3
Redes Sociais

No contexto da Internet, redes sociais podem ser definidas como locais (sites) em que pessoas se conectam umas às outras através de laços sociais baseados em afinidades, interesses em comum ou em amizades existentes no mundo real. Geralmente, esses interesses são descritos em perfis (*profiles*) que podem ser públicos dependendo do grau de privacidade escolhido pelo usuário.

As redes sociais são primariamente focadas em interações e possuem um enorme potencial de agir como plataformas para o compartilhamento de dados de contexto obtidos dos sensores disponíveis nos dispositivos dos usuários, possibilitando assim a criação de sensores sociais de usuários e de seus contatos na rede social (17). São exemplos bem conhecidos de redes sociais o Facebook, o Twitter, o Orkut e o Google Plus.

3.1
Redes Sociais Pervasivas (RSP)

As redes sociais pervasivas expandem o contexto das redes sociais ao utilizar informações de contexto físico extraídas a partir de sensores no dispositivo móvel dos usuários para agregar valor às informações disponíveis sobre o usuário. O Google Latitude é um exemplo de rede social pervasiva que utiliza a localização do usuário obtida de seu dispositivo conectado a Internet para posicioná-lo em um mapa no qual também poderá ver a posição mais recente de cada um de seus amigos que fazem parte da rede Latitude.

A importância das redes sociais pervasivas é discutida por Rosi et al. (17). Os autores destacam que o potencial dos sensores sociais, provedores de informações sociais, pode ser muito maior que o dos sensores pervasivos em algumas situações e fatos que:

- existem apenas na mente do usuário e não podem ser extraídos por sensores pervasivos, mas podem ser deduzidos a partir de suas interações nas redes sociais;
- são revelados pelos sensores sociais e poderiam também ser revelados pelos sensores pervasivos mas a inexistência destes últimos não impede a descoberta do contexto. Um exemplo é a informação de geolocalização

presente nas fotos compartilhadas no Flickr que pode revelar a localização do usuário mesmo que não haja um sensor de localização;

– expressam informações sobre situações futuras que não poderiam ser deduzidas por outros tipos de sensores. Um exemplo dessa situação pode ser encontrado quando um usuário compartilha no Facebook uma intenção de ir a um certo lugar. Essa intenção não poderia ser deduzida por meio de sensores pervasivos.

Apesar das redes sociais pervasivas possuírem um potencial elevado, Rosi et al. destacam que os sensores sociais possuem os mesmos problemas de precisão que os sensores pervasivos ou até mais. Monitorar o comportamento de um usuário no Facebook e deduzir suas atividades pode requerer uma interação constante com a rede.

As redes sociais devem disponibilizar as informações dos usuários através de APIs que tornam possível o desenvolvimento de aplicações sociais. Nas seções a seguir, serão apresentadas as redes sociais utilizadas neste trabalho e com as quais o Mobile Social Gateway se integra. Também serão apresentadas quais informações relevantes podem ser extraídas de cada uma dessas redes e como elas implementam o protocolo OAuth (42), detalhado na seção 6.1, para autenticação e autorização de requisições às suas APIs.

3.2 Facebook

O conteúdo da plataforma é disponibilizado através da Graph API (18) que permite a leitura e escrita de dados do e para o Facebook. Ela mudou o paradigma de escrita e leitura de uma maneira orientada a métodos para uma que utiliza objetos (perfis de usuários, amigos, *posts*, fotos etc) e seus relacionamentos ou conexões com outros objetos, o que simplifica a API e a torna mais consistente.

A API apresenta uma visão simples e consistente do grafo social do Facebook, representando de forma uniforme os objetos (por exemplo, pessoas, fotos, eventos e páginas) e as conexões entre eles (por exemplo, relações de amigo, conteúdo compartilhado, e *tags* de fotos). Cada objeto no grafo social tem um identificador (ID) único e é possível acessar suas propriedades através de URLs bem formadas:

```
https://graph.facebook.com/Name-orID
```

Os objetos existentes na arquitetura do Facebook são listados abaixo, seguidos de um exemplo:

- Usuário: Victor Pantoja
- Page: página web da Coca-Cola
- Evento: Facebook Developer Garage Austin
- Grupo: grupo de desenvolvedores do Facebook
- Aplicação: o app BuddyPoke
- Mensagens de *status*: uma mensagem de *status* de Markus Endler
- Fotos: uma foto de um evento
- Álbum de fotos: fotos do mural da Coca-Cola
- Fotos de perfil: foto do perfil do usuário
- Vídeos: uma palestra sobre a Graph API do Facebook
- Notas: uma nota anunciando uma app do Facebook para iPhone 3.0
- Check-ins: check-in no "Guinness"pub

Todos os objetos estão ligados uns aos outros através de relacionamentos, chamados conexões (no jargão da API do Facebook). Por exemplo, Victor Pantoja é um fã da página web da Coca-Cola, ou Markus e Victor são amigos. As conexões suportadas para pessoas e páginas incluem: amigos, *newsfeed*, feed do perfil (para o mural do usuário), preferências, cinema, música, livros, notas, permissões de acesso, *tags* de fotos, álbuns de fotografias, *tags* de vídeo, *upload* de vídeo, eventos, grupos, *check-ins*, etc.

A API permite imediatamente o acesso a toda a informação pública disponível sobre um determinado objeto do Facebook. No entanto, para obter informações adicionais ou mesmo enviar um *post* em nome do usuário, é necessário obter um *token* de acesso que deverá ser usado em toda a requisição subsequente. O processo de obtenção deste *token* é descrito na seção 3.2.1.

3.2.1 Processo de Autenticação e Autorização

O processo de autenticação e autorização da API do Facebook utiliza o protocolo OAuth 2.0 e envolve três etapas (19):

1. autenticação do usuário: garante que o usuário é quem ele diz ser (login no Facebook);

2. autorização da aplicação social: garante que o usuário esteja ciente exatamente de quais dados e recursos está disponibilizando para ela;

3. autenticação da aplicação: garante que a aplicação está agindo em nome do usuário.

Existem dois fluxos semelhantes para obtenção do *token* de acesso: *server-side* e *client-side*. O primeiro fluxo é usado nos casos em que a autenticação é controlada pela parte da aplicação que roda no servidor. O segundo é usado quando as chamadas para a Graph API são feitas a partir do frontend desta aplicação, como uma chamada AJAX, ou a partir de um cliente (em um dispositivo móvel).

O processo de autenticação do usuário e de autorização da aplicação são tratados simultaneamente. Inicialmente, o usuário é redirecionado para uma caixa de diálogo OAuth, passando o ID da aplicação, gerado automaticamente quando ela foi criada, e a URL para a qual o usuário será redirecionado quando o processo de autorização for concluído.

Uma vez autenticado, o usuário é convidado a autorizar o acesso da aplicação às suas informações básicas, disponíveis publicamente por padrão no Facebook. Se a autorização for concedida, o usuário será redirecionado para a URL de *callback* especificada acrescida de um código de autorização. Se a aplicação precisar de permissões extras, ela deve solicitá-las ao usuário. Neste ponto, os desenvolvedores devem ter cuidado: de acordo com o Facebook, há uma forte correlação inversa entre o número de permissões que a aplicação solicita e a quantidade de usuários que irão permiti-las. Ou seja: quanto mais permissões solicitadas, menor a quantidade de usuários que irão concedê-las.

Finalmente, no terceiro passo, a fim de ser autenticada, a aplicação deve passar o código de autorização e sua chave secreta para a Graph API. Se autenticado com êxito, o servidor de autorização irá retornar o *token* de acesso com o qual será possível realizar requisições autorizadas em nome do usuário, bastando incluí-lo nas requisições. Por exemplo:

```
https://graph.facebook.com/victor.pantoja?access_token=...
```

retornará informações adicionais sobre Victor Pantoja.

3.2.2 Informações Relevantes Obtidas através do Facebook

A API Graph suporta atualizações em tempo real (20) que permitem que aplicativos subscrevam-se a alterações em determinados objetos ao invés de realizarem *pooling* nos servidores do Facebook. Sempre que ocorre uma alteração em algum objeto assinado, o Facebook faz uma requisição HTTP POST para a URL de *callback* previamente especificada pela aplicação enviando uma lista contendo as alterações realizadas naquele objeto.

Atualmente, as aplicações podem se subscrever aos seguintes objetos:

– usuários: notificações sobre campos específicos e conexões correspondentes a nós de usuário;
– permissões: notificações quando os usuários alterarem as permissões concedidas à aplicação;
– páginas: notificações quando as páginas que possuam integração com o aplicativo alterarem suas propriedades públicas.

Esse mecanismo de interação assíncrona com o Facebook (tipo Publish/Subscribe) é bastante útil para casos em que a aplicação social necessite extrair informações de contexto da rede social em tempo real. Por exemplo, a aplicação de carona ACS poderia, a partir de uma atualização de *status* do usuário e a notificação enviada pelo Facebook, deduzir sua localização e identificar uma mudança de trajetória.

Pode-se extrair algumas outras informações interessantes do Facebook como os *posts* mais relevantes (ou "quentes") que representam o assunto mais discutido no mural do usuário e que, possivelmente, possui um alto grau de relevância. Outra informação são as páginas que o usuário da aplicação móvel "curtiu"e que podem corroborar deduções sobre as suas preferências. Assim, um usuário que curtiu uma página de futebol provavelmente gosta desse esporte.

3.3 Twitter

A API do Twitter (21) permite aos desenvolvedores acessarem algumas das suas primitivas chaves como *timeline*, atualizações de *status* e informações do usuário. O processo de autenticação no Twitter API utiliza o protocolo OAuth 1.0a e é descrito a seguir:

1. a aplicação social deve obter *tokens* de acesso OAuth para agir em nome de um usuário do Twitter e, então;

2. autorizar todas as requisições HTTP feitas à API do Twitter.

Os *tokens* de acesso são obtidos em três etapas. Na primeira, figura 3.1, a aplicação deve obter um request *token* enviando uma mensagem assinada para o Twitter. A solicitação possui um parâmetro oauth_callback com a URL para a qual o usuário será redirecionado após a autenticação. O corpo da resposta irá conter os parâmetros oauth_token_secret, oauth_token e oauth_callback_confirmed.

Na segunda etapa (autenticação e autorização), o usuário é redirecionado para o Twitter, passando o request *token* através do parâmetro oauth_token. Há três cenários possíveis, representados na figura 3.2:

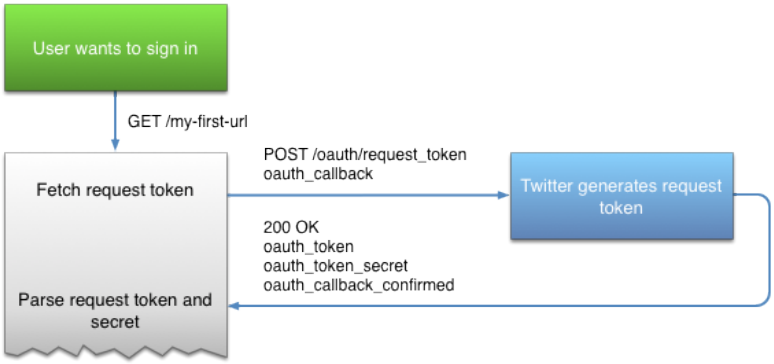

Figura 3.1: Primeira Etapa do Processo de Autenticação do Twitter (22)

1. logado e aprovado: se o usuário está logado e já aprovou as permissões necessárias para a aplicação, ele será imediatamente autenticado;

2. logado mas não aprovado: se o usuário está conectado, mas ainda não aprovou a aplicação, um pedido para compartilhar o acesso com ela será exibido;

3. não logado: se o usuário não está conectado, ele deverá digitar suas credenciais e, então, permitir que o aplicativo acesse suas informações.

Após uma autenticação bem sucedida, o usuário será redirecionado para a URL de callback com um request *token* OAuth válido.

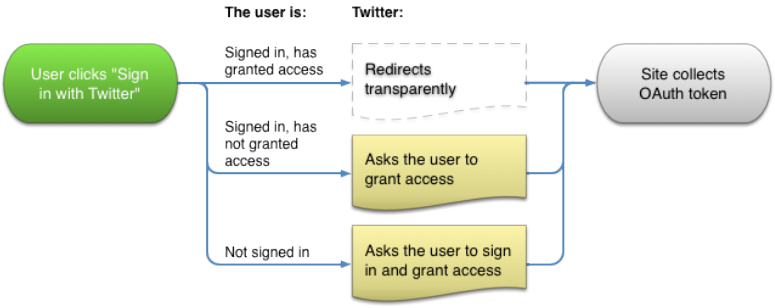

Figura 3.2: Segunda Etapa do Processo de Autenticação do Twitter (22)

Na etapa 3 (figura 3.3), a aplicação deve converter o request *token* em um *token* de acesso, através de uma requisição para o Twitter, contendo o valor oauth_verifier obtido na etapa 2. Em caso de sucesso, a reposta conterá os parâmetros oauth_token, oauth_token_secret, user_id (identificador do usuário no Twitter), e screen_name (apelido do usuário). Esse *token* de autorização deverá ser utilizado nas requisições à API do Twitter.

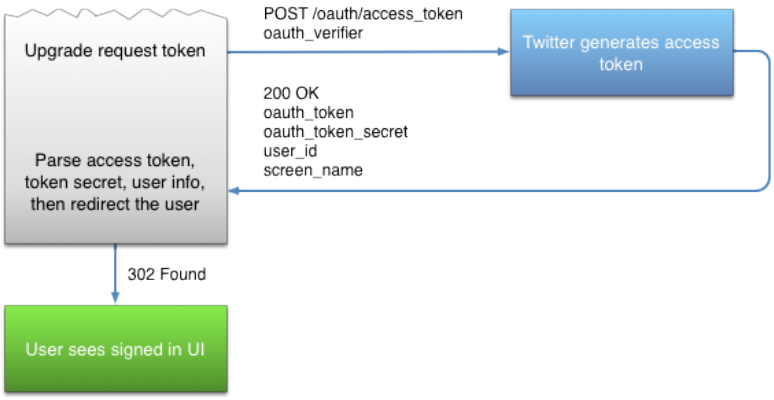

Figura 3.3: Terceira Etapa do Processo de Autenticação do Twitter (22)

3.3.1 Informações Relevantes Obtidas através do Twitter

Da mesma forma que no Facebook, pode-se determinar, a partir de mensagens dos usuários no Twitter (tuítes ou tweets), uma série de informações de contexto dos usuários de forma direta, como a posição geográfica do usuário obtida a partir do GPS do dispositivo e usada como *tag* no *post*, ou de forma indireta caso o usuário mencione o nome do lugar em que se encontra, como por exemplo, o Pão de Açucar. Em ambos os casos (direto e indireto), pode-se determinar o conjunto de pessoas que se encontram no mesmo local, ou seja, que compartilharam o mesmo contexto de "estar no Pão de Açucar"e, então, sugerir novas amizades ou locais de possível interesse nas proximidades.

3.4 Google+

A API do Google+ (23) é centrada em torno de pessoas, atividades e comentários. Cada um desses recursos é representado utilizando-se o formato JSON. Todas as requisições para a API do Google+ para acessar dados não-públicos do usuário devem ser autorizadas por um usuário autenticado. A API utiliza o protocolo OAuth 2.0 para autenticação e autorização. A requisição da aplicação social deve contar um *token* OAuth 2.0 ou a sua chave gerada na sua criação:

- se a requisição necessitar de autorização, ela deverá incluir o *token* OAuth;
- caso contrário, ela deverá incluir a chave da aplicação ou o *token*, ou ambos.

O Google suporta vários fluxos de autenticação, baseados no OAuth 2.0, que cobrem os cenários envolvendo servidores web, JavaScript, aplicativos

instalados em dispositivos. Os seguintes passos são comuns a todos esses cenários:

1. quando a aplicação é criada, ela recebe um ID de cliente e uma chave.

2. quando ela necessita acessar dados do usuário, ele deve solicitar ao Google um escopo específico de acesso.

3. o Google exibe um diálogo OAuth para o usuário, solicitando que ele realize seu login no sistema. Após o login, o usuário visualizará as permissões solicitadas pelo aplicativo e decidirá se quer ou não concedê-las. Este processo é chamado de "consentimento do usuário".

4. se o usuário aprovar, o Google fornecerá ao aplicativo um *token* de acesso de curta duração. *Tokens* de acesso são válidos apenas para o conjunto de operações e recursos descritos quando da sua solicitação. No entanto, ele pode ser utilizado várias vezes para operações similares.

5. o aplicativo solicita os dados do usuário, incluindo o *token* de acesso ao pedido. Os *tokens* de acesso são enviados para a API no cabeçalho do request HTTP ou como um parâmetro de *query string*.

6. se o Google determina que o pedido e o *token* são válidos, ele retorna os dados solicitados.

3.4.1 Informações Relevantes Obtidas através do Google+

A API do Google Plus permite a extração de uma série de informações interessantes, assim como a do Facebook, mas ainda está em um estágio inicial de desenvolvimento. Algumas informações essenciais para aplicações que relacionam os contextos do usuários ainda não estão disponíveis, como a listagem de seus amigos, mesmo na versão autenticada. No entanto, a API pode complementar as informações sobre o contexto atual de um usuário principalmente por tornar disponível sua localização atual, utilizada, inclusive, pelo Google Latitude.

Na prática, poderia-se obter do Facebook algum *post* do usuário dizendo, por exemplo, que ele está indo para um determinado local e através da API do G+ seria possível saber onde ele está no momento e propor uma carona, algo semelhante ao proposto pela aplicação ACS.