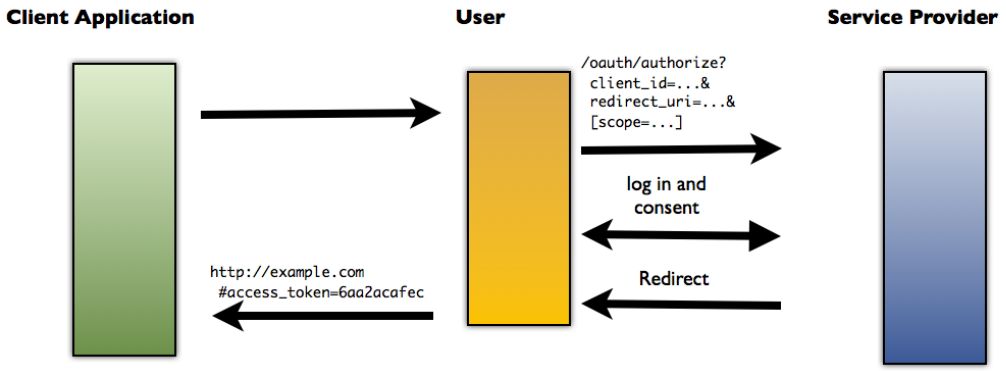

Figura 4.4: Fluxo usuário-agente OAuth2.0[37]

A utilização do *access token* é a mesma em todos os fluxos, só acrescentando o parâmetro *access_token* na URL do recurso. Dado que OAuth2.0 exige a utilização de um

[37] Tomado de http://www.sociallipstick.com/?p=239

canal seguro (HTTPS) não é necessário assinar digitalmente as requisições como feito em OAuth1.0a.

4.2 Formatos padrões para representar a informação dos usuários

Na Web 2.0 são encontrados com frequência dois formatos de intercâmbio de dados: o XML (eXtensible Markup Language)[38] e o JSON (Javascript Object Notation)[39] e também vários microformatos[40] derivados. Nas redes sociais é comum utilizar XML como base de outros formatos derivados, como RDF, RSS e ATOM[41].

4.2.1 RSS

É um formato baseado em XML, comumente utilizado para comunicar as atividades, notícias e atualizações de amigos ou aplicações. O RSS possibilita aos usuários da internet se inscrever em sites que fornecem "*feeds*" RSS. Esses sites mudam ou atualizam seu conteúdo regularmente. Para isso, são utilizados feeds RSS que recebem essas atualizações. Dessa maneira, o utilizador permanece informado de atualizações em vários sites sem precisar visitá-los (W3C, 2002).

4.2.2 RDF

O modelo de dados RDF (W3C, 2004) é similar ao utilizado em métodos de modelagem conceitual, como Entidade-Relacionamento ou diagramas de classes. Está baseado na ideia de criar sentenças sobre recursos (em particular recursos Web) na forma de sujeito-predicado-objeto. Essa expressão é conhecida como triplas na terminologia RDF. O sujeito indica um recurso e o predicado indica os aspectos do recurso, expressando uma relação entre o sujeito e o objeto. Por exemplo, uma forma de representar a sentença "O céu tem a cor azul" por meio de uma tripla RDF seria: "O céu"

---

[38] http://www.w3.org/XML/
[39] http://www.json.org/
[40] É um conjunto de formatos abertos projetados para adicionar semântica em qualquer documento XML.
[41] Atom é um protocolo para ler e escrever informações na Web

é o sujeito, “tem a cor” é o predicado, e “azul” é o objeto. RDF é um modelo abstrato representado em vários formatos.

Esse mecanismo para descrever recursos é na proposta do W3C para a Web Semântica. Uma coleção de sentenças RDF intrinsecamente representa um grafo rotulado dirigido. A seguir é apresentado um exemplo de RDF em formato XML:

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1/">
      <rdf:Description rdf:about="http://en.wikipedia.org/wiki/Tony_Benn">
            <dc:title>Tony Benn</dc:title>
            <dc:publisher>Wikipedia</dc:publisher>
      </rdf:Description>
</rdf:RDF>
```

4.2.3 vCard

vCard é um formato para intercâmbio de informação pessoal. As vCards (VERSIT CONSORTIUM, 1996) são usualmente anexadas nas mensagens de e-mail e o seu formato é representado em XML ou em texto puro, podendo ser adicionado a um documento RDF. As vCards, inicialmente chamadas de Versitcard, foram propostas pela Versit Consortium, grupo formado pela Apple, AT&T, IBM e Siemens. Em 1996, o formato foi entregue ao Internet Mail Consortium[42]. A última versão do formato é a 3.0. Comumente, as vCards possuem extensão “.vcf” e podem ser representadas em documentos RDF. Também são utilizadas para mensagens instantâneas, como é o caso do protocolo XMPP[43].

[42] http://www.imc.org/
[43] http://xmpp.org/

4.2.4 Friend of a Friend

FOAF (FOAF PROJECT, 2000) é uma ontologia[44] para ser interpretada por máquinas, que tem por objetivo descrever pessoas, suas atividades e relações com outras pessoas e objetos. FOAF é composto por um vocabulário expresso por meio de RDF (Resource Description Framework) e OWL (Web Ontology Language)[45]. Os perfis FOAF são usados para encontrar, por exemplo, todas as pessoas morando na Europa, ou todos os amigos em comum com outras pessoas. Isso é conseguido por meio da definição de relações entre pessoas. Cada perfil tem um único identificador (como e-mail, jabberID[46] ou URL da página pessoal) usado para definir essas relações.

O projeto FOAF começou em 2000 e foi considerada a primeira aplicação social para Web semântica, uma vez que combina a tecnologia RDF e as preocupações da Web social. Na Figura 4.1 é apresentado um exemplo de documento FOAF.

[44] Uma ontologia é um esquema conceptual dentro de um domínio de dados com a finalidade de facilitar a comunicação e a troca de informação entre diferentes sistemas e entidades.
[45] http://www.w3.org/2004/OWL/
[46] http://www.jabber.org/

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

<#JW>
   a foaf:Person ;
   foaf:name "Jimmy Wales" ;
   foaf:mbox <mailto:jwales@bomis.com> ;
   foaf:homepage <http://www.jimmywales.com/> ;
   foaf:nick "Jimbo" ;
   foaf:depiction <http://www.jimmywales.com/aus_img_small.jpg> ;
   foaf:interest <http://www.wikimedia.org> ;
   foaf:knows [
      a foaf:Person ;
      foaf:name "Angela Beesley"
   ] .

<http://www.wikimedia.org>
   rdfs:label "Wikipedia" .
```

Figura 4.1: Documento FOAF

4.2.5 Activity Stream

Um fluxo de atividades (*activity stream*) é uma lista de atividades recentes feitas por uma pessoa dentro de uma página web, por exemplo, o componente "News Feed" do Facebook é um fluxo de atividades. Desde a adoção do componente "News Feed" pelo Facebook, outros sítios na internet têm produzido suas próprias versões, provocando a proliferação de diferentes fluxos de atividade na internet. O projeto Activity Streams (2011) tem por objetivo desenvolver um protocolo de fluxo de atividades para distribuir atividades por meio de aplicações web na internet.

O protocolo Activity Streams é uma especificação aberta que consiste de um conjunto atividades (*activities*). Uma atividade (*activity*) é formada por um ator (*actor*), um verbo (*verb*), um objeto (*object*) e um alvo (*target*). A atividade descreve uma ação realizada por uma pessoa ou por um objeto, por exemplo: "Geraldine enviou uma foto a seu álbum" ou "João compartilhou um vídeo".

O objetivo da especificação Activity Streams é providenciar suficientes metadados acerca de uma atividade, de modo que um consumidor possa apresentar ao usuário

a informação num formato humanamente legível. Isso inclui a construção de sentenças claras acerca da atividade que aconteceu, representações visuais da atividade ou combinação de atividades similares para serem apresentadas.

Na especificação do Activity Streams, um objeto é uma coisa, real ou imaginária, que participa de uma atividade. O objeto é a entidade que realiza a ação ou a entidade na qual é realizada a ação. Essa especificação utiliza JSON para a representação do fluxo de atividades. Um exemplo desse fluxo de atividades é apresentado na Figura 4.5.

```
{
  "items" : [
    {
      "published": "2011-02-10T15:04:55Z",
      "foo": "some extension property",
      "generator": {
        "url": "http://example.org/activities-app"
      },
      "provider": {
        "url": "http://example.org/activity-stream"
      },
      "title": "Martin posted a new video to his album.",
      "actor": {
        "url": "http://example.org/martin",
        "objectType": "person",
        "id": "tag:example.org,2011:martin",
        "foo2": "some other extension property",
        "image": {
          "url": "http://example.org/martin/image",
          "width": 250,
          "height": 250
        },
        "displayName": "Martin Smith"
      },
      "verb": "post",
      "object" : {
        "url": "http://example.org/album/my_fluffy_cat.jpg",
        "objectType": "photo",
        "id": "tag:example.org,2011:my_fluffy_cat",
        "image": {
          "url": "http://example.org/album/my_fluffy_cat_thumb.jpg",
          "width": 250,
          "height": 250
        }   },
      "target": {
        "url": "http://example.org/album/",
```

```
"objectType": "photo-album",
"id": "tag:example.org,2011:abc123",
"displayName": "Martin's Photo Album",
"image": {
  "url": "http://example.org/album/thumbnail.jpg",
  "width": 250,
  "height": 250
}  }   }  ] }
```

Figura 4.5: Formato Activity Streams

4.3 Serviços Web RESTful e uso de conteúdo de fontes externas

Os sites de redes sociais criaram plataformas de desenvolvimento de pequenas aplicações, possibilitando que terceiros criem suas próprias aplicações para esses sites. Essas aplicações, conhecidas como mashups, utilizam conteúdos locais e externos para fornecer um novo serviço dentro do site. As plataformas utilizadas pelos mashups normalmente são baseadas em serviços Web RESTful.

4.3.1 Serviços Web RESTful

Uma API baseada em serviços Web é a abordagem mais popular das aplicações na internet para atingir interoperabilidade. O W3C define um serviço Web como um sistema de software projetado para comunicação máquina a máquina sobre uma rede (W3C, 2004). A comunicação é geralmente é feita acima do protocolo HTTP com serialização.

Atualmente na internet há duas grandes classes de implementação de serviços Web: os serviços Web baseados em SOAP e os serviços Web baseados em REST. Os serviços Web baseados em SOAP utilizam protocolos tais como SOAP, WSDL, UDDI, entre outros. Essa classe de serviços Web ainda impõem muitos desafios em ambientes heterogêneos e que requeiram o uso de protocolos mais avançados, como por exemplo, WS-CDL ou o BPEL4WS. Os serviços Web baseados em REST ou RESTful são mais simples e possuem curvas de aprendizado mais suaves. Essa opção de interoperabilidade usa a simplicidades do HTTP e das URIs (Uniform Resource Identifier) para expor serviços em aplicações Web. Uma vantagem dos serviços RESTful é que eles não implicam

no uso de XML para estruturação do conteúdo transmitido. Os serviços Web RESTful podem dar suporte a formatos mais leves como, por exemplo, JSON, comum em aplicações AJAX.

Um conceito importante nas arquiteturas REST e serviços Web RESTful é a existência de recursos (elementos de informação) acessíveis por meio de um identificador global (URI). Para manipular esses recursos, os componentes da rede (clientes e servidores) comunicam-se mediante uma interface padrão (HTTP) e trocam representações desses recursos (MENDES, 2010). Um exemplo é apresentado na Figura 4.6 em que um cliente HTTP usa requisições de tipo GET para acessar os serviços Web RESTful disponibilizados pela aplicação Web.

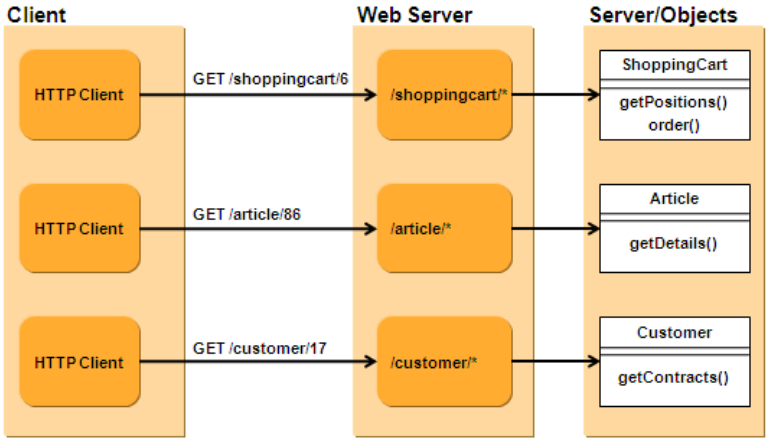

Figura 4.6 Exemplo de serviços RESTful[47]

No escopo das redes sociais, os serviços Web mais utilizados são os baseados em REST. Esses serviços ajudam a interoperar a rede social com aplicações desenvolvidas por terceiros, como aplicações desktop, aplicações em dispositivos móveis ou aplicações Web. Para isso a plataforma da rede social disponibiliza uma API expondo serviços para o

[47] Tomado de http://predic8.com/rest-webservices.htm

acesso e manipulação do grafo social. Exemplos desse tipo de API é o OpenSocial e Facebook Plataform, ilustrado na Figura 4.7.

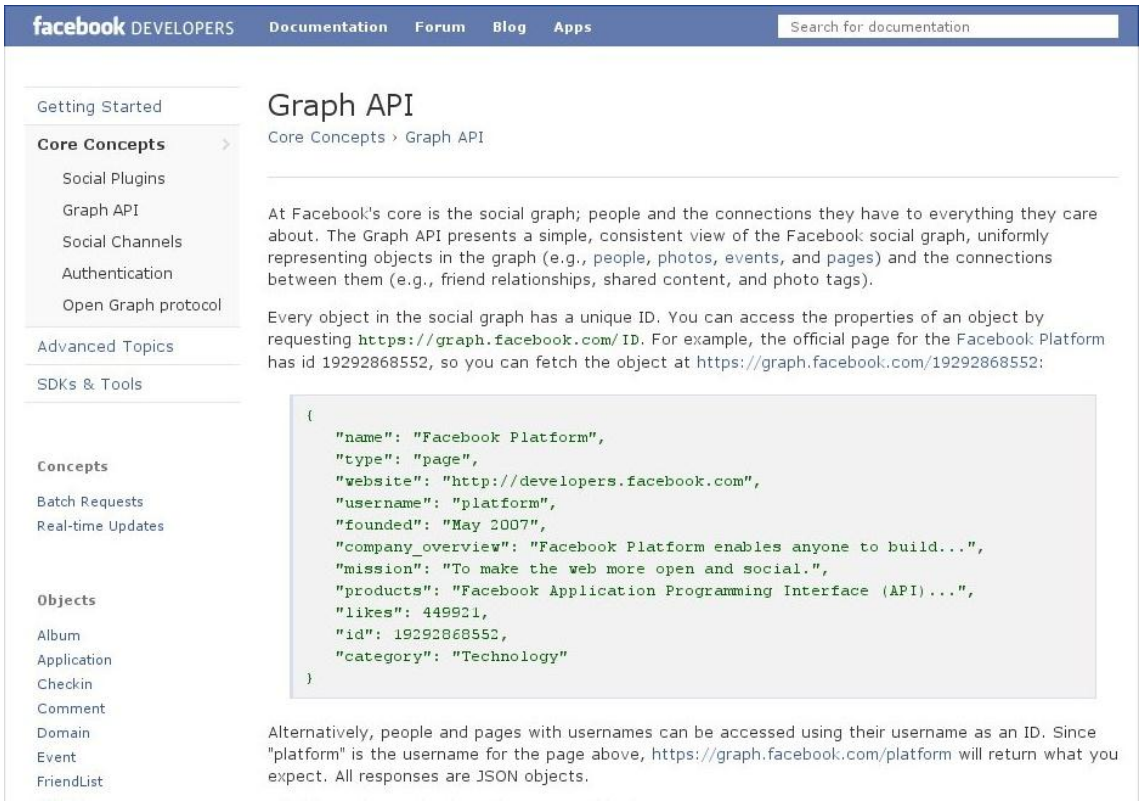

Figura 4.7: API RESTful do Facebook.

4.4 Definição de políticas de gerenciamento e segurança dos dados do usuário

Atualmente não há uma tecnologia totalmente dedicada para gerenciar as políticas sobre as informações dos usuários, dado que essas políticas são dependentes do objetivo e estrutura do serviço de rede social. As políticas de gerenciamento podem ser restritivas, como no caso do Facebook, no qual os usuários do serviço de rede social só podem enxergar os dados dos usuários amigos, ou abertas, como no caso do Twitter, no qual qualquer usuário enxerga os dados de outro usuário sem necessidade de relação entre eles.

Essas politicas também definem o que o serviço de rede social pode fazer com os dados do usuário. Um exemplo é a concessão de direitos autorais que algumas redes sociais fazem com os dados dos seus usuários, podendo vender os dados aos anunciantes de produtos e serviços, ou também tornado os dados de domínio público sob uma licença como "Creative Commons BY"[48].

Uma tecnologia que ajuda parcialmente a definir essas políticas é OAuth, que possibilita gerenciar só as políticas relacionadas às informações geradas ou consumidas por aplicações externas, como é o caso das plataformas do Facebook e OpenSocial, que fornecem um conjunto de permissões a serem utilizadas pelas aplicações externas para ler ou escrever dados no serviço de rede social.

[48] A licença "Creative Commons BY" permite copiar, exibir, distribuir e executar uma obra.

5 Componentes do Groupware Workbench para prover interoperabilidade

Com o objetivo de dar suporte às técnicas apresentadas anteriormente, foi proposto um conjunto de cinco componentes. Esses componentes foram implementados seguindo o modelo de componentes do Groupware Workbench (2010), apresentado na seção a seguir.

5.1 Groupware Workbench

As aplicações desktop foram impulsionadas pelo advento das interfaces gráficas com o usuário e dos toolkits de widgets, que possibilitaram a programadores casuais construir aplicações arrastando e configurando componentes. Ao conceber e desenvolver os serviços de um sistema colaborativo na forma de componentes de software, o desenvolvedor monta um groupware específico para as necessidades de colaboração. As ferramentas são selecionadas de um kit de componentes para apoiar a dinâmica estabelecida (GREENBERG, 2007).

Na abordagem proposta pelo Groupware Workbench, os componentes são organizados em função do modelo 3C de colaboração (comunicação, coordenação e cooperação) (ELLIS, GIBBS e REIN, 1991) (FUKS, RAPOSO, *et al.*, 2007). Ao enxergar o problema sob a perspectiva do modelo 3C e utilizar a componentização organizada em função desse modelo, as alterações na colaboração são mapeadas ao suporte computacional, que é substituído ou acrescentado na medida da necessidade.

Para dar suporte ao desenvolvimento de sistemas colaborativos, foram estabelecidos dois níveis de componentização. O primeiro nível é constituído de

ferramentas colaborativas que, por sua vez, são montadas com componentes que implementam funcionalidades específicas relacionadas à colaboração. Esses componentes são distribuídos em kits de componentes organizados em função do modelo 3C para que desenvolvedores montem aplicações colaborativas. Nessa abordagem, cada ferramenta usa componentes de comunicação, coordenação e de cooperação independentemente da sua classificação 3C (GEROSA, RAPOSO, *et al.*, 2006).

O Groupware Workbench estrutura as aplicações nele desenvolvidas utilizando componentes chamados collablets, que representam uma parte da aplicação, podendo ser acoplados e desacoplados uns dos outros em tempo de implantação ou mesmo de execução. Cada collablet é composto por: entidades, uma implementação da interface de negócio (business) e controladores (controllers). Na camada de apresentação, utilizam-se páginas Java Server Pages (JSP) e *widgets* de interface gráfica codificados como tagfiles (tagfiles são trechos de código JSP reusáveis armazenados em arquivos com a extensão .tag) (MICHALSKY, 2012).

Toda aplicação tem um collablet principal, que representa o ponto de entrada da aplicação e gerencia todos os demais componentes por meio de subordinação e dependência. A subordinação define a estrutura de navegação da aplicação. Um collablet é subordinável a apenas um único outro collablet, no entanto, é possível que ele tenha vários subordinados, sendo que não é possível ser subordinado a si mesmo e nem criar ciclos de subordinação.

A dependência entre collablets acontece quando um necessita de outro para operar corretamente. Há dois tipos de dependências: fortes, em que um collablet não opera sem suas dependências, e fracas, em que pode operar sem as suas dependências.

A arquitetura técnica das aplicações do Groupware Workbench é similar a uma arquitetura Web em geral, conforme ilustrado na Figura 5.1. Os Widgets são utilizados por outros componentes para a integração no nível de apresentação. Os JSPs

implementam a visualização do componente e os recursos são arquivos auxiliares (imagens, scripts etc.). Os controladores são responsáveis por receber e processar os dados provenientes da camada de apresentação, efetuando despachos para a camada de lógica de negócio. Nessa camada, fica a interface Business, que é responsável pela comunicação do componente com o Groupware Workbench; a classe ComponentMgrInstance, que implementa a instância do componente; e a classe Entity, que contém os atributos dos componentes e seus comportamentos. Por fim, os dados são acessados e persistidos no banco de dados, que guarda as informações dos componentes e do arcabouço Groupware Workbench.

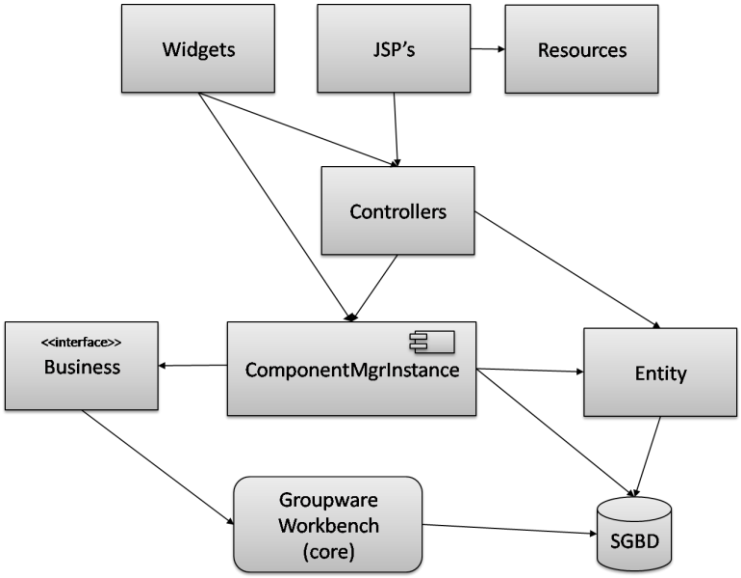

Figura 5.1: Arquitetura de Serviço do Groupware Workbench (MICHALSKY, 2012)

Como exemplo é descrita a implementação da funcionalidade Comentário instanciada como o collablet CommentMgr, apresentado na Figura 5.2. A classe CommentMgrInstance é aquela que implementa a interface Business. Sendo o Object o objeto que está sendo comentado, User, o usuário que está efetuando o comentário, o Collablet, instância do collablet que está gerando os comentários e, finalmente, Comment que é o objeto resultante.

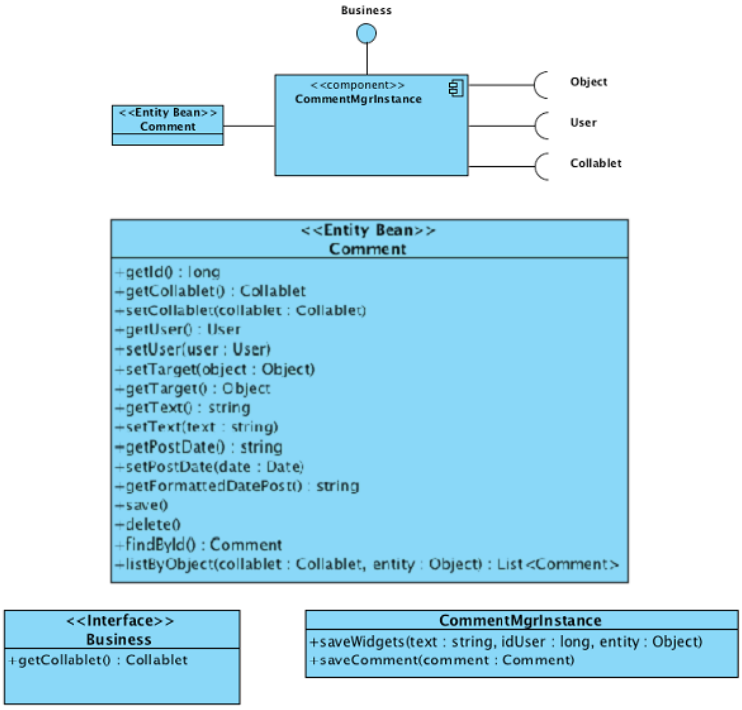

Figura 5.2: Componente CommentMgr

O CommentMgr implementa dois widgets de interface, um para atribuir um comentário de um usuário a um conteúdo e outro que retorna a lista de comentários atribuídas a esse conteúdo. Os widgets são inseridos na página JSP do Collablet, como, por exemplo, um gerenciador de fotos, adicionando os fragmentos JSTL exemplificados na Figura 5.3.

```
        <comment:addComment commentMgr="${commentMgr}"
idObject="${photo.id}" user="${sessionScope.userLogin}" />

        <comment:getComments commentMgr="${commentMgr}" entity="${photo}"
wrapClass="comments_show_internal" />
```

Figura 5.3: Fragmento JSTL do widget CommentMgr

A parte inferior da Figura 5.4 exemplifica o uso dos widgets do CommentMgr em um sistema colaborativo, a rede social para compartilhamento de imagens Arquigrafia. Juntamente com o componente CommentMgr estão compondo essa interface os collablets tag, binomialMgr, descriptionMgr, rating e photoMgr.

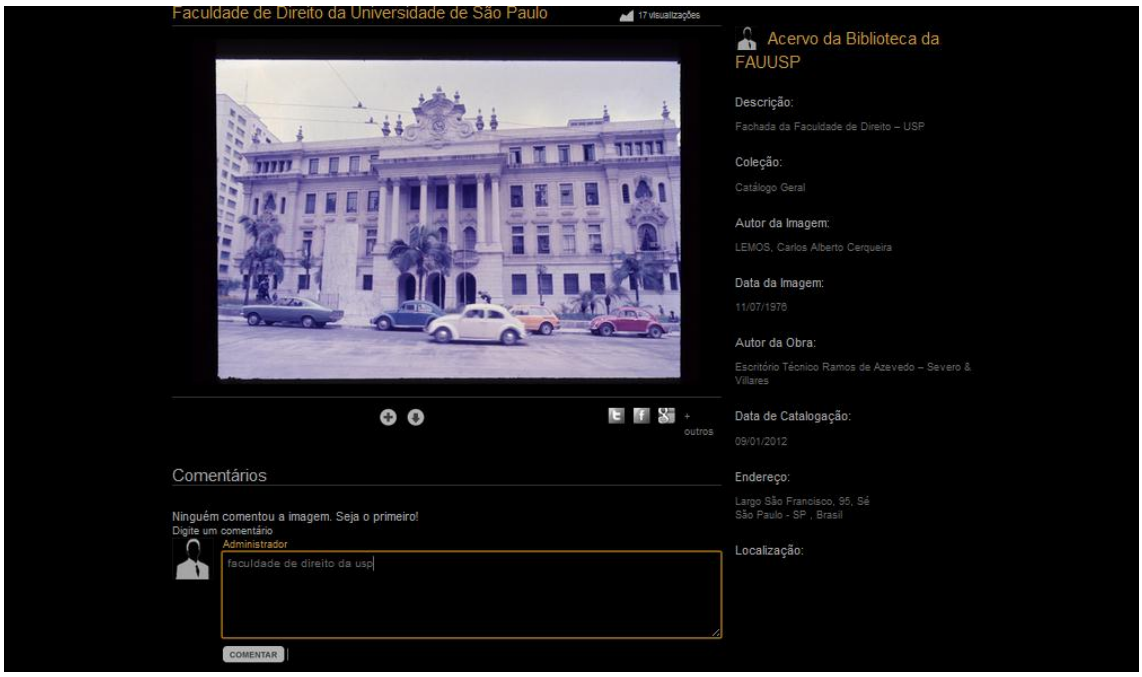

Figura 5.4: Componente de comentário no Arquigrafia

O Groupware Workbench é software livre e está disponível no site www.groupwareworkbench.org.br.

5.2 Descrição dos componentes de interoperabilidade

Os cinco componentes propostos objetivam oferecer suporte às questões relacionadas com interoperabilidade técnica e semântica do projeto Data Portability. Dado que a interoperabilidade operacional é definida pelas regras de negócio e objetivos de cada rede social e devido ao escopo do presente trabalho, as perguntas relacionadas à interoperabilidade operacional não foram analisadas. Os componentes propostos são os seguintes:

Single Sign-On: Esse componente visa a reusar as contas de outras redes sociais e associar as contas externas a uma conta criada na rede social local. Esse componente foi criado utilizando as APIs de autenticação e autorização fornecidas pelas redes sociais. O componente implementa a parte *consumer* da especificação OAuth1.0a, obtendo assim acesso ao perfil dos usuários em outras redes sociais.

A arquitetura do componente segue o modelo fornecido pelo Groupware Workbench apresentado na Figura 5.1. O seguinte diagrama de classes (Figura 5.5) mostra a estrutura interna do componente.

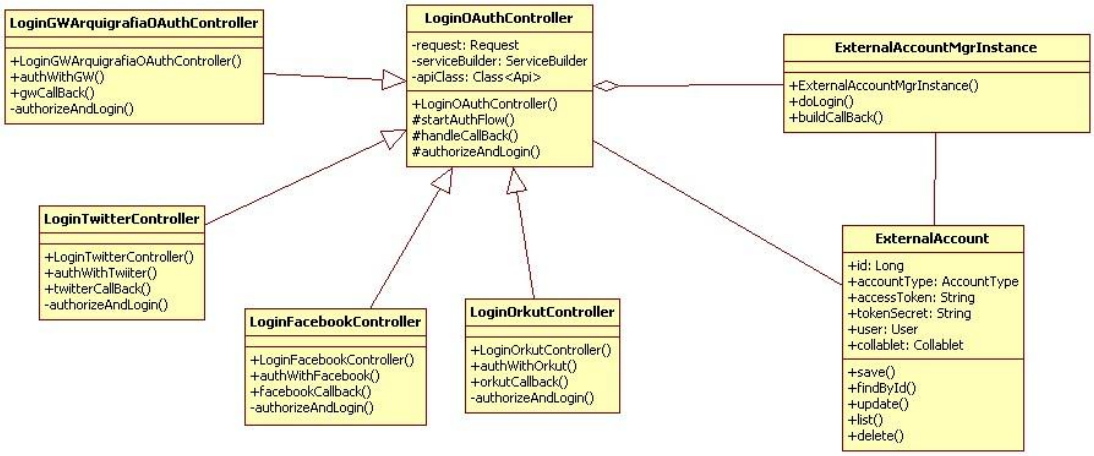

Figura 5.5: Diagrama de classes do componente Single sign-on[49]

O componente possibilita a utilização da conta de quatro redes sociais: Facebook, Orkut, Twitter e Arquigrafia, representadas cada uma com uma classe *controller* da Figura 5.5. Esses *controllers* herdam de um *controller* abstrato chamado "LoginOAuthController" que possui os elementos básicos para realizar a comunicação com as redes sociais. A classe "ExternalAccountMgrInstance" possui o método chamado de "doLogin", que registra um novo usuário e associa um objeto de tipo "ExternalAccount", que armazena o "accessToken" e "tokenSecret", para depois acessar os recursos do usuário nas redes sociais externas.

[49] Os parâmetros dos métodos das classes foram tirados por motivos de apresentação.

Login Service: O componente tem por objetivo fornecer o serviço de *Single Sign-On* para que os usuários possam reusar sua conta em outros sistemas de redes sociais, eliminando a necessidade de criar uma nova conta. O componente está baseado na parte *service provider* da especificação OAuth1.0a apresentada na seção 4.1.2.1. Portanto, os sistemas *desktop* ou web que sejam compatíveis com a especificação OAuth1.0a poderão reusar uma conta gerenciada por esse componente. A arquitetura básica do componente é apresentada no diagrama de classes da Figura 5.6.

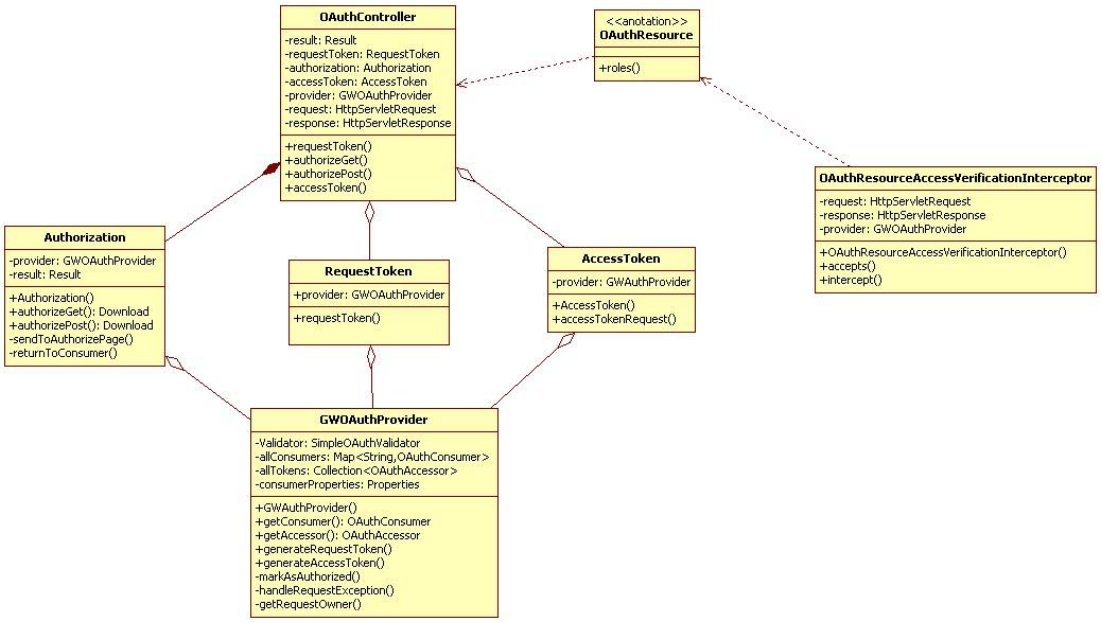

Figura 5.6: Diagrama de classes do componente Login Service

O funcionamento do componente está dividido em duas fases, a fase de autorização e a fase de controle de acesso. A fase de autorização tem por objetivo obter um *access token*, a classe principal nesta fase é o *controller* "OAuthController", que possui métodos para solicitar um *access token*, autenticar um usuário e validar o *access token*. Os *access tokens* gerados pelo *controller* são armazenadas num objeto da classe "GWAuthProvider" para sua posterior utilização. A segunda fase do funcionamento do

componente é composta pela anotação "OAuthResource" e pelo *interceptor* "OAuthResourceAccessInterceptor". A anotação "OAuthResource" marca os métodos dos *controllers* para serem gerenciados pelo OAuth, e o interceptor "OAuthResourceAccessInterceptor" é quem identifica os métodos marcados e executa ou não o método segundo as permissões do usuário armazenadas numa instancia de "GWOAuthProvider".

Add Friends: Esse componente possibilita visualizar os amigos de outras redes sociais e oferece suporte ao envio de solicitações de amizade. O componente utiliza três redes sociais para enviar os convites, Facebook, Orkut e Twitter, mas sua estrutura é extensível para outras redes sociais. A Figura 5.7 mostra a estrutura interna do componente.

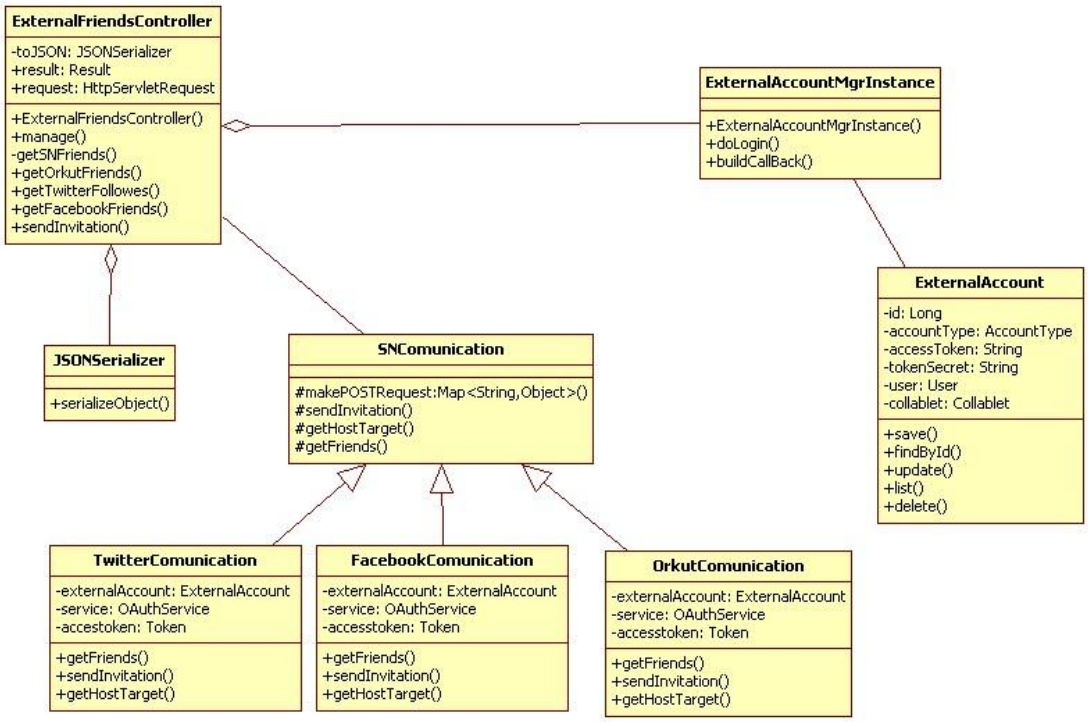

Figura 5.7 Diagrama de classes do componente AddFriends

O componente inicia o funcionamento com a classe ExternalFriendsController por meio da qual são requisitados os conjuntos de amigos das outras redes sociais. Também é encarregada de enviar o convite de amizade para um usuário da rede social. Para oferecer essas funcionalidades, ExternalFriendsController utiliza os dados das contas externas de um usuário cadastrado localmente, obtidas por meio da classe ExternalAccountMgrInstance, e também a configuração para a comunicação com cada uma das redes sociais externas, representadas pela classe abstrata SNCommunication. Essa classe define os elementos básicos de comunicação para qualquer rede social. As classes TwitterCommunication, FacebookCommunication e OrkutCommunication definem os parâmetros específicos para realizar a comunicação, como URLs das APIs e estrutura JSON para codificação e decodificação. Por último, o *controller* ExternalFriendsController utiliza uma classe auxiliar para codificar em JSON as respostas dos seus métodos.

Activity Feed: Tem por objetivo exportar o relatório das atividades a outras redes sociais. Esse componente utiliza o formato Activity Stream apresentado na Seção 4.2.5, para comunicar as atividades feitas pelo usuário dentro da rede social local. O "Activity Feed" está inicialmente ligado ao componente CommentsMgr do qual extrai as ações, que no caso desse componente são comentários realizados pelo usuário. A Figura 5.8 apresenta o diagrama de classes do componente Activity Feed.

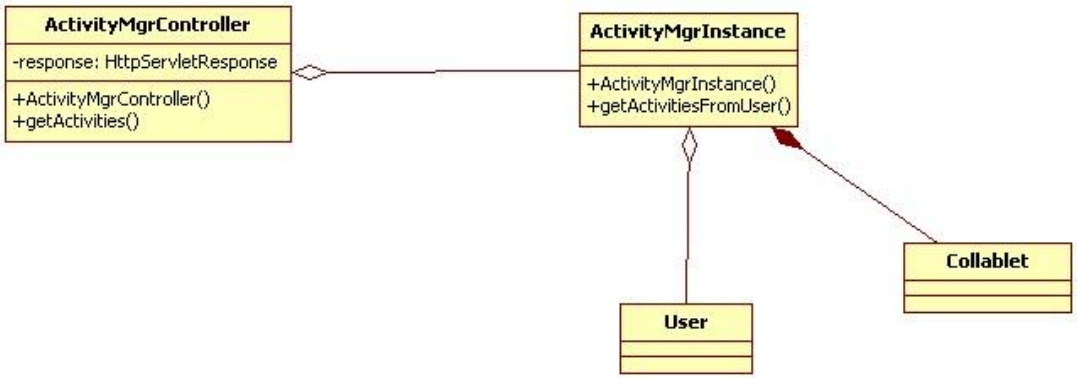

Figura 5.8: Diagrama de classes do componente Activity Feed

A estrutura do componente é formada pelo *controller* ActivityMgrController, que é o encarregado de comunicar as ações capturadas pelo componente, e formatá-las segundo o padrão ActivityStream. A classe ActivityMgrInstance é a responsável pelo enlace com outros componentes e a relação das ações com os usuários.

ProfileExport: O componente visa a exportar o perfil e grafo social dos usuários a outros serviços da internet, utilizando a ontologia FOAF, descrita na Seção 4.2.4. A representação FOAF do perfil e grafo social está codificada nos formatos XML e JSON para sua transmissão via HTTP. O diagrama de classes que representa o componente é apresentado na Figura 5.9.

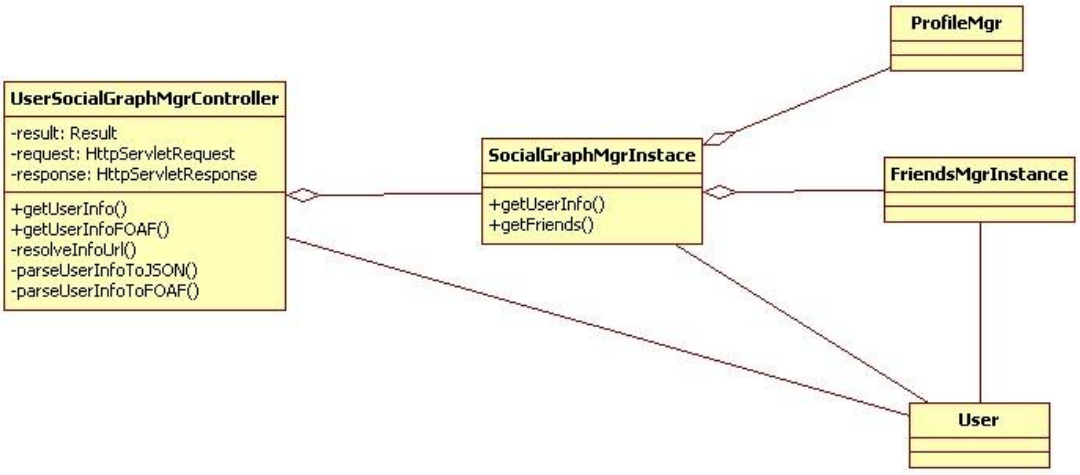

Figura 5.9: Diagrama de classes do componente ProfileExport.

O componente é formado por duas classes principais. O *controller* UserSocialGraphMgrController é encarregado de realizar a comunicação com outros sistemas e a codificação dos dados nos formatos XML e JSON. A classe SocialGraphMgrInstance é encarregada de procurar o usuário e obter seu perfil e grafo social, por meio do componente FriendsMgrInstance. Por último, essa classe transmite as informações para o *controller* UserSocialGraphMgrController para que sejam disponibilizadas para outros sistemas.

6 Avaliação no Arquigrafia

Para fins de prova de conceito, os componentes foram implantados numa rede social sem interoperabilidade tornando-a uma rede social com interoperabilidade técnica e semântica. A rede social escolhida para tal fim foi a rede social Arquigrafia, que é construída utilizando componentes do Groupware Workbench, facilitando a implantação dos novos componentes de interoperabilidade. Junto com Arquigrafia foram escolhidas as redes sociais Facebook, Orkut e Twitter dado que, segundo o IBOPE (2010), são as redes sociais mais utilizadas no Brasil, e é muito provável que o público-alvo do Arquigrafia seja parte de uma dessas redes sociais.

Facebook, Orkut e Twitter possuem esquemas de interoperabilidade parecidos, os três disponibilizam seus dados por meio de serviços web. No caso do Facebook e Twitter são utilizados serviços web RESTFul, e no caso do Orkut são utilizados serviços web baseados em RPC. Os três serviços de redes sociais utilizam OAuth para gerenciar as permissões sobre os serviços web que são disponibilizados. A seguir são apresentados o Arquigrafia e cada um dos serviços de redes sociais com os quais foi montada a prova de conceito.

6.1 Rede Social Arquigrafia

Em parceria com a Faculdade de Arquitetura e Urbanismo da USP (FAU-USP), utilizando os componentes do Groupware Workbech, está em construção uma rede social para o estudo colaborativo da arquitetura brasileira por meio de imagens. A Figura 6.1 apresenta a página de entrada da rede social Arquigrafia que está com sua versão beta disponível em http://www.arquigrafia.org.br. Essa rede possibilitará compor um amplo acervo público digital de imagens originais da arquitetura brasileira, sendo que as imagens

serão cedidas e catalogadas pelos próprios usuários, conforme dados de registro, palavras-chave e conceitos plástico-espaciais para sua inserção, acervamento e apresentação (ROZESTRATEN, MARTINEZ, *et al.*, 2010). O banco de imagens poderá ser usado nas aulas e nos trabalhos acadêmicos e a partir da leitura coletiva feita pelos usuários, será possível identificar tendências, estilos, grupos de interpretação etc. não previstos inicialmente.

Figura 6.1: Protótipo do Arquigrafia

No momento da escrita deste texto, o Arquigrafia possui catorze componentes diretamente relacionados com redes sociais, a saber: UserMgr, ProfileMgr, CommentMgr, GeoReferenceMgr, TagMgr, BinomialMgr, RatingMgr, RoleMgr, ArquigrafiaMgr, MailMgr, CounterMgr, FriendMgr, PhotoMgr, AlbumMgr.

- **UserMgr:** É encarregado do cadastro e acesso dos usuários.
- **ProfileMgr**: É encarregado de gerenciar as informações do perfil do usuário

- **CommentMgr:** Tem por objetivo coletar comentários sobre as fotos da rede social.
- **GeoReferenceMgr:** Mostra a geo-localização de uma fotografia previamente cadastrada.
- **TagMgr:** Possibilita cadastrar rótulos nas fotografias da rede social.
- **BinomialMgr:** Encarregado de cadastrar a opinião de um usuário sobre uma fotografia por meio de caraterísticas arquitetônicas.
- **RatingMgr:** Possibilita avaliar o gosto de um usuário por uma fotografia outorgando um valor inteiro de um até cinco.
- **RoleMgr:** Gerencia os papéis e permissões que um usuário tem na rede social.
- **ArquigrafiaMgr:** É o componente que representa à rede social toda, e do qual todos os outros componentes dependem.
- **MailMgr:** Envia vários tipos de notificações por correio eletrônico aos usuários da rede social.
- **CounterMgr:** Mantém um registo do número de visitas em um objeto.
- **FriendMgr**: Gerencia os vínculos entre usuários da rede social, criando o grafo social.
- **PhotoMgr**: Esse componente é encarregado de gerenciar as fotografias enviadas pelos usuários, podendo definir fotografias privadas, visíveis para amigos e totalmente públicas.
- **AlbumMgr**: Gerencia coleções de fotografias e, da mesma forma que o componente PhotoMgr, o usuário pode definir álbuns privados, visíveis para amigos ou públicos.

Antes deste trabalho, o Arquigrafia não possuía interoperabilidade com outra rede social e as informações dos usuários eram isoladas dentro da rede social. O Arquigrafia também é software livre e está disponível em http://code.google.com/p/gw-arquigrafia.

6.2 Facebook

Facebook é um serviço de rede social na internet que foi lançado em 4 de fevereiro de 2004, operado pela Facebook Inc. Em fevereiro de 2012, o Facebook tinha mais de 845 milhões de usuários ativos (PROTALINSKI, 2012). Os usuários se registram e, após isso, criam um perfil pessoal, adicionam outros usuários como amigos e trocam mensagens. Além disso, os usuários participam de grupos de interesse, organizados por escola, trabalho ou faculdade, ou outras características, e categorizam seus amigos em listas como "as pessoas do trabalho" ou "amigos íntimos".

O Facebook possibilita interagir com outros serviços na internet mediante dois módulos: o "Graph API" e o módulo "Authentication".

6.2.1 Graph API

Apresenta uma visão simples e consistente do grafo social do Facebook, representando uniformemente objetos no grafo (pessoas, fotos, eventos e páginas) e as conexões entre elas (as relações de amigos, conteúdo compartilhado e tags de fotos). O Graph API também oferece um conjunto de serviços web RESTful que possibilita recuperar e inserir informações dentro do grafo social do Facebook.

Todo objeto dentro da Graph API tem um identificador único para acessar o objeto. Por exemplo, na url "https://graph.facebook.com/19292868552" o identificador único é 19292868552, que corresponde à pagina oficial do "Facebook Platform". O resultado de solicitação utilizando o método GET é mostrado na Figura 6.2.

```
{
   "name": "Facebook Platform",
   "website": "http://developers.facebook.com",
   "username": "platform",
   "founded": "May 2007",
   "company_overview": "Facebook Platform enables anyone to build...",
   "mission": "To make the web more open and social.",
   "products": "Facebook Application Programming Interface (API)...",
   "likes": 449921,
   "id": 19292868552,
   "category": "Technology"
}
```


Figura 6.2 Informações da pagina do Facebook Platform

O Graph API trabalha com os seguintes objetos:

- **Achievement(Instance):** Instância para um objetivo atingido por um usuário
- **Album:** Um álbum de fotos
- **Application:** Uma aplicação registrada no Facebook Platform
- **Checkin:** Uma verificação de localidade
- **Comment:** Um comentário
- **Domain:** Um domínio de uma página web
- **Event:** Um evento
- **FriendList:** Uma lista de amigos do Facebook
- **Group:** Um grupo do Facebook
- **Insights:** Estatísticas sobre aplicações, páginas ou domínios.
- **Link:** Um link compartilhado
- **Message:** Uma mensagem
- **Note:** Uma nota
- **Offer:** Uma oferta publicada por uma página
- **Order:** Um objeto da ordem associada com créditos do Facebook
- **Page:** Uma página do Facebook
- **Photo:** Uma foto individual dentro de um álbum
- **Post:** Um registro individual no “Profile Feed”
- **Question:** Uma pergunta solicitada por um usuário

- **QuestionOption:** Uma opção de resposta a uma pergunta (question)
- **Review:** Uma revisão de uma aplicação
- **Status:** Uma mensagem no mural do usuário
- **Thread:** Um conjunto de mensagens com um mesmo tópico.
- **User:** Um perfil de um usuário
- **Video**: Um vídeo

Todos os objetos no Graph API são ligados uns aos outros por meio de relações chamadas conexões. Por exemplo, no caso: João é amigo de Pedro, a relação é "é amigo", o Facebook oferece acesso às seguintes relações:

- **Friends**: Solicita uma lista de amigos.
- **News feed**: Solicita as ações realizadas por um usuário.
- **Profile feed (Wall)**: Solicita as ações realizadas por amigos de um usuário.
- **Likes**: Solicita os objetos curtidos por um usuário.
- **Movies**: Solicita uma lista de filmes curtidos por um usuário.
- **Music**: Solicita uma lista das músicas curtidas por um usuário.
- **Books**: Solicita uma lista dos livros curtidos por um usuário.
- **Notes**: Solicita as notas de um usuário.
- **Permissions**: Solicita as permissões que tem um usuário sobre um objeto.
- **Photo Tags**: Solicita as fotos nas quais um usuário fui marcado.
- **Photo Albums**: Solicita uma lista dos álbuns de um usuário.
- **Video Tags**: Solicita os vídeos nos quais um usuário foi marcado.
- **Video Uploads**: Solicita os vídeos enviados por um usuário.
- **Events**: Solicita uma lista dos eventos do usuário.
- **Groups**: Solicita uma lista dos grupos nos quais participa o usuário
- **Checkins**: Solicita uma lista dos lugares onde o usuário marcou como um lugar visitado.

- **Objects with Location**: Solicita uma lista dos objetos do usuário que possuem geo-referência (latitude e longitude).

6.2.2 Authentication

Possibilita que outros serviços ou aplicações identifiquem um usuário do Facebook, leiam e escrevam dados por meio da "Graph API" respeitando as permissões atribuídas a esses serviços ou aplicações. O módulo "Authentication" é baseado na especificação OAuth2.0, que oferece o serviço de autenticação e autorização. Portanto, cada requisição que um usuário faça ao sistema terá que estar acompanhada de um *access token*. Por exemplo, a requisição:

https://graph.facebook.com/220439?access_token=23424134wer

Acessa o objeto com identificador "220439", e depende das permissões atribuídas ao *access token* "23424134wer", que é enviado como parâmetro na requisição.

6.3 Orkut

O Orkut é uma rede social do Google, criada em 24 de Janeiro de 2004 com o objetivo de ajudar seus membros a conhecer pessoas e manter relacionamentos. No Brasil, a rede social tem mais de 30 milhões de usuários, mas foi ultrapassada pelo Facebook (MARTINS, 2011). Na Índia também é a segunda rede social mais visitada. O Orkut tem mais de 66 milhões de usuários ativos no mundo (ALEXA INC., 2012).

O Orkut está baseado na versão 8 da plataforma Open Social, que possibilita a comunicação com aplicações externas. O Open Social fornece três módulos para realizar essa comunicação: módulo de Autenticação, módulo serviços web JSON-RPC e o módulo serviços web REST.

O módulo de serviços web JSON-RPC possibilita a comunicação com aplicações externas utilizando o protocolo JSON-RPC, foi o primeiro módulo de comunicação a ser

construído para a especificação Open Social, e atualmente é o módulo que mais oferece recursos aos desenvolvedores de aplicações. Seu funcionamento é acionado por chamadas HTTP de tipo POST em formato JSON. Essa requisição retorna um objeto JSON com o resultado do método chamado. Por exemplo, a solicitação das informações do usuário utilizador:

URL: http://www.orkut.com/social/rpc

Corpo da requisição POST:

```
{
 “method” : “people.get”,
   “id” : “myself”,
   “params” : {
     “userId” : “@me”,
     “groupId” : “@self”
   }
}
```

Resposta da requisição:

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
Date: Wed, 18 May 2011 18:46:31 GMT
Expires: Wed, 18 May 2011 18:46:31 GMT
Cache-Control: private, max-age=0
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
X-XSS-Protection: 1; mode=block
Server: GSE
Transfer-Encoding: chunked

{“id”:”myself”,”data”:{“isViewer”:true,”id”:”12345678”,”thumbnailUrl”:
”http://static1.orkut.com/img/i_nophoto64.gif”,”name”:{“familyName”:”Xyz”,”gi
venName”:”Abc”}}}
```

O outro módulo de serviços web REST tem o mesmo objetivo que o módulo baseado em JSON-RPC, mas utiliza serviços web RESTful para acessar às informações dos usuários. Esse módulo não implementa todas as solicitações de informação definidas pelo módulo JSON-RPC e ainda está em desenvolvimento.

O módulo de Autenticação está baseado na especificação OAuth1.0a e é o encarregado de identificar um usuário da rede social e gerenciar as permissões que o usuário tem sobre os objetos dentro do Orkut. Esse módulo possibilita o acesso às APIs dos módulos de serviços web JSON-RPC e REST.

6.4 Twitter

É uma rede social para *microblogging*, que possibilita aos usuários enviar e receber atualizações pessoais de outros contatos (em textos de até 140 caracteres, conhecidos como "*tweets*"), por meio do *website* do serviço, por SMS e por aplicações específicas de gerenciamento. As atualizações são exibidas no perfil de um usuário e também enviadas a outros usuários seguidores que tenham assinado para recebê-las.

O Twitter possibilita a comunicação com aplicações externas por meio de uma API de serviços web RESTful e gerencia as permissões sobre essa API por meio do OAuth1.0a. A especificação completa da API de serviços web RESTful pode ser encontrada em https://dev.twitter.com/docs/api

6.5 Implantação dos componentes no Arquigrafia

Os cinco componentes de interoperabilidade foram implantados no Arquigrafia de acordo com o diagrama de componentes apresentado na Figura 6.3. É importante notar que a arquitetura do Arquigrafia obedece à estabelecida pelo Groupware Workbench. Para realizar a comunicação entre os componentes de interoperabilidade e as redes sociais foram criados pontos de acesso (aplicações) nos três serviços de redes sociais utilizados para a prova de conceito.

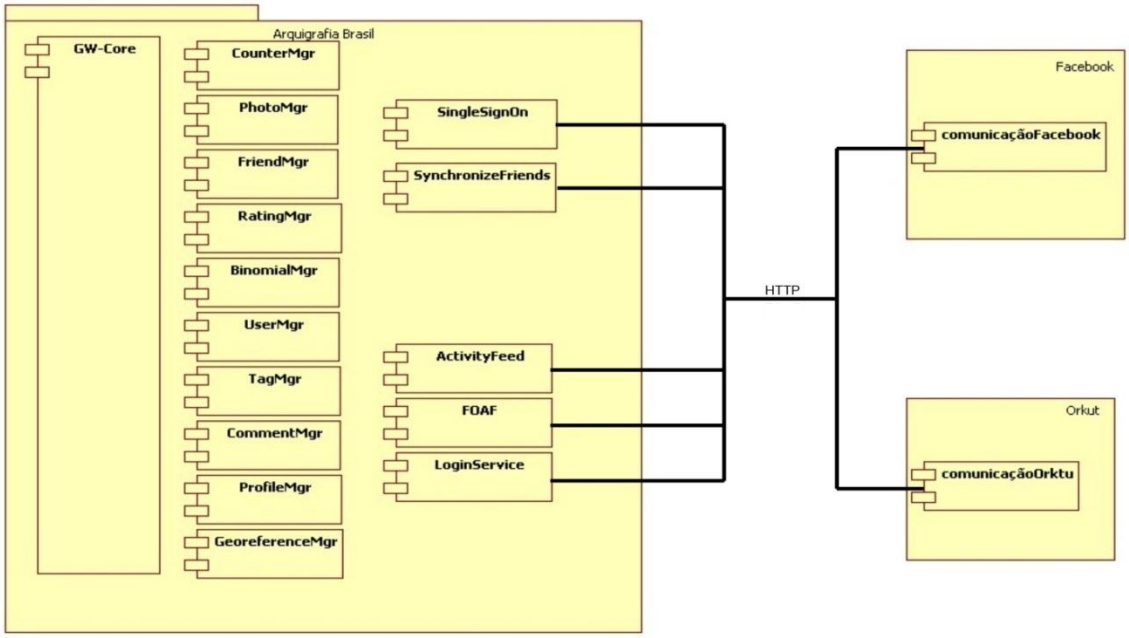

Figura 6.3 Diagrama de componentes Arquigrafia

A seguir é descrito o fluxo de utilização de cada componente de interoperabilidade no Arquigrafia.

6.5.1 Single sing-on

A interface de usuário do componente foi construída com três botões na página de login do Arquigrafia (Figura 6.4). Cada botão possibilita o uso de uma conta de uma rede social. O fluxo de autenticação no caso do Facebook e do Orkut é o especificado pelo padrão OAuth2, e no caso do Twitter é o especificado pelo OAuth1.0a (Seção 4.1.2). A Figura 6.5 mostra o fluxo de autenticação para o Facebook, a Figura 6.6 para o Orkut e a Figura 6.7 para o Twitter.

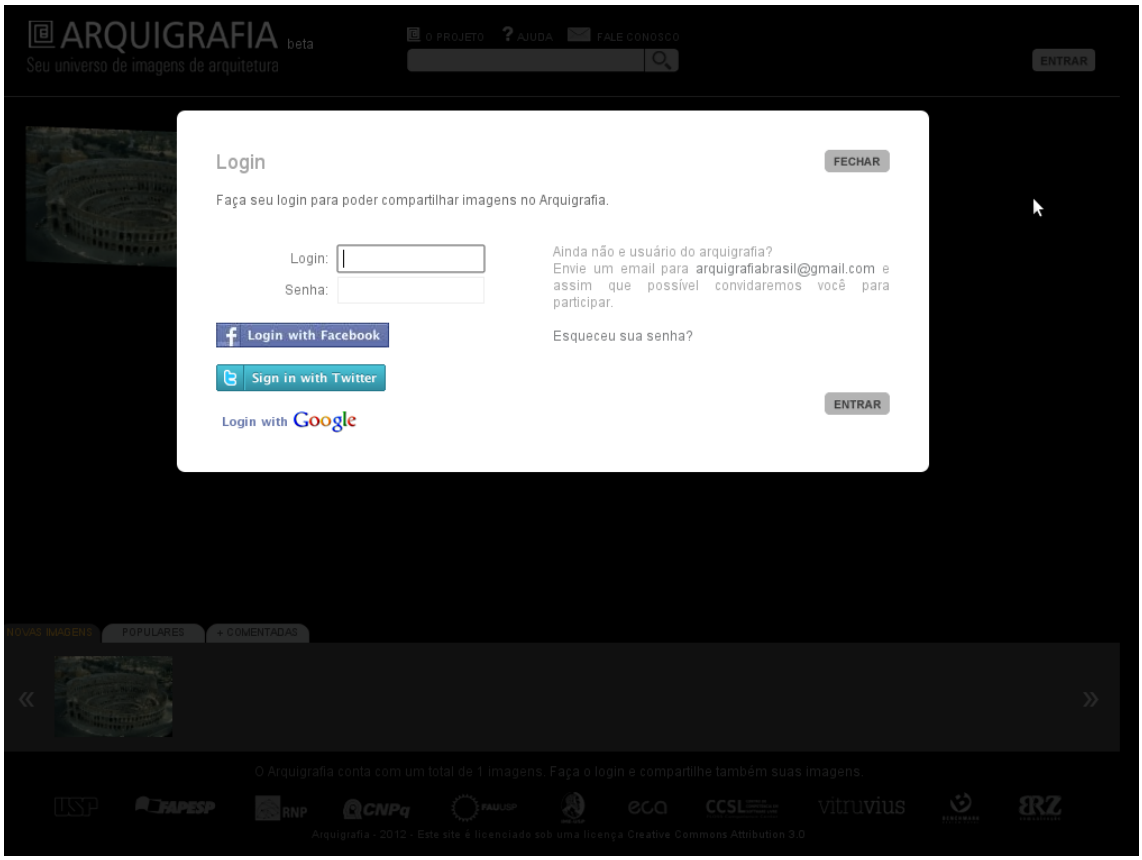

Figura 6.4: Pagina de Login do Arquigrafia

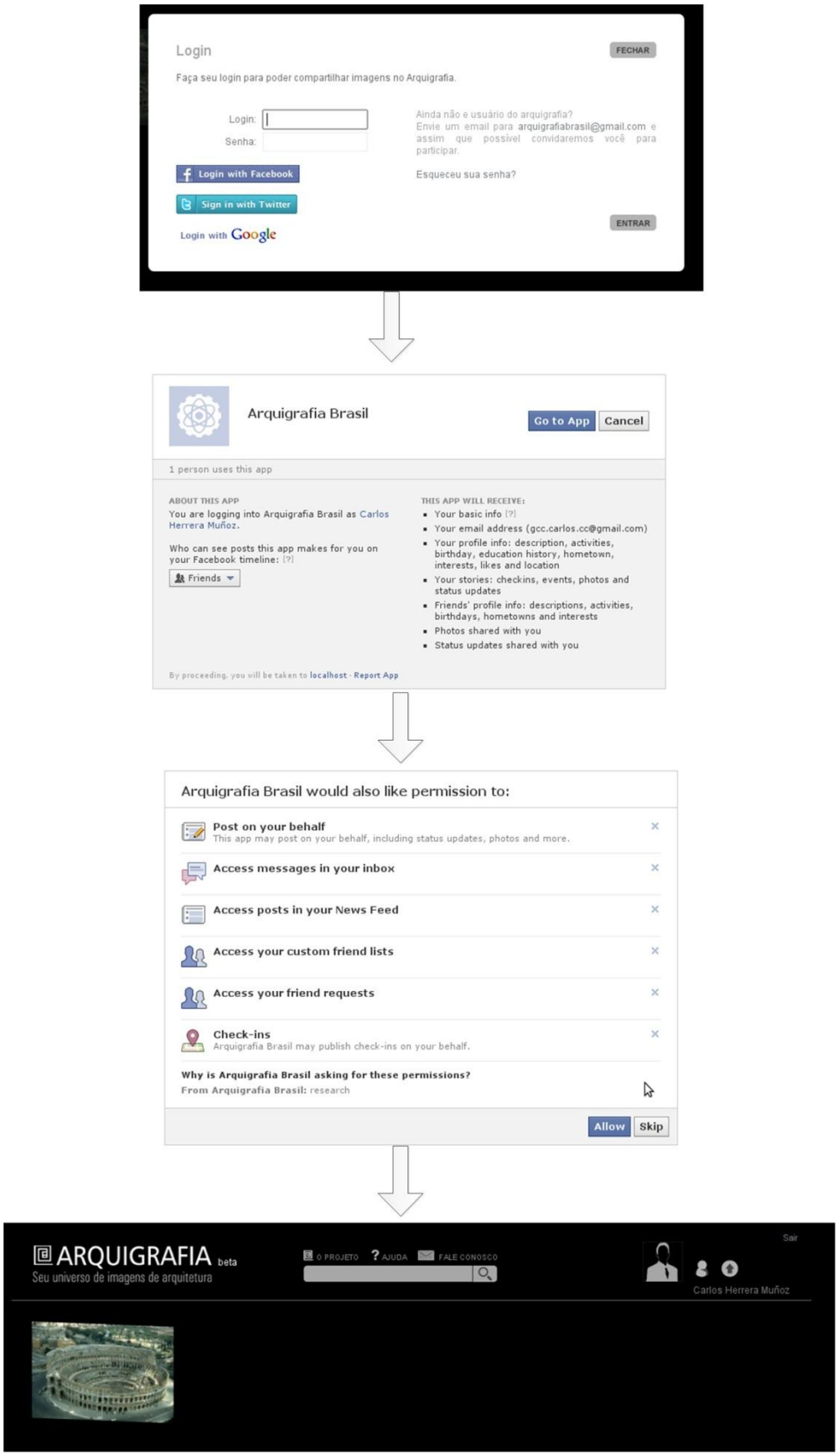

Figura 6.5: Fluxo de autenticação para Facebook

Figura 6.6: Fluxo de autenticação para Orkut

Figura 6.7: Fluxo de autenticação para Twitter

O componente cria uma conta externa e junto com essa conta é armazenado o *access token*, que possibilita posteriormente o acesso aos dados do usuário da rede social

externa. Quando uma conta de tipo externa é excluída do Arquigrafia também é excluído o *access token* associado.

6.5.2 Login Service

A utilização do componente inicia com um usuário do Arquigrafia que deseja aproveitar seus dados em outro sistema na internet. No outro sistema o usuário seleciona a opção "Arquigrafia Login" para ser redirecionado para uma tela de *login* do Arquigrafia e depois para outra tela de concessão de permissões. Se o usuário aprova a concessão de permissões, é direcionado para o primeiro sistema, que cria uma conta com os dados do usuário e possibilita seu acesso. Caso o usuário não consiga iniciar sessão ou não aceite a concessão de permissões, é redirecionado para o primeiro sistema sem possibilidade de iniciar sessão (Figura 6.8).

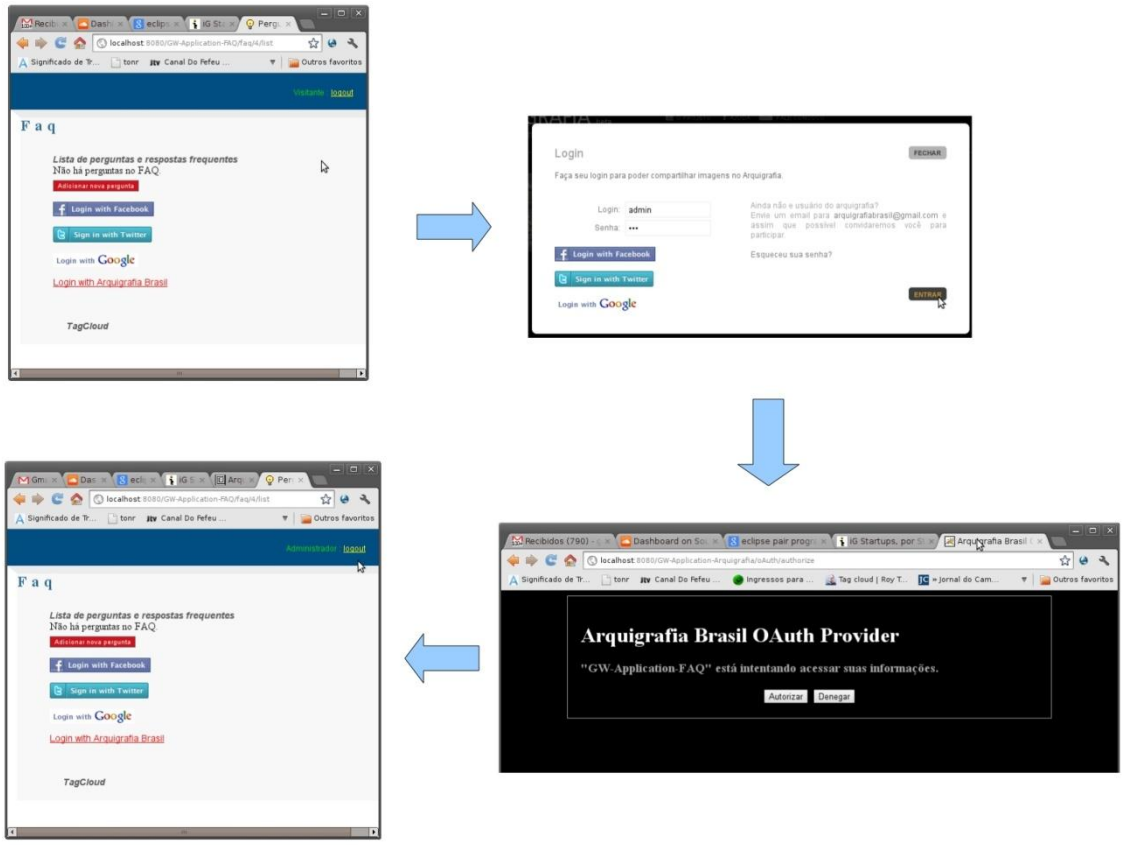

Figura 6.8: Fluxo do componente Login Service

6.5.3 Add Friends

O funcionamento do componente é baseado no envio de convites com enlace para participar da rede social Arquigrafia. Para a utilização desse componente, o usuário deve ter iniciado sessão na rede social Arquigrafia e ter associado pelo menos uma conta de rede social externa (Facebook, Twitter ou Orkut). Logo depois, o usuário acessa "Add Friends", onde acha uma lista com todos os amigos das suas redes sociais externas associadas. O componente na sua interface possibilita ao usuário enviar as mensagens de convite aos seus amigos. A Figura 6.9 e a Figura 6.10 mostram exemplos do funcionamento do componente utilizando a rede social Facebook e a rede social Twitter.

Figura 6.9: Componente "Add Friends" utilizando Facebook

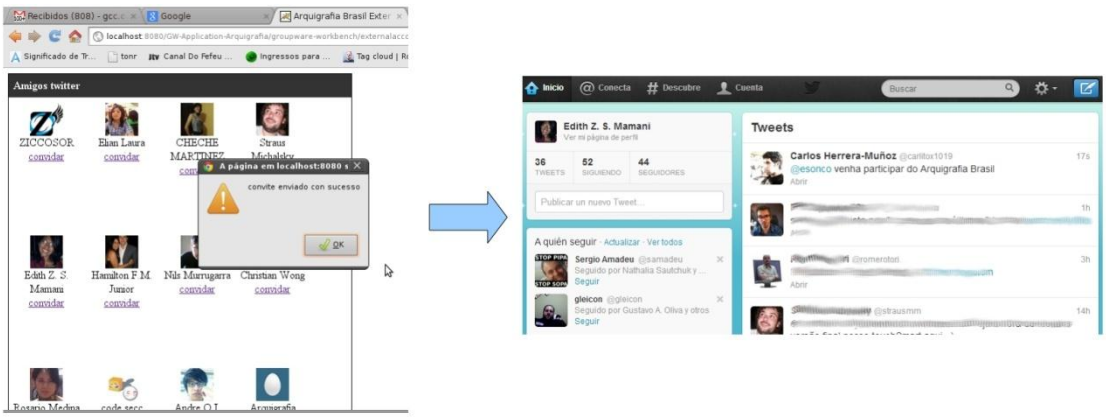

Figura 6.10: Componente "Add Friends" utilizando Twitter

6.5.4 Activity Feed

O funcionamento desse componente depende das ações realizadas em outros componentes chamados de componentes associados. Por exemplo, para a ação de comentar uma foto, o componente associado é o "CommentMgr", e para a ação de enviar uma foto, o componente associado é o "PhotoMgr". Esses componentes associados devem estar declarados na configuração do componente "Activity Feed". Cada usuário do Arquigrafia pode gerenciar as permissões sobre as ações que serão coletadas e exportadas pelo componente "Activity Feed".

A exportação das ações é realizada utilizando o padrão "Activity Stream" em formato JSON. Para a exibição das ações exportadas foi criado um *widget,* que pode ser embarcado em qualquer aplicação web. A Figura 6.11 mostra a utilização do componente "Activity Feed" associado ao componente "CommentMgr" do Arquigrafia. Na parte direita da figura aparece o *widget* que lista todos os comentários exportados.

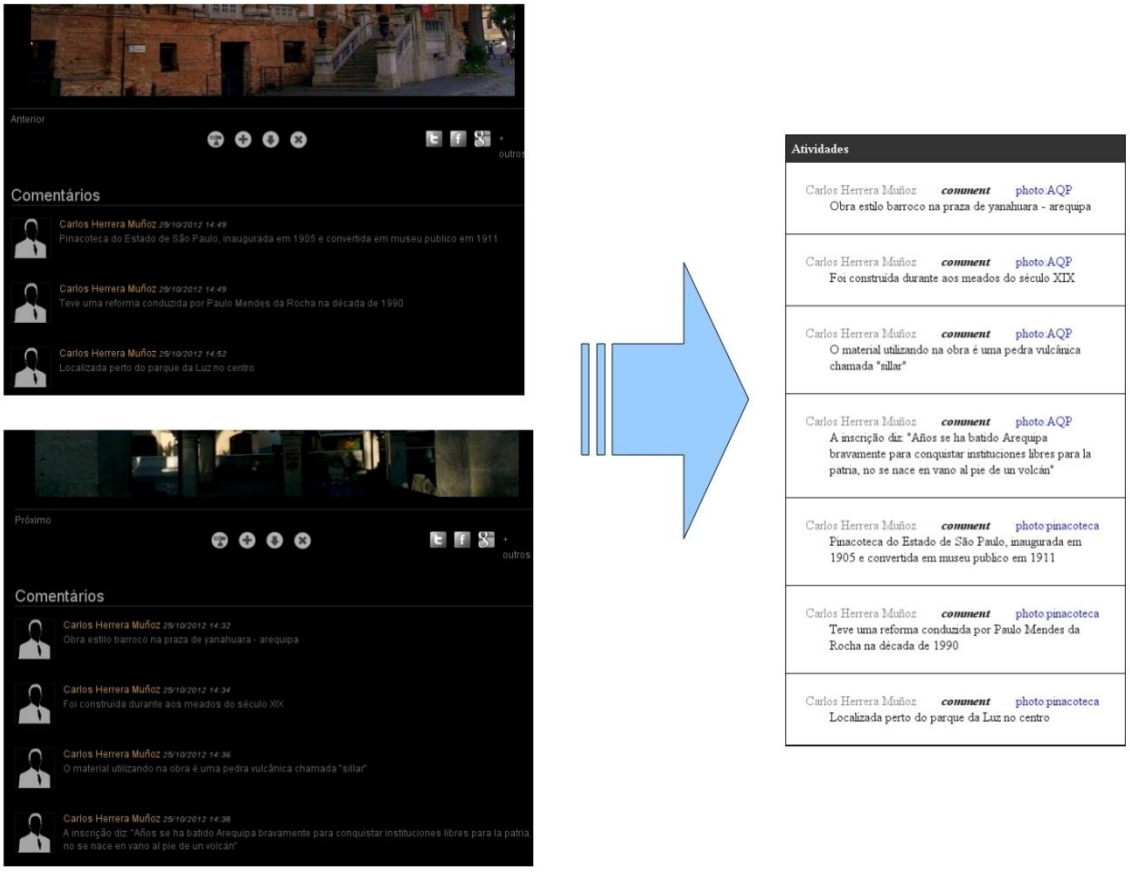

Figura 6.11: Utilização do componente "Activiy Feed"

6.5.5 ProfileExport

O funcionamento do componente "Profile Export" depende de dois componentes, o componente "ProfileMgr" e o "FriendsMgr". "Profile Export" obtém as informações do perfil armazenadas no componente "ProfileMgr" e as informações do grafo social armazenados no componente "FriendsMgr". Com essas duas fontes, é elaborado um documento que descreve o usuário e suas relações com outros usuários dentro da rede social Arquigrafia e esse documento é exportado usando o padrão "Friend of a friend", utilizando para sua representação o formato XML ou JSON segundo convenha. O documento que descreve o usuário é acessível por qualquer aplicação desde que o usuário tenha concedido as permissões necessárias, que são gerenciadas mediante o protocolo OAuth1.0a descrito na seção 4.1.2.1. A Figura 6.12 mostra as informações do usuário

Carlos Herrera Muñoz do Arquigrafia sendo exportadas para uma aplicação *desktop.* A aplicação acessa o documento utilizando o protocolo OAuth1.0a, decodifica-o e mostra as informações do usuário e seus amigos. A Figura 6.13 mostra o documento "Friend of a friend" do usuário utilizado no exemplo.

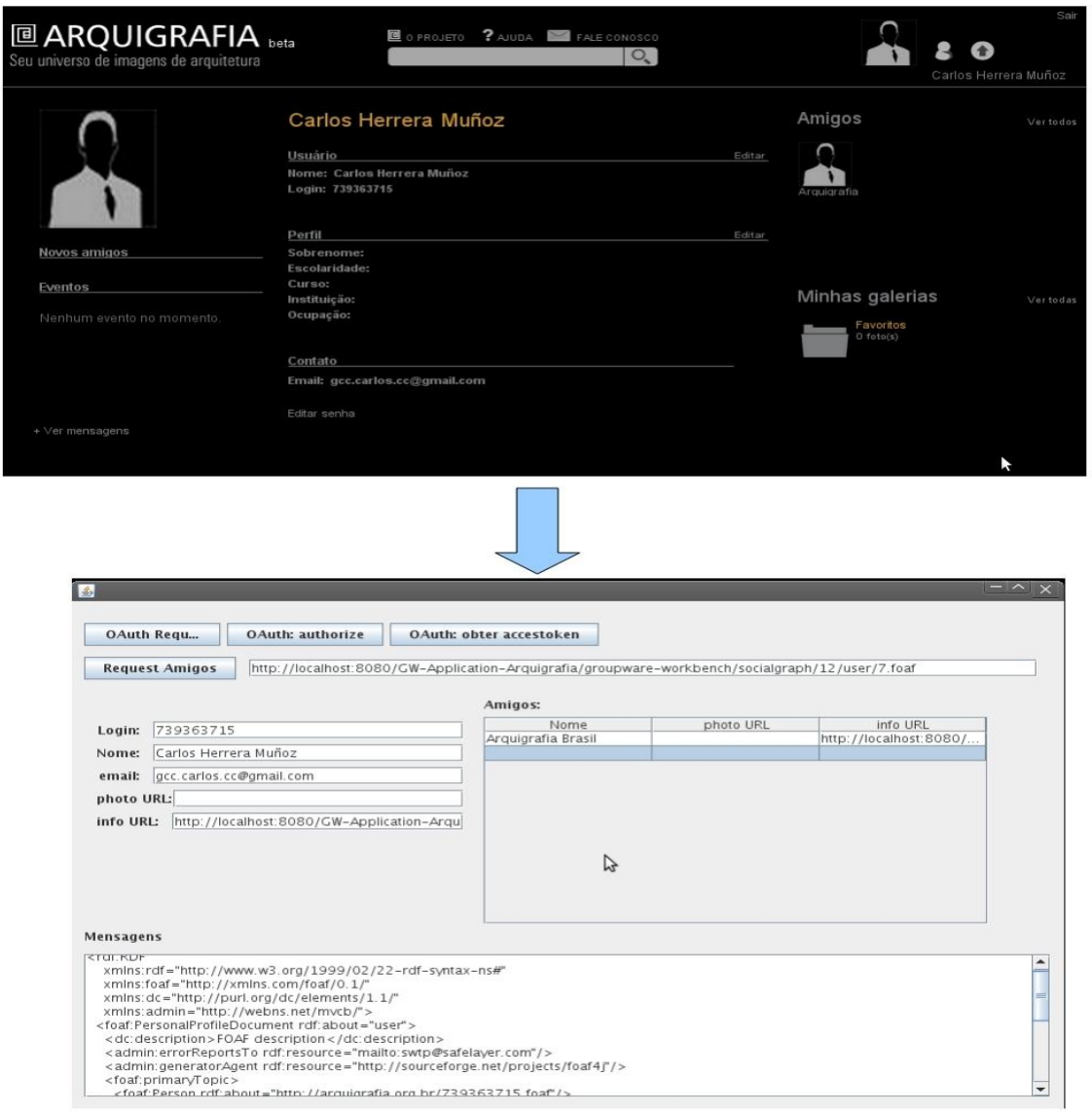

Figura 6.12: Funcionamento do componente "Profile Export"

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
```

```
    xmlns:foaf="http://xmlns.com/foaf/0.1/"
    xmlns:dc="http://purl.org/dc/elements/1.1/"
    xmlns:admin="http://webns.net/mvcb/">
  <foaf:PersonalProfileDocument rdf:about="user">
    <dc:description>FOAF description</dc:description>
    <admin:errorReportsTo rdf:resource="mailto:swtp@safelayer.com"/>
    <admin:generatorAgent rdf:resource="http://sourceforge.net/projects/foaf4j"/>
    <foaf:primaryTopic>
      <foaf:Person rdf:about="http://arquigrafia.org.br/739363715.foaf"/>
    </foaf:primaryTopic>
  </foaf:PersonalProfileDocument>
  <foaf:Person rdf:about="http://arquigrafia.org.br/739363715.foaf">
    <foaf:nick>739363715</foaf:nick>
                        <foaf:name>Carlos Herrera MuÃ±oz</foaf:name>
    <foaf:img rdf:parseType="Resource"></foaf:img>
    <foaf:mbox rdf:resource="gcc.carlos.cc@gmail.com"/>
    <foaf:knows>
        <foaf:Person rdf:about="http://arquigrafia.org.br/885386432.foaf">
          <foaf:nick>885386432</foaf:nick>
          <foaf:name>Arquigrafia</foaf:name>
          <foaf:img rdf:parseType="Resource"></foaf:img>
          <foaf:mbox rdf:resource="arquigrafiabrasil@gmail.com"/>
        </foaf:Person>
    </foaf:knows>
  </foaf:Person>
</rdf:RDF>
```

Figura 6.13: Documento FOAF do usuário *Carlos Herrera Muñoz* do Arquigrafia

6.6 Análise em relação ao Projeto DataPortability

Com os componentes de interoperabilidade, o Arquigrafia ganhou novas funcionalidades diretamente relacionadas com os dados dos usuários e a forma como esses dados são comunicados a outras aplicações na internet. Uma forma de saber o grau de interoperabilidade da rede social é criando um documento de política de portabilidade. Esse documento foi proposto pelo projeto DataPortability, para dizer aos usuários que dados eles podem trazer e tirar da rede social.

O documento de política de portabilidade é composto por dez perguntas, que foram mostradas no Capítulo 3. A seguir é apresentado o documento de política de portabilidade para a rede social Arquigrafia levando em consideração como os componentes desenvolvidos afetam cada uma das respostas.

APIs e formato de dados

As APIs utilizadas estão especificadas na documentação de cada componente, todas elas baseadas em serviços web RESTful. Com respeito ao formato dos dados, são utilizados formatos padrão em cada componentes de interoperabilidade. Para a exportação do perfil é utilizado o formato Friend of a friend (FOAF), e para a exportação das atividades feitas por um usuário o formato Activity Feed. Esses dois formatos estão descritos nas seções 4.2.4 e 4.2.5 respectivamente.

Identidade e autenticação

Os usuários têm duas possibilidades, criar uma nova conta no Arquigrafia ou reusar alguma de suas contas das redes sociais Facebook, Twitter e Orkut. Essa funcionalidade é fornecida pelo componente Single Sign-On.

Trabalhar com conteúdo armazenado em outro lugar

Para os usuários que possuem conta em uma das redes sociais Facebook, Twitter ou Orkut, as informações do perfil dessas redes sociais externas podem ser importadas para a rede social Arquigrafia, podendo ser atualizadas a qualquer momento. Os componentes encarregados de importar as informações do perfil são o Single Sign-On e o ProfileMgr.

Ver atualizações

Arquigrafia não monitora ações acontecidas em outras redes sociais, porque as atividades realizadas em outras redes sociais muitas vezes não estão relacionadas ao objetivo do Arquigrafia, que é coletar fotografias, comentários e opiniões acerca da arquitetura brasileira.

Difundir alterações feitas aqui

As alterações feitas por um usuário no Arquigrafia são difundidas para outras aplicações na internet. O componente responsável pela difusão é o Activity Feed, que utiliza o formato Activity Stream representado em JSON. As atualizações só podem ser divulgadas se o usuário concede as permissões respectivas por meio do componente Login Service, que gerencia as permissões por meio de OAuth.

Acessar a partir de outros produtos

O Arquigrafia possibilita aos seus usuários usarem suas credenciais para acessarem outras aplicações na internet. O componente encarregado de fornecer essa funcionalidade é o "Login Service", que implementa a parte do fornecedor de serviço da especificação OAuth1.0a. Portanto, qualquer aplicação que deseje acessar o Arquigrafia tem que implementar a parte cliente do OAuth1.0a. Além de possibilitar o acesso à rede social, o componente também gerencia as permissões sobre os dados que são acessados.

Cópia de segurança

Por enquanto, o Arquigrafia não fornece uma forma de fazer uma cópia de segurança dos dados fornecidos e atualmente qualquer usuário do Arquigrafia pode acessar os dados de outro usuário, mas não pode modificá-los. Com respeito à administração de direitos digitais, a rede social oferece opção ao usuário de escolher uma licença da "Creative Commons BY" [50] cada vez que efetua um envio.

Dados públicos

Um usuário do Arquigrafia, por meio dos componentes de interoperabilidade, pode acessar remotamente os dados fornecidos por ele ou por outro usuário da rede social, desde que possua as permissões respectivas para acessar os recursos. Uma pessoa que não

[50] A licença Creative Commons BY permite copiar, exibir, distribuir e executar uma obra.

é usuária do Arquigrafia também pode acessar dados que são disponibilizados como públicos.

Fechar uma conta

Por enquanto, o Arquigrafia não possibilita apagar todos os dados fornecidos à rede social no momento de fechar uma conta. Quando um usuário solicita fechar sua conta, esta não é apagada, mas desativada, dado que outros usuários podem ter criado conteúdos baseados nos dados de outras pessoas como, por exemplo, comentários ou avaliações nas fotos.

Onde são armazenados os conteúdos

Por enquanto, o Arquigrafia não divulga onde os dados são armazenados, e também não possibilita ao usuário escolher onde seus dados devem ser armazenados. Atualmente as informações da rede social são armazenadas no serviço de computação em nuvem do Centro de Computação Eletrônica da Universidade de São Paulo, podendo ser migradas para outro serviço a qualquer momento, sem aviso prévio.

6.7 Análise em relação aos requisitos de interoperabilidade para redes sociais

Depois de ter feito o documento de política de interoperabilidade para a rede social Arquigrafia, vamos analisar cada um dos requisitos de interoperabilidade levantados no Capítulo 3.

Possibilidade de adicionar amigos entre redes sociais

O requisito indica que a rede social Arquigrafia pode adicionar qualquer usuário de outra rede social. Esse requisito é cumprido por meio do componente AddFriends, que envia solicitações de amizade aos usuários das redes sociais interoperáveis com Arquigrafia

(Facebook, Twitter ou Orkut). A solicitação de amizade possui um vínculo para a página do Arquigrafia, onde o novo usuário pode iniciar uma sessão com a conta da rede social da qual veio.

Possibilidade de acessar o relatório de atividades dos amigos de outras redes sociais

O Arquigrafia possibilita acessar o relatório de atividades de outras redes sociais, desde que o usuário que solicita o relatório de atividades possua as permissões necessárias. O componente encarregado de exportar o relatório de atividades é o Activity Stream.

Possibilidade de acessar a lista de amigos dos amigos de outras redes sociais

O Arquigrafia possibilita navegar o perfil e grafo social de outra rede social, ou qualquer outra aplicação na internet, desde que a aplicação externa possua as permissões necessárias. Essas permissões são gerenciadas pelo OAuth e o componente Profile Export é encarregado de exportar o perfil e o grafo social.

Possibilidade de enviar mensagens aos amigos sem se importar a que rede social pertença

Esse requisito foi cumprido utilizando o sistema de correio eletrônico já existente no Arquigrafia. Todos os usuários do Arquigrafia possuem um login único o qual dá nome ao endereço do sistema de mensagens, por exemplo, qualquer correio eletrônico enviado a carlos@arquigrafia.org.br poderá ser lido na caixa de mensagens do usuário. Essa estratégia foi baseada no sistema de mensagens do Facebook.

7 Conclusão

O principal objetivo desta dissertação foi implementar por meio de componentes de software técnicas para possibilitar a interoperabilidade de redes sociais na web. Foram identificadas quatro técnicas recorrentes na interoperabilidade de sistemas na internet:

- Utilização de um sistema de autenticação padrão compartilhado entre serviços de redes sociais na internet.
- Utilização de um formato padrão para representar a informação dos usuários.
- Utilização de serviços Web RESTful e uso de conteúdo de fontes externas.
- Definição de políticas de gerenciamento dos dados do usuário por parte do serviço de rede social.

Após identificar as técnicas, foi feito um estudo das possíveis tecnologias que poderiam implementar as técnicas. O estudo ajudou na escolha das tecnologias e deu como resultado a proposta de cinco componentes:

- Single Sign-On
- Login Service
- Profile Export
- Add Friends
- Activity Stream

O modelo de componentes implementado foi o definido pelo Groupware Workbench, que tem como características, o reúso e a facilitada implantação. Os componentes foram implantados na rede social Arquigrafia, que é uma rede social que tem por objetivo coletar fotografias, avaliações e comentários acerca da arquitetura brasileira. O Arquigrafia não possuía interoperabilidade com outra rede social ou serviço na internet.

Com os componentes, o Arquigrafia ganhou novas funcionalidades com respeito à sua interoperabilidade.

Foi criado o documento de política de portabilidade proposto pelo projeto DataPortability, que é baseado em dez questões que descrevem que dados os usuários de um serviço podem compartilhar com outro serviço na internet. Além da análise feita com o documento de política de portabilidade foi feito uma análise com respeito aos requisitos apresentados por Tiwari (2008).

Em relação ao documento de política portabilidade, algumas questões não têm relação com os objetivos ou políticas da rede social Arquigrafia. Por exemplo, a última questão do documento de portabilidade visa a informar ao usuário a zona geográfica em que estão sendo armazenados seus dados. A questão não tem relação com Arquigrafia porque os dados da rede social não são afetados pela localização geográfica, como seriam, por exemplo, os dados de uma aplicação do governo.

Em relação aos requisitos de interoperabilidade para redes sociais, os quatro requisitos foram completados, dessa forma Arquigrafia pode receber mensagens de outras redes sociais, possibilita navegar o perfil e grafo social de um usuário, exportar as atividades acontecidas na rede social e adicionar amigos de outras redes sociais.

Com respeito à comunicação entre redes sociais, um modelo recorrente é o baseado em *social aggregators,* apresentado no terceiro capítulo. Nesse modelo, uma rede social atua como agregador de outras, importando algumas de suas informações. Por exemplo, temos o caso das redes sociais Facebook e Twitter, nas quais o Facebook atua como agregador do Twitter e todas as mensagens recebidas no Twitter também aparecem no Facebook, caso o usuário deseje.

A tecnologia chave para conseguir o comportamento anteriormente descrito é o OAuth, visto que além autenticar um usuário, possibilita o compartilhamento de recursos na internet respeitando políticas de privacidade. Essa tecnologia está bastante difundida.

Atualmente as empresas que oferecem os serviços de redes sociais, estão enfocando seus esforços em produzir APIs de comunicação de fácil uso (WAGNET, 2011). Sem a utilização de um formato de dados padrão para a comunicação dos dados, o número de APIs diferentes cresce, diminuindo o grau de interoperabilidade semântica entre as redes sociais, de modo que um dado representado numa rede social não necessariamente pode significar o mesmo em outra. Por isso, para diminuir esse problema é importante o uso de formatos de dados padrões ou ontologias.

Atualmente a interoperabilidade entre sistema na internet encontra-se fragmentada, tendo os sistemas seus próprios mecanismos para interoperar com outros, mas há uma tendência recente, iniciada nos sistemas de redes sociais a utilizar tecnologias como OAuth e Activity Strems. Com os componentes desenvolvidos neste trabalho espera-se que a implantação de recursos de interoperabilidade em outros sistemas seja facilitada.

Um último ponto importante de mencionar é que uma mesma pessoa que possui contas em varias redes sociais pode desempenhar papeis diferentes em cada uma delas. Por exemplo, um professor universitário possui uma conta na rede social da sua universidade na qual discute temas acadêmicos e o mesmo professor também possui uma conta numa rede social como Facebook, na qual se comunica com seus amigos. O problema com isso é ter cuidado na utilização das informações de uma rede social em outra, dado que não é sempre que o usuário gostaria de reusar suas informações em diferentes redes.

7.1 Contribuições

Como uma das principais contribuições desta dissertação é identificação das técnicas de interoperabilidade nas redes sociais. Essa identificação foi baseada no estudo de diversos sistemas de redes sociais na internet e também na revisão de publicações na área. Outra contribuição desse trabalho é o estudo das tecnologias para interoperar redes sociais. Foram avaliadas diversas tecnologias orientadas a interoperar sistemas na internet, das quais foram escolhidas as que melhor se adaptavam aos sistemas de redes sociais, levando em consideração que fossem baseadas em padrões abertos.

Outra das principais contribuições é um conjunto de componentes de interoperabilidade para redes sociais, baseado na plataforma Groupware Workbench (2010). Esse conjunto de componentes foi implantado na rede social Arquigrafia, que ganhou interoperabilidade com as redes sociais Facebook, Twitter e Orkut. O conjunto de componentes é distribuído sob a licença LGPL[51] e encontra-se no endereço http://code.google.com/p/groupware-workbench/.

7.2 Trabalhos relacionados

Um dos trabalhos interessantes de experimentar no futuro, é investigar a tecnologia WebID[52], que é um sistema de autenticação baseado em chaves publicas e privadas, sem a necessidade de criar senhas. Esse sistema ainda encontra-se em fase de desenvolvimento e não chegou às redes sociais.

Outro tema que fica aberto é como identificar e associar contas de diferentes redes sociais de uma mesma pessoa. Isso poderia ser feito analisando os dados básicos da pessoa como seu nome, e-mail, ou também buscando similaridades nos grafos sociais.

---

[51] http://www.gnu.org/licenses/lgpl.html
[52] http://www.w3.org/wiki/WebID

Outro trabalho interessante de explorar é com respeito às politicas de gerenciamento de dados por parte dos sistemas de redes sociais, tentando achar uma forma padrão de como gerenciar a privacidade e compartilhamento dos dados nas redes sociais.

Por último, um trabalho futuro é pesquisar como informações sensíveis ao contexto são trocadas entre redes sociais. Informações que sejam validas em uma rede social, podem não fazer sentido em outra rede social ou comprometer a privacidade do usuário.

Referências Bibliográficas

AGIMO. Interoperability Technical Framework for the Australian Government, Junho 2003. Disponivel em: <http://www.agimo.gov.au/publications/2005/04/agtifv2>. Acesso em: Maio 2012.

ALEXA INC. Statistics Summary for orkut.com, Junho 2012. Disponivel em: <http://www.alexa.com/siteinfo/orkut.com>. Acesso em: 08 Agosto 2012.

ANDERSEEN, D.; DAWES, S. **Government information management**. [S.l.]. 1991.

ANDERSON ANALYTICS. Anderson Analytics. **Overlap Among Major Social Network Services**, 09 Julho 2009. Disponivel em: <http://www.tomhcanderson.com/2009/07/09/overlap-among-major-social-network-services/>. Acesso em: 10 Agosto 2011.

ARMS, W. **Thoughts about Interoperability in the NSDL**. Cornell University. 2000.

BOJARS, U. et al. **Social Network and Data Portability using**. Social Aspects of the Web. Austria: [s.n.]. 2008.

BOMFIM, M. N. D. C.; SAMPAIO, F. F. **A Web 2.0, suas tecnologias e aplicações educacionais**. Universidade Feredal de Rio de Janeiro. 2008.

CARDOZO, L. M. **Internet, Propagando pessoal:** Redes sociais na. XXXI Congresso Brasileiro de Ciências da Comunicação. [S.l.]: [s.n.]. 2008.

COMPETE. Compete Pulse. **Connecting the Social Graph:** Member Overlap at OpenSocial and Facebook, 12 Novembro 2007. Disponivel em: <http://blog.compete.com/2007/11/12/connecting-the-social-graph-member-overlap-at-opensocial-and-facebook/>. Acesso em: 08 ago. 2011.

CURRY, R.; KIDDLE, C.; SIMMONDS, R. **Social Networking and Scientific Gateways**. 5th Grid Computing Environments Workshop. [S.l.]: [s.n.]. 2009.

DATA PORTABILITY PROJECT. Portability Policy Questions, 2010. Disponivel em: <http://portabilitypolicy.puzzlingevidence.net/questions.html/>.

DAVID, P.; GREENSTEIN, S. **The economics of compatibility standards: an introduction to recent research**. [S.l.]. 1990.

DAVID, P.; STEINMUELLER, W. Economics of compatibility standards and competition in telecommunication networks. In: _____ **Information Economics and Policy**. [S.l.]: [s.n.], 1994.

DEITEL, J. **Ajax, Rich Internet Applications e desenvolvimento**. São Paulo: Pearson Prentice Hall, 2008.

DUARTE, F.; QUANDT, C.; SOUZA, Q. **O Tempo das Redes**. [S.l.]: [s.n.], 2008.

EICTA. EICTA White Paper on Interoperability and Standardisation, 2006. Disponivel em: <http://www.eicta.org/fileadmin/user_upload/document/document1166544474.pdf>.

ELGG. Elgg the open source social networking engine, 2009. Disponivel em: <http://www.elgg.org/>. Acesso em: jun. 2011.

ELLIS, C. A.; GIBBS, S. J.; REIN, G. L. Groupware - Some Issues and Experiences. **Communications of the ACM, v. 34**, p. 38-58, 1991.

FOAF PROJECT. Introducing FOAF, 2000. Disponivel em: <http://www.foaf-project.org/original-intro>. Acesso em: jun. 2011.

FUKS, H. et al. The 3C Collaboration Model. In: KOCK, N. **The Encyclopedia of E-Collaboration**. [S.l.]: [s.n.], 2007. p. 637-644.

GARUD, R.; JAIN, S.; KUMARASWAMY, A. Institutional Entrepreneurship in the Sponsoring of Common Technological Standards: The Case of Sun Microsystems and Java. **Academy of Management Journal**, 2000.

GEROSA, M. A. et al. **Component-Based Groupware Development Based on the 3C Collaboration Model**. Simpósio Brasileiro de Engenharia de Software - SBES. Florianópolis: [s.n.]. 2006.

GOLBECK, J.; ROTHSTEIN, M. **Linking social networks on the Web with FOAF:** a semantic web case study. Proceedings of the 23rd national conference on Artificial intelligence. Chicago, Illinois: [s.n.]. 2008.

GRAHAM, I. ET AL. The Dynamics of EDI Standard Development. In: _____ **Technology Analysis & Strategic Management**. [S.l.]: [s.n.], 1995.

GREENBERG, S. Toolkits and interface creativity. **Journal of Multimedia Tools and Applications**, p. 139-159, 2007.

GROUPWARE WORKBENCH. Groupware Workbench. **Groupware Workbench**, 2010. Disponivel em: <http://www.groupwareworkbench.org.br>. Acesso em: jun. 2011.

HANSETH, O.; MONTEIRO, E. Standards and standardization processes. In: _____ **Understanding information infrastructure**. [S.l.]: [s.n.], 1998.

HOLMES, J. Enterprise mashups. **The Arquitecture Journal**, 2010. Disponivel em: <http://msdn.microsoft.com/en-us/architecture/bb906060.aspx>. Acesso em: Janeiro 2011.

IBM. IBM Connections, 2008. Disponivel em: <http://www-01.ibm.com/software/lotus/products/connections/>. Acesso em: jun. 2011.

IBOPE. Redes sociais no mundo, que poder é esse?, Junho 2010. Disponivel em: <http://www.ibope.com.br/carta/redes_sociais_saiba_mais.htm>. Acesso em: 18 Fevereiro 2011.

IDABC. European Interoperability Framework for pan-European eGovernment Services. **Oficina de Publicaciones Oficiales de las Comunidades Europeas**, 2004. Disponivel em: <http://europa.eu.int/idabc/en/document/3761>. Acesso em: Maio 2012.

IDABC. **Interoperable Delivery of European eGovernment Services to public**, 2007. Disponivel em: <http://ec.europa.eu/idabc/en/chapter/3>. Acesso em: Maio 2012.

IEEE. **IEEE Standard Computer Dictionary**. [S.l.]: [s.n.], 1990.

ISO. Information technology -- Interoperability with assistive technology (AT). **Interoperability with assistive technology**, 2011. Disponivel em: <http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=53770>. Acesso em: 09 jun. 2011.

JCP. JSR 168: Portlet Specification, 2009. Disponivel em: <http://www.jcp.org/ja/jsr/detail?id=168>. Acesso em: jun. 2011.

JSON. Introducing JSON. **JSON**, Decembro 1999. Disponivel em: <http://www.json.org/>. Acesso em: Junho 2011.

LANDSBERGEN; WOLKEN, J. **Realizing the promise: government information systems and the fourth generation of information technology**. [S.l.]. 2001.

MARTINS, R. Facebook ultrapassa Orkut em usuários no BR, diz revista; Ibope não confirma, 4 Setembro 2011. Disponivel em: <http://blogs.estadao.com.br/rodrigo-martins/2011/09/04/facebook-ultrapassa-orkut-em-usuarios-e-acaba-com-reinado-de-7-anos-no-br/>. Acesso em: 2012 Agosto 8.

MENDES, M. Web Services, sim… WSDL e SOAP … nem sempre, 7 Novembro 2010. Disponivel em: <http://blog.marcomendes.com/category/tecnologias-java/>. Acesso em: 2011 Junho 10.

METCALF. MetCalf's law, 1993. Disponivel em: <http://www.ec.njit.edu/~robertso/infosci/metcalf.html>. Acesso em: fev. 2011.

MICHALSKY, S. **Componentes de Software no desenvolvimento de aplicações colaborativas para Web: Evolução da plataforma Groupware Workbench**. IME-USP. São Paulo. 2012.

MIKA, P. **Social Networks and Semantic Web**. [S.l.]: Springer, 2007.

MINISTÉRIO DO PLANEJAMENTO, ORÇAMENTO E GESTÃO. **Guia de Interoperabilidade, cartilha técnica**. [S.l.]. 2010.

MINISTÉRIO DO PLANEJAMENTO, ORÇAMENTO E GESTÃO. **Guia de Interoperabilidade: Cartilha técnica**. [S.l.]. 2011.

MOSTARDA, M. et al. **Towards an OpenID-based solution to the Social Network Interoperability**. W3C Workshop on the Future of Social Networking. [S.l.]: [s.n.]. 2009.

OAUTH. OAuth Community Site, 2007. Disponivel em: <http://oauth.net/>. Acesso em: jun. 2011.

OPENID. OpenID Foundation website, 2006. Disponivel em: <http://openid.net/>. Acesso em: jun. 2011.

O'REILLY, T. What is Web 2.0: Design patterns and business models for the next generation of software, 2005. Disponivel em: <http://oreilly.com/web2/archive/what-is-web-20.html>. Acesso em: set. 2010.

PROTALINSKI, E. Facebook has over 845 million users, 01 Fevereiro 2012. Disponivel em: <http://www.zdnet.com/blog/facebook/facebook-has-over-845-million-users/8332>. Acesso em: 15 Julho 2012.

RECUERO, R. **Redes Sociais na Internet**. [S.l.]: [s.n.], 2009.

RODRIGUEZ, A. RESTful Web services: The basics. **IBM Developer Works**, 6 Novembro 2008. Disponivel em: <http://www.ibm.com/developerworks/webservices/library/ws-restful/>. Acesso em: Junho 2011.

ROZESTRATEN, A. S. et al. **Rede Social Arquigrafia-Brasil Design de um ambiente online baseado em transdisciplinaridade e colaboração**. SBSC. [S.l.]: [s.n.]. 2010.

SCHWAGEREIT, F.; SCHERP, A.; STAAB, S. **Representing Distributed Groups with dgFOAF**. 7th Extended Semantic Web Conference. [S.l.]: [s.n.]. 2010.

STREAMS, A. Activity Streams, 2011. Disponivel em: <http://activitystrea.ms/>.

TAPIADOR DEL DUJO, A. et al. **Identidad Extendida en Redes Sociales**. VII Jornadas de Ingeniería Telemática JITEL. Alcalá de Henares, España: [s.n.]. 2008.

TASSEY, G. **Standardization in technology-based markets**. [S.l.]. 2000.

THEODOROU. Technical Interoperability: Report about Important Issues, Requirements and State-of-the-Art, 2007. Disponivel em: <http://www.qualipso.org/media/A3/D3.1.1B.pdf>. Acesso em: Maio 2012.

TIWARI, A. OpenSocial Should Enable Social Network Interoperability, Janeiro 2008. Disponivel em: <http://abhishek.tiwari.com/2007/11/11/opensocial-should-enable-social-network-interoperability/>. Acesso em: abr. 2011.

VERSIT CONSORTIUM. vCard The Electronic Business Card, set. 1996. Disponivel em: <http://www.imc.org/pdi/vcard-21.txt>. Acesso em: maio 2011.

W3C. RSS 2.0 specification, 2002. Disponivel em: <http://feed2.w3.org/docs/rss2.html>. Acesso em: jun. 2011.

W3C. RDF, fev. 2004. Disponivel em: <http://www.w3.org/RDF/>. Acesso em: jun. 2011.

W3C. Web Services Architecture, 11 Fevereiro 2004. Disponivel em: <http://www.w3.org/TR/ws-arch/>. Acesso em: 2011 Junho 1.

WAGNET, J. The Increasing Importance of APIs in Web Development, 2011. Disponivel em: <http://net.tutsplus.com/articles/news/the-increasing-importance-of-apis-in-web-development/>. Acesso em: Janeiro 2012.

WILLIAMS, R. Universal Solutions or Local Contingencies: Tensions and Contradictions in the Mutual Shaping of Technology and Work Organization. In: _____ **Innovation Organizational Change and Technology**. [S.l.]: [s.n.], 1997.

WORDPRESS. WordPress, Writing a Plugin, 2011. Disponivel em: <http://codex.wordpress.org/Writing_a_Plugin>. Acesso em: jun. 2011.

Apêndice

Outros sistemas de componentes relacionados ao Groupware Workbench

Além do Groupware Workbench, há outros modelos de componentes baseados em interfaces Web, que possuem características similares. Nesta seção são analisados quatro modelos de componentes, e comparados com o Groupware Workbench nos seguintes aspectos:

- Composição de componentes
- Habilitar e desabilitar o componente em tempo de execução
- Nível de conhecimento e número de tecnologias para implantar um componente
- Suporte para dispositivos móveis
- Licença

As plataformas analisadas são IBM Connections, Elgg, Porlets e WordPress. Elas foram escolhidas por sua popularidade e impacto na Web.

1. **IBM Connections**

É uma plataforma para desenvolvimento de redes sociais na Web 2.0 orientadas ao mundo empresarial. É desenvolvida pela Lotus Software, e o projeto começou no ano de 2007, sendo a primeira versão lançada no dia 27 de junho de 2008; atualmente, está na versão 3.0.1, lançada em abril de 2011 (IBM, 2008). A plataforma é composta de dez componentes:

- Homepage - é um portal no qual podem se incluir informações de diferentes fontes.
- Profiles - é utilizado para achar pessoas na organização.
- Comunidades - é um espaço colaborativo para pessoas com interesse em trabalhar juntas.
- Blogs - é um serviço de blogging.

- Bookmarks - é um serviço social de marcadores.
- Activities - é um gerenciador de tarefas para grupos que trabalham juntos num projeto.
- Files - é um serviço para armazenar e compartilhar arquivos eletrônicos.
- Wikis - é um sistema baseado em wiki para publicar e editar conteúdo.
- Fórum - é um sistema em que as pessoas podem discutir sobre interesses comuns.
- Search - é um sistema para buscas.

Adicionalmente a plataforma tem suporte para dispositivos móveis Blackberry, Android e IPhone. Usa padrões abertos como ATOM e RSS para integrar com outras aplicações, e fornece uma API baseada em REST para desenvolvedores. Também podem ser adicionados widgeds externos como os Google Gadgets. A arquitetura de componentes está baseada em SOA, de modo que os componentes podem ser hospedados independentemente uns dos outros.

O IBM Connections possui muitas características similares ao Groupware Workbench, como a composição de componentes, habilitar e desabilitar componentes em tempo de execução e suporte para dispositivos móveis. Uma caraterística importante é o nível baixo de conhecimento técnico requerido para montar um componente, o qual é feito por meio do recurso de arrastar e soltar.

Algumas diferenças com o Groupware Workbench são o tipo de licença e o escopo: IBM Connections tem uma licença privativa, enquanto o Groupware Workbench possui uma licença livre, e em referência ao escopo, o IBM Connections foca-se somente em redes sociais no âmbito empresarial, já o Groupware Workbench concentra-se em sistemas colaborativos em geral.

2. Plataforma Elgg

O Elgg (ELGG, 2009) é uma plataforma para desenvolvimento de redes sociais escrita em PHP, está baseado no padrão MVC e define um sistema de ações – eventos que

tornam possível estender a plataforma por meio de *plug-ins*. Os *plug-ins* possibilitam modificar ou estender as funcionalidades no Elgg, podendo prover desde um editor HTML até um sistema completo de fórum. Um plug-in do Elgg consiste em um conjunto de arquivos PHP, que seguem uma estrutura de nomes de arquivos e diretórios padronizados. A comunicação entre a plataforma e o *plug-in* é por meio de eventos; a plataforma fornece um conjunto de eventos padrão em que as ações dos *plug-ins* podem ser registradas.

Como primeira diferença dessa plataforma com o Groupware Workbench é que Elgg está focado inteiramente a redes sociais, enquanto o Groupware Workbench foca-se em sistemas colaborativos em geral.

Outra diferença é que no Elgg não se possibilita composição de componentes, portanto, não é possível ter um *plug-in* dentro do outro. Uma última diferença é que a plataforma Elgg não fornece suporte para habilitar e desabilitar componentes em tempo de execução.

Elgg possui caraterísticas similares ao Groupware Workbench como o tipo de licença (ambas possuem uma licença livre), o nível de conhecimento técnico requerido para montar uma aplicação e o suporte para dispositivos móveis.

3. Portlets

Um Portlet (JCP, 2009) é um componente visual independente que é utilizado para disponibilizar informações dentro de uma página Web. Um Portlet pode ser utilizado em qualquer portal, promovendo-se assim o reúso. Esse fator fez que esse modelo ganhasse grande popularidade junto às equipes de desenvolvimento de portais para Web. O Portlet possui uma estrutura similar a um servlet mais a adição de um arquivo de implantação chamado porlet.xml. A especificação do modelo de Portlets é a aberta.

No modelo de Portlets não é possível montar um novo portlet a partir de outros, como no Groupware Workbench. Também não é possível a navegação entre Portlets, pois

estes se comunicam exclusivamente com o contêiner, ao contrário do modelo de componentes do Groupware Workbench, em que é possibilitada a navegação e comunicação entre componentes. A diferença com o Groupware Workbench é que os Portlets não oferecem suporte nativo para dispositivos móveis. Como caraterísticas similares temos que nos dois está permitido habilitar e desabilitar componentes sem a necessidade de reiniciar a aplicação inteira, e que não precisam de alto nível técnico para montar os componentes.

4. WordPress

É um sistema de gestão de conteúdo focado na criação de blogs, desenvolvido em PHP e Mysql, com a licença GPL, Tem um grande sucesso devido ao número de *plug-ins* disponíveis na internet e sua facilidade de uso. O modelo de plug-ins é bastante simples, a plataforma oferece pontos de extensão chamados *hooks,* os quais são de dois tipos: *actions* e *filters.* Os *actions* são executados quando se produz um evento no sistema, e os *filters* são executados quando vai ser armazenado ou apresentado algum tipo de o texto. O código do plug-in é armazenado numa pasta especial para depois serem habilitados no painel de controle (WORDPRESS, 2011).

WordPress compartilha caraterísticas similares com o Groupware Workbench. A primeira referente a licenças onde as duas plataformas possuem licenças abertas, outra caraterística é que as duas requerem um nível técnico baixo para montar os plug-ins. As duas também podem habilitar e desabilitar componentes em tempo de execução. Como diferenças temos que o WordPress não dá suporte a composição de componentes na interface e nem a dispositivos móveis.

	Groupware Workbench	IBM Connections	Elgg	Portlets	WordPress
Composição de componentes	Suporta	Suporta	Não suporta	Não suporta	Não suporta
Habilitar / desabilitar em tempo de execução	Suporta	Suporta	Não suporta	Suporta	Suporta
Conhecimento técnico	Baixo	Nenhum	Baixo	Baixo (dependendo do portal)	Suporta
Dispositivos móveis	Suporta	Suporta	Suporta	Não suporta	Não suporta
Licença	LGPL	Privada	GPLv2 e MIT	Especificação aberta	GPLv2

Tabela A1: Comparação do Groupware Workbench com outras tecnologias de componentes