

UNIVERSIDADE FEDERAL DO PAMPA

BRUNO CAMPOS DE VASCONCELLOS

**DESENVOLVIMENTO DE UM APLICATIVO INTEGRADO AO FACEBOOK E AO
SISTEMA DO PROJETO IGUAL PARA COMPARTILHAMENTO E
RECOMENDAÇÃO DE OBJETOS DE APRENDIZAGEM**

**Bagé
2013**

BRUNO CAMPOS DE VASCONCELLOS

DESENVOLVIMENTO DE UM APLICATIVO INTEGRADO AO FACEBOOK E AO SISTEMA DO PROJETO IGUAL PARA COMPARTILHAMENTO E RECOMENDAÇÃO DE OBJETOS DE APRENDIZAGEM

Trabalho de Conclusão de Curso apresentado ao Curso de Especialização em Sistemas Distribuídos com Ênfase em Banco de Dados da Universidade Federal do Pampa, como requisito parcial para obtenção do Título de Especialista.

Orientador: Sandro da Silva Camargo

Coorientador: Cristian Cechinel

**Bagé
2013**

BRUNO CAMPOS DE VASCONCELLOS

DESENVOLVIMENTO DE UM APLICATIVO INTEGRADO AO FACEBOOK E AO SISTEMA DO PROJETO IGUAL PARA COMPARTILHAMENTO E RECOMENDAÇÃO DE OBJETOS DE APRENDIZAGEM

Trabalho de Conclusão de Curso apresentado ao Curso de Especialização em Sistemas Distribuídos com Ênfase em Banco de Dados da Universidade Federal do Pampa, como requisito parcial para obtenção do Título de Especialista.

Trabalho de Conclusão de Curso defendido e aprovado em: 1º de agosto de 2013.

Banca examinadora:

Prof. Dr. Sandro da Silva Camargo
Orientador
UNIPAMPA

Prof. Me. Érico Marcelo Hoff do Amaral
UNIPAMPA

Prof.^a Me. Sandra Dutra Piovesan
UNIPAMPA

Dedico este trabalho a meus pais, filho e esposa.

RESUMO

O presente trabalho apresenta o desenvolvimento de um aplicativo integrado ao sistema do Projeto IGUAL – *Innovation for Equality in Latin American Universities* – de maneira que usuários do sistema possam utilizá-lo pelo *Facebook*. O sistema do Projeto IGUAL tem como objetivo a disponibilização de recursos inovadores de aprendizagem relacionados aos conteúdos introdutórios de programação de computadores, assim como possibilitar o desenvolvimento de lições e cursos utilizando esses conteúdos. O *Facebook* conta com diversas ferramentas de comunicação, compartilhamento de arquivos e criação de grupos restritos. Ferramentas estas que podem facilitar a dinamização e a continuidade de estudos, principalmente em tarefas extraclasse. A integração do aplicativo desenvolvido ao *Facebook* foi realizada por meio de *WEB Services REST*. No aplicativo integrado, cada objeto de aprendizagem pode ser acessado via *Facebook*, e os usuários têm a possibilidade de utilizar, avaliar (curtir), realizar comentários e compartilhar os objetos de aprendizagem na rede social. Espera-se que essa integração aumente a procura e a utilização dos materiais didáticos, cursos e lições desenvolvidos e disponibilizados pelo projeto IGUAL. Para facilitar o uso, o próprio aplicativo realiza uma recomendação de objetos de aprendizagem, fazendo com que estudantes acessem novos assuntos com base em suas preferências anteriores.

Palavras-chave: *Facebook*, educação, projeto IGUAL, objetos de aprendizagem, algoritmos de recomendação.

ABSTRACT

The present work presents the development of a tool integrated to the system of IGUAL Project - *Innovation for Equality in Latin American Universities*– so that such system can be used inside *Facebook*. The IGUAL Project system aims to provide innovative learning resources related to the introductory topics of computer programming, as well as to allow the development of lessons and courses using such resources. Facebook provides a variety of tools for communication, file sharing and group creation that can help in the teaching and learning process, especially in non formal environments. The integration of IGUAL Project system was done through the use of REST Web Services. In the integrated tool, each learning resource can be accessed via Facebook and the users are allowed to use, like, make comments and share the resources with their social networks. It is expected that the tool increases the search and the use of the resources, lessons and courses available inside IGUAL Project system. The integrated tool also recommends learning resources to the users based on their previous preferences.

Keywords: Facebook, education, IGUAL Project, learning objects, recommendation algorithms.

LISTA DE FIGURAS

Figura 1 - Tela inicial do sistema do Aprende (Oferta de cursos e lições).....	16
Figura 2 - Exemplo do uso do método eQueryService.....	18
Figura 3 - retorno getList.	19
Figura 4 - retorno do método getLAService	20
Figura 5 - Retorno do método queryByIdService.	21
Figura 6 - Informações Básicas para criação de um Aplicativo Facebook.....	24
Figura 7 - Fluxo de login aplicativo Facebook e retorno.....	25
Figura 8 - Arquitetura básica de um aplicativo Facebook.	27
Figura 9 - Fórmula da correlação de Pearson.....	29
Figura 10 - Matriz de Avaliações.	30
Figura 11 - Fórmula distância euclidiana	30
Figura 12 - Matriz da distância euclidiana.....	31
Figura 13 - Interseção entre dois usuários.	31
Figura 14 - Relação entre as métricas Precision e Recall.	32
Figura 15 - Integração entre componentes Mahout.....	34
Figura 16 - Tela inicial de busca.	36
Figura 17 - Representação gráfica da arquitetura do IGUALS.....	38
Figura 18 - Modelo físico de banco de dados.	40
Figura 19 - Armazenamento de dados para recomendação.	44
Figura 20 - Instalação do IGUALS.	45
Figura 21 - Workflow do IGUALS.	46
Figura 22 - Tela inicial do IGUALS.....	46
Figura 23 - Tela de busca de objetos de aprendizagem.	47
Figura 24 - Recomendação de objetos de aprendizagem.....	48
Figura 25 - Opção curtir lançada no Facebook.	48
Figura 26 - Compartilhar objeto de aprendizado.	49
Figura 27 - Exemplo de objeto de aprendizagem.....	49

LISTA DE TABELAS

Tabela 1 - Requisição ao RDF representação de curso.....	17
Tabela 2 - Retorno no formato XML da representação de curso.	17
Tabela 3 - Retorno de informações básicas a partir da Graph API.....	26
Tabela 4 - Descrição das tabelas do banco de dados.	39
Tabela 5 - Método do aplicativo responsável por redirecionar para URL de autenticação.	40
Tabela 6 - Função que chama Open Graph na opção curtir.	41
Tabela 7 - Consta a chamada com a biblioteca RESTFB.	42
Tabela 8 - Código fonte para chamar WEB Service.....	42
Tabela 9 - Código fonte que retorna atividades buscadas pela recomendação.	42
Tabela 10 - Código fonte para o recomendador utilizando Tanimoto Coefficient.....	45

LISTA DE SIGLAS

API – *Application Programming Interface* (Interface de Programação de Aplicativos)

FQL - *Facebook Query Language* (Linguagem de consulta *Facebook*)

HTTP – *Hypertext Transfer Protocol* (Protocolo de Transferência de Hipertexto)

IGUAL – *Innovation for Equality in Latin American Universities* – (Inovação para Equiparação nas Universidades Latino-americanas)

JSF – *Java Server Faces*

LMS – *Learning Management System* – (Sistema de Gestão de Aprendizagem)

MVC – *Model View Controller* (Modelo Visão Controlador)

ORM – *Object Relation Mapping* (Mapeamento objeto-relacional)

RDF – *Resource description framework* (Framework para descrição de recursos)

REST - *Representational State Transfer* (Transferência de Estado Representativo)

SDK - *Software Development Kit* (Kit de Desenvolvimento de Software)

SGBD – Sistemas Gerenciadores de Banco de Dados

URL - *Uniform Resource Locator* (Localizador-Padrão de Recursos)

SUMÁRIO

1	INTRODUÇÃO.....	11
1.1	Problema de pesquisa	13
1.2	Objetivos	13
1.2.1	Geral.....	13
1.2.2	Específicos	13
1.3	Metodologia.....	13
1.4	Organização do trabalho	14
2	TRABALHOS CORRELATOS	36
2.1	Izzui	36
2.2	Integrar Moodle ao Facebook	36
2.3	Uso do recurso grupos do <i>Facebook</i> como um LMS	37
2.4	eFront	37
3	PROJETO IGUAL.....	15
3.1	Funcionamento do Sistema.....	16
3.2	Arquitetura	16
3.3	Serviços Disponíveis.....	17
3	FACEBOOK	22
3.1	Aplicativos Web Facebook	23
3.3	APIs de Acesso a Dados	26
4.	RECOMENDAÇÃO DE OBJETOS DE APRENDIZAGEM.....	28
4.1	Filtragem Colaborativa	28
4.2	Item-based versus User-based	28
4.3	Medidas de similaridade.....	29
4.3.1	Pearson	29
4.3.2	Distância Euclidiana	30
4.3.3	Tanimoto	31
4.6	Mahout Apache.....	33
4.6.1	Easyrec	34
5.	IGUALS	36
5.1	Introdução.....	37
5.2	Tecnologias utilizadas.....	37

5.3	Arquitetura	38
5.4	Implementação.....	39
5.4.1	Persistência de dados	39
5.4.2	Projeto físico do Banco de Dados	39
5.4.3	Serviços do projeto IGUAL utilizados	42
5.4.4	Recomendador	43
5.5	Instalação	45
5.6	Exemplos de uso.....	45
5.7	Avaliações dos resultados	49
6.	CONCLUSÃO E TRABALHOS FUTUROS.....	51

1 INTRODUÇÃO

Algoritmos são a base da programação de computadores que, por sua vez, é um importante tópico da formação básica de cursos relacionados à computação. Diversos trabalhos já comprovaram a dificuldade existente no aprendizado nessa área, o que conduz a altos índices de evasão e de reprovação. Ainda assim, havendo uma aprovação nestas disciplinas, estudantes podem esbarrar por situações de alta complexidade, reprovando ou evadindo mais à frente no curso (Batalha e Bohadana, 2008). Com o objetivo de diminuir as dificuldades de aprendizado em algoritmos, principalmente para alunos egressos de escolas públicas, foi criado o projeto IGUAL - *Innovation for Equality in Latin American Universities*. Dentro desse projeto está sendo desenvolvendo um sistema WEB, denominado Aprende, que permite a criação de cursos, lições e objetivos de aprendizagem, assim como o cadastro e a disponibilização de objetos de aprendizagem para popular esses cursos e lições. Como resultado desse sistema, espera-se disponibilizar um novo recurso de ensino/aprendizagem, focado primordialmente para alunos egressos de escolas públicas, a fim de que eles possam melhorar seu desempenho, diminuindo suas diferenças técnicas em relação aos egressos de escolas privadas.

Segundo Capobianco (2010), as tecnologias da informação e comunicação (TICs) oferecem recursos para potencializar os processos na área de educação, abrindo novas possibilidades para complementar o ensino formal. O uso de redes sociais no ensino e aprendizagem permite que grupos de pessoas se reúnam com mais facilidade, realizem pesquisas de forma sincronizada, além de compartilhar mídias de diversos tipos, facilitando o aprendizado (Pedagogy in Action, 2013).

Sites de redes sociais, especificamente o *Facebook*, recebem acessos diários do mundo inteiro. Com diversas formas de interação, estes sites são simples, atraindo principalmente os mais jovens. A necessidade de divulgação do projeto IGUAL e o não conhecimento do perfil de usuários abriram espaço para a criação de um aplicativo WEB social, denominado IGUALS (IGUAL Social), vinculado ao *Facebook*, com a proposta de facilitar o acesso à educação e, conseqüentemente, à divulgação de objetos de aprendizagem e, posteriormente dos cursos e lições de programação de computadores. Essa conexão permite a utilização de ferramentas de avaliação, conhecidas como “curtir”, “compartilhar” e “comentários” do site da rede social. Após o uso de forma linear, as informações armazenadas serão reaproveitadas e os objetos de aprendizagem serão recomendados, por meio da filtragem colaborativa.

O número de usuários cadastrados no *Facebook* cresce rapidamente. No Brasil, há em torno de 70 milhões (Socialbakers, 2013). Isto significa uma geração de dados relevantes por parte destes usuários, como suas preferências, localização e idade. O desenvolvimento de aplicativos para o Facebook vem crescendo rapidamente. Empresas vêm explorando esse tipo de solução para definir melhor as preferências de seus clientes e divulgar produtos. Como exemplo, pode ser citado o *Slideshare*¹, além de empresas que desenvolvem aplicativo para comércio eletrônico em redes sociais.

Para que as informações relevantes e permitidas sejam utilizadas por desenvolvedores de aplicativos, os engenheiros de software do *Facebook* criaram uma plataforma de desenvolvimento de aplicativos para ser acessada, através de SDKs (software development kits), pelas mais diversas linguagens de programação e plataformas disponíveis, tanto móveis, como desktop ou WEB. O *Facebook* oferece uma plataforma de desenvolvimento constituída por APIs (*Application Program Interface*) acessíveis por meio de *WEB Service* do tipo REST. As APIs permitem, de acordo com as permissões concedidas ao aplicativo, ler, incluir, alterar e apagar dados de usuários (nome, preferências, opções de curtir, compartilhamento, comentários, álbum de fotos, entre outras).

A integração feita via *WEB Service REST* permite explorar recursos de sistemas distribuídos, demonstrando a capacidade e eficiência desse tipo de mecanismo em ambientes heterogêneos e geograficamente distantes.

Criar um aplicativo social para integrar o Aprende e o Facebook proporciona ao projeto uma maior visibilidade, eleva seu potencial social e leva conhecimentos de programação de computadores a um número maior de usuários.

O uso das informações relacionadas aos usuários do IGUALS permite o recolhimento de metadados avaliativos, que foram utilizados na recomendação de materiais dentro do aplicativo. Desta forma, conforme os usuários utilizarem o sistema, poderão receber do próprio aplicativo, recomendações de objetos de aprendizagem.

¹ <http://www.slideshare.net>

1.1 Problema de pesquisa

O Projeto IGUAL na sua concepção foi definido que seria desenvolvida uma integração do projeto com redes sociais, utilizando recomendação de itens, chamado “Social Sharing”. No entanto, a Universidade responsável pelo desenvolvimento não pode concluir a integração.

O projeto precisa de visibilidade, conhecimento do perfil dos estudantes que o utilizam, que recursos são buscados por eles e identificar a qualidade dos objetos de aprendizagem.

1.2 Objetivos

1.2.1 Geral

Desenvolver um aplicativo social que integre o *Aprende* e o *Facebook*, que disponibilize objetos de aprendizagem e permita a recomendação de objetos de aprendizagem para a comunidade de usuários da ferramenta.

1.2.2 Específicos

- Desenvolver um aplicativo que integre o *Aprende* e o *Facebook*, para disponibilizar objetos de aprendizagem e permitir a recomendação a partir da utilização de informações sociais coletadas no *Facebook*.

- Permitir o compartilhamento de objetos de aprendizagem em redes sociais.

- Coletar informações sobre a qualidade dos objetos de aprendizagem junto à comunidade de usuários de uma rede social de grande utilização.

- Estudar as APIs disponíveis do *Facebook*.

- Estudar as APIs criadas para o *Aprende*.

1.3 Metodologia

Para o desenvolvimento deste trabalho foram realizadas reuniões periódicas com o coordenador do Projeto IGUAL local na UNIPAMPA - campus Bagé, Prof. Cristian Cechinel, para esclarecer objetivos do Projeto e funcionamento do sistema.

Pesquisas foram feitas para obter o levantamento bibliográfico sobre: Facebook relacionados à educação, desenvolvimento de aplicativos para Facebook e sistemas de recomendação.

Para a implementação do IGUALS foram pesquisadas linguagens de programação WEB e um banco de dados de licença livre para persistência de dados. A documentação

do desenvolvimento de *Facebook* foi estudada e integração de aplicações à rede social. Para a integração entre o IGUALS e o Projeto IGUAL foi estudada a documentação do Projeto, especificamente o *WEB Services REST* disponível.

Na implementação do aplicativo, foi utilizada orientação a objetos e mapeamento objeto-relacional na camada de persistência de dados.

Com avaliações levantadas a partir do aplicativo, foi utilizada a biblioteca *Apache Mahout* para implementar um recomendador.

1.4 Organização do trabalho

O trabalho é dividido em mais 5 capítulos, detalhados assim: O Capítulo 2: descreve o funcionamento básico e a arquitetura do Apende. Capítulo 3: descreve os recursos disponíveis para integração de aplicativos com o Facebook e a sua utilização na educação. Capítulo 4 (Recomendação de Objetos de Aprendizagem): seção conceitual e introdutória a sistemas de recomendação e ferramentas disponíveis para construí-los. Capítulo 5 (IGUALS): descreve o objetivo do aplicativo, seu desenvolvimento e uso em detalhes. Capítulo 6 (Conclusão e Trabalhos Futuros): apresenta a conclusão sobre o trabalho e trabalhos futuros.

2 PROJETO IGUAL

Neste capítulo são descritos os objetivos do Projeto IGUAL, as motivações que tornaram o projeto viável, os componentes que formam o Aprende, bem como seu funcionamento.

O IGUAL – *Innovation for Equality in Latin American Universities*² é um projeto financiado pelo programa ALFA III da União Europeia. Atualmente o projeto é desenvolvido por sete universidades, sendo elas: *Tampere University of Applied Sciences* (Finlândia), *Universidad Autónoma de Aguascalientes* (México), *Universidad de los Andes* (Colômbia), *Escuela Superior Politécnica del Litoral* (Equador), Universidade Federal do Pampa (Brasil), *Universitatea Tehnica de Constructii Bucuresti* (Romênia), e *Universidad Austral de Chile* (Chile). As soluções apresentadas pelo projeto IGUAL serão úteis no processo de aprendizagem para estudantes da América Latina principalmente os de escola pública. No meio educacional, as escolas privadas são mais privilegiadas que as escolas públicas por terem mais recursos como: melhores tecnologias, estrutura física e materiais pedagógicos superiores.

Objetos de aprendizagem podem ser disponibilizados pela WEB, em fóruns ou compartilhados de usuário a usuário. Para centralizar e facilitar a disponibilidade surgem os repositórios de objetos de aprendizagem (Ochoa *et al.*, 2012). O projeto IGUAL tem como um de seus principais produtos uma ferramenta WEB, baseada num repositório de objetos de aprendizagem, cujo objetivo é auxiliar no processo de ensino-aprendizagem da programação introdutória de computadores. Ela permite que os professores criem cursos e lições, com base em objetos de aprendizagem existentes no repositório, oferecendo aos acadêmicos um espaço personalizado as suas características de aprendizagem que lhe dê mais recursos para superar suas dificuldades.

Para que a contribuição do projeto IGUAL seja ainda maior, estima-se que em torno de 2000 estudantes de universidades parceiras participarão do projeto piloto.

Objetos de aprendizagem desenvolvidos dentro do escopo do Projeto IGUAL serão focados em auxiliar os estudantes em tópicos básicos de programação. Tais recursos normalmente contêm informações relacionadas com o código fonte de programas que são desenvolvidos em linguagens específicas, tais como: C, Python ou Java (Cechinel *et al.*, 2011).

² IGUAL Project – www.igualproject.org

Os recursos do projeto ficarão disponíveis para acesso de forma pública através do endereço da ferramenta *WEB*, podendo ser utilizados por alunos e professores de qualquer instituição de ensino.

2.1 Funcionamento do Sistema

A ferramenta do projeto IGUAL mantém um banco de dados com cursos, lições e atividades. Dois perfis de usuários são definidos: professores e estudantes. Os professores criam modelos de cursos em uma comunidade *online* e associam seus materiais de aprendizagem. Os estudantes podem seguir cursos e lições e aprender de acordo com suas necessidades. A partir do aplicativo que será apresentado neste trabalho, os estudantes poderão compartilhar objetos de aprendizagem, utilizá-los ou baixá-los no *Facebook*, além de receber recomendações de novos objetos.

A Figura 1 mostra a tela inicial do Aprende onde são oferecidos cursos e lições para os acadêmicos, além de uma opção de busca.

Figura 1 - Tela inicial do sistema do Aprende (Oferta de cursos e lições)



Fonte: Projeto IGUAL

2.2 Arquitetura

O Aprende é composto principalmente pelo repositório e o serviço de recomendação. A forma de armazenamento escolhida é o RDF (*resource description framework*), *4store*, esta escolha feita pela característica de escalabilidade e distribuição.

O repositório armazena e recupera informações do sistema. Diferentemente dos tradicionais bancos de dados relacionais, RDF é utilizado para descrever metadados de forma mais eficiente (Ochoa *et al.*, 2012).

As entidades que compõem o projeto são: *Users, Courses, Lessons, Objectives, Learning Activities, User Actions, Learning Paths* e as *taxonomies*.

A estrutura básica em RDF é composta por *resource, property, value*. Portanto, os recursos baseados em URI (*uniform resource identifier*), são os objetos de aprendizagem, *property* são os campos requisitados e o *value* os valores das requisições. Na tabela 1 é mostrada uma requisição URI, formada por *resource* (recurso), *property* (propriedade) e *value* (valor) ao repositório RDF utilizando as propriedades: *title* e *language*.

Tabela 1 - Requisição ao RDF representação de curso.

Resource	Property	Value
http://igualproject/COURSE/ID#1	http://igualproject/title	Fundamentals of Java
http://igualproject/COURSE/ID#1	http://igualproject/language	en

Fonte: (Ochoa *et al.*, 2012)

Ao fazer uma requisição para salvar uma entidade no repositório, uma estrutura no formato XML é devolvida. Na tabela 2 é mostrado este retorno.

Tabela 2 - Retorno no formato XML da representação de curso.

Course Metadata in xml format
<pre><course> <courseID>1003</courseID> <title>Some course</title> <description>A brief description of the course /description> <language>en</language> <author>xochoa</author> </course></pre>

Fonte: (Ochoa *et al.*, 2012)

Para prover acesso externo, uma API (*Application Programming Interface*) via *WEB Service REST* é disponibilizada.

2.3 Serviços Disponíveis

O Aprende disponibiliza um *WEB Service REST* para acesso ao seu banco de dados, responsável pela comunicação e integração entre o IGUALS e o projeto. O objetivo

é armazenar metadados de entidades, criar, excluir e atualizar objetos de aprendizagem, cursos e lições. Inclui também acesso ao armazenamento de informações sobre o usuário, suas ações e preferências. Os quatro principais métodos que utilizamos nesse trabalho são apresentados a seguir, sendo eles: `eQueryService`, `getListService`, `getLAService` e `queryByIdService`.

2.3.1 eQueryService

O método `eQueryService` é, responsável por receber uma `string` de busca e retornar informações sobre atividades de aprendizado e tem a seguinte declaração:

eQueryService

```
public java.lang.String eQueryService(java.lang.String entity,
                                     java.lang.String input1,
                                     )
```

Como pode ser visto na figura 2, o método `eQueryService` recebe uma `string`, no caso a palavra “Java”. O retorno mostra um arquivo XML com atributos de todos objetos de aprendizagem que contém a palavra na descrição, facilitando a busca para o usuário.

Figura 2 - Exemplo do uso do método `eQueryService`.

```
-<xml>
  <status>OK</status>
  <message>108,109,110,111,112,113,137,140</message>
  -<result_list>
    -<la>
      <id>108</id>
      <title>Exercício Linhas Telefônicas n1</title>
      -<description>
        Exercícios propostos para serem implementados em JAVA: classes, atributos e métodos previamente declarados nos respectivos
      </description>
      <language>pt-BR</language>
      <author>91</author>
      <owner_id>91</owner_id>
      -<bin_location>
        http://200.126.23.124:8080/WebAPI/getLA.jsp?id=108&source=0
      </bin_location>
      -<source_location>
        http://200.126.23.124:8080/WebAPI/getLA.jsp?id=108&source=1
      </source_location>
      <uri_location/>
      <type>ACTIVITY</type>
    </la>
    -<la>
      <id>109</id>
      <title>Exercício Linhas Telefônicas n2</title>
      -<description>
        Implementação e modificação de métodos em classes JAVA, considerando que tais classes já foram desenvolvidas como resultado de
        aprendizagem
      </description>
      <language>pt-BR</language>
      <author>91</author>
      <owner_id>91</owner_id>
```


Figura 4 - retorno do método getLAService.

```

- <lom>
  - <general>
    - <title>
      <string language="pt-BR">Outros operadores utilizados em C++</string>
    </title>
    <language>pt-BR</language>
  - <description>
    - <string language="pt-BR">
      Outros operadores utilizados em C++. como igualdade, AND, OR, XOR, complemento, entre outros
    </string>
    </description>
  - <keyword>
    <string language="pt-BR">AND; OR; C++; operadores</string>
    </keyword>
  </general>
- <technical>
  <format>application/pdf</format>
  - <bin_location>
    http://200.126.23.124:8080/WebAPI/getLA.jsp?id=100&source=0
  </bin_location>
  - <source_location>
    http://200.126.23.124:8080/WebAPI/getLA.jsp?id=100&source=1
  </source_location>
  </technical>
- <educational>

```

Fonte: Gerado a partir do WEB Service.

2.3.4 queryByIdService

O método *queryByIdService* é responsável por retornar objetos de aprendizagem utilizando como parâmetro o tipo de identificador de objetos e os códigos internos de identificação de cada um deles. A declaração está a seguir:

```

public java.lang.String queryByIdService(java.lang.String storageType,
                                         java.lang.String ids)

```

Na figura 5, retorno do método *queryByIdService* com diversos atributos do objeto de aprendizagem, entre eles: título, descrição e localização.

Figura 5 - Retorno do método queryByIdService.

```

-<xml>
  <status>OK</status>
  <message>100:1;101:1;</message>
  -<result_list>
    -<la>
      <id>100</id>
      <title>Outros operadores utilizados em C++</title>
      -<description>
        Outros operadores utilizados em C++. como igualdade, AND, OR, XOR, complemento, entre outros
      </description>
      <language>pt-BR</language>
      <author>91</author>
      <owner_id>91</owner_id>
      -<bin_location>
        http://200.126.23.124:8080/WebAPI/getLA.jsp?id=100&source=0
      </bin_location>
      -<source_location>
        http://200.126.23.124:8080/WebAPI/getLA.jsp?id=100&source=1
      </source_location>
      <url_location/>
      <type>ACTIVITY</type>
    </la>
    -<la>
      <id>101</id>
      <title>Introducao sobre ponteiros em C</title>

```

Fonte: Gerado a partir do WEB *Service*.

Este capítulo mostrou o funcionamento do Projeto IGUAL e do Sistema, ao mesmo tempo em que foram estudados recursos disponíveis para a realização da integração entre o Aprende e aplicativos externos por meio do WEB *Service* REST.

3 FACEBOOK

No presente capítulo serão abordadas estatísticas de uso do *Facebook*, sua capacidade e estudos realizados no meio educacional e alguns exemplos relacionados. Além de recursos de desenvolvimento de aplicativos no *Facebook*, integração entre aplicativos e a rede social.

Segundo UOL (2012), o *Facebook* é a maior rede social do mundo, tornando-se uma ferramenta muito presente no cotidiano das pessoas. Nessa rede social é possível compartilhar imagens, arquivos e conversar com pessoas através de redes de amigos. Toda pessoa cadastrada no *Facebook* fornece informações básicas como nome, e-mail, profissão, atividade profissional, vida acadêmica, estado civil, relacionamento entre pessoas, website e telefones. O conjunto de todas as informações sobre o usuário chama-se perfil. À medida que usuários utilizam a rede, eles podem conectar-se a outras, adicioná-las como amigos e aumentar sua rede. O *Facebook* recomenda novos amigos de acordo as suas relações, como amigos em comum.

De acordo com o próprio *Facebook* (Facebook, 2010), diversas ferramentas disponíveis na rede social podem ser utilizadas no apoio a educação. Por exemplo, usuários podem criar páginas para os conteúdos de suas disciplinas, criar conteúdos e compartilhá-los dentro da rede social, disponibilizando tarefas para os acadêmicos, além de incentivar a criação de grupos e discussões de tópicos específicos.

Apesar de existirem críticas contra o uso do *Facebook* na Educação (Rezac, 2013), diversos autores seguem estudando os impactos dessa rede social como um espaço informal que pode auxiliar no processo de ensino-aprendizagem. Por exemplo, Mazman e Usluel (2010) projetaram um modelo estrutural para explicar a motivação dos usuários em utilizar o *Facebook* com fins educacionais e apontaram que a adoção do *Facebook* está positivamente relacionada com a oportunidade de colaboração, além da comunicação entre os pares e a possibilidade de compartilhar recursos. De acordo com os autores, a maioria dos usuários está entre 18 e 25 anos e são universitários, o que facilita a compreensão do uso da rede social e alta colaboração entre usuários. Ainda Qiyun Wang e Huay Lit Woo (2011), apontam que professores podem usar o recurso de grupos do *Facebook* como um LMS (*learning management system*) para publicar informações de cursos, troca de materiais de cursos com estudantes, e conduzir discussões *online*. Estudantes podem usar os grupos do *Facebook* para atividades de colaboração. De acordo

com Muñoz e Towner (2009), as capacidades do *Facebook* podem beneficiar tanto professores quanto alunos, provendo atividades além da sala de aula, criando uma sala online e aumentando o tempo de interação entre alunos e professores.

Durante as pesquisas para a elaboração deste trabalho, foram encontrados aplicativos para *Facebook* específicos para a educação. Dentre eles, pode-se citar o *WorldCat* – aplicativo que possibilita a busca em bibliotecas de usuários do aplicativo e diversas outras do mundo. Podem ser encontrados livros, *audiobooks*, vídeos, e outras mídias. Disponível em <https://apps.facebook.com/worldcat/> (WorldCat, 2013).

Ractham e Firpo (2011) realizaram um trabalho durante 5 meses entre professores utilizando o *Facebook*. Foram compartilhados conhecimentos e experiências na rede entre alunos e professores e os autores notaram uma interação muito maior e uma aproximação entre alunos e professores.

A fundação *Gates* e o *Facebook* estão se unindo em um evento para reunir os melhores desenvolvedores de aplicativos, especialistas em educação, entusiastas de tecnologia, para desenvolver aplicativos específicos para área de educação em nível superior (Fortune, 2013).


3.1 Aplicativos Web Facebook

Aplicativos WEB para *Facebook* são criados de forma totalmente externa à rede social e integrados à plataforma do *Facebook* através de API (*Application Programming Interface*) providas pela rede. Os aplicativos podem ser criados através de várias linguagens de programação que contenham recursos HTTP.

Redes sociais como o *Facebook* disponibilizam dados por diversas razões, entre elas, promover a rede através de aplicativos de terceiros.

Figura 6 - Informações Básicas para criação de um Aplicativo Facebook.

Aplicativos ▸ Aplicativo Social IGUAL ▸ Básicas



The screenshot shows the 'Basic Information' section of a Facebook application. At the top, there is a profile picture of a blue wireframe head and the application name 'Aplicativo Social IGUAL'. Below this, the 'App ID' is 141423939283586 and the 'App Secret' is 96f0ca646ebf62b273811f1440c3b040 (with a 'redefinir' link). The 'Informações básicas' section contains several fields: 'Display Name' is 'Aplicativo Social IGUAL (Editar...)', 'Namespace' is 'appeduc', 'Contact Email' is 'brunoc.vasconcellos@gmail.com', and 'App Domains' is 'localhost'. The 'Hosting URL' field has a message: 'You have not generated a URL through one of our partners (Get one)'. The 'Sandbox Mode' is set to 'Desativada'. Below this, there is a section titled 'Selecione o modo como seu aplicativo se integra com Facebook' with a checked option 'Site com login utilizando Facebook' and a 'Site URL' field containing 'http://localhost:8080'.

Fonte: Arquivo Pessoal.

Ao criar um aplicativo WEB *Facebook*, é preciso definir algumas configurações como nome do aplicativo, localização externa, *namespace*, email de contato e domínio do aplicativo, e ainda uma localização externa chamada Site URL. As informações básicas solicitadas para a criação de um aplicativo integrado ao *Facebook* estão dispostas na figura 6. Há também informações sobre autenticação entre aplicativo e o *Facebook*, pode-se mencionar o *App ID*, e também uma senha, chamada de *App Secret*.

3.2 Autenticação

Para acessar recursos do *Facebook*, é necessário pedir permissão ao usuário para utilizar suas informações. O *Facebook* utiliza o protocolo *OAuth* na versão 2.0. De acordo com *OAuth* (2013), este protocolo é focado no desenvolvimento do lado cliente de forma simples, enquanto disponibiliza fluxos para autorização em aplicações WEB, *desktop*, *mobile* e dispositivos domésticos.

Para realizar a autenticação e dar acesso aos dados necessários para aplicação, o *Facebook* define algumas categorias de permissão, as principais são: *extends permissions* e *open graph permissions*. As *extends permissions* dão acesso a diversas informações e

permitem a possibilidade de publicar e deletar dados em nome do usuário pelo aplicativo. São divididas em duas categorias: *read permissions* e *publish permissions*.

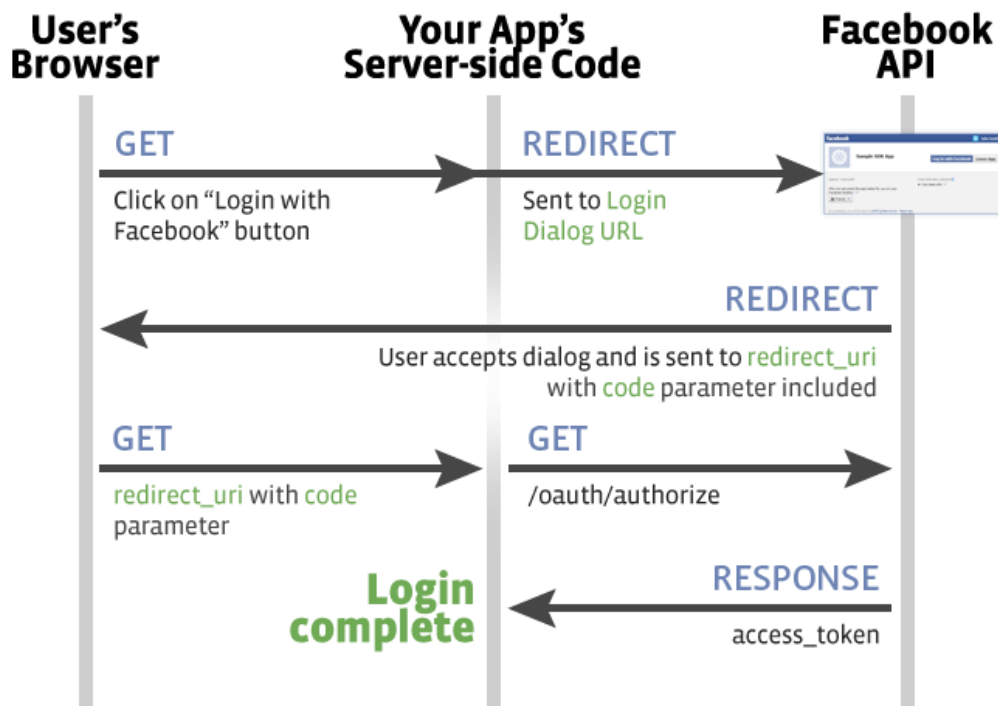
Nas *Read permissions*, pode-se citar como exemplo, a permissão *read_stream* – permite acessar todos os *posts* do mural (espaço onde as mensagens são postadas) e permite que o aplicativo execute buscas no mural (Facebook, 2013a).

As permissões *Publish Permissions*, contêm como principal permissão a *publish_actions*, responsável por permitir que o aplicativo poste conteúdos como comentários e curtir para os amigos no mural (Facebook, 2013a).

De acordo com Facebook (2013b), a *open graph* permite publicar ações definidas no aplicativo e lê-las posteriormente. Como exemplo pode-se citar a opção “curtir”.

O fluxo de *login* começa com a aplicação fazendo uma requisição para a *Dialog URL*, usuário aceita o *Dialog* e é enviado a *redirect_uri* levando o *code*, código gerado pelo *Facebook* para controle interno. Então é retornado um *token access* com um limite de tempo, dando acesso aos recursos necessários. A figura 7 demonstra o fluxo do *login* realizado.

Figura 7 - Fluxo de login aplicativo Facebook e retorno.



Fonte: (Facebook)

3.3 APIs de Acesso a Dados

Para desenvolver aplicativos integrados ao *Facebook* é necessário obter autorização aos recursos que se deseja acessar e também forma de acesso, aqui apresentados como APIs.

Facebook disponibiliza uma série de APIs para auxiliar desenvolvedores a criarem aplicativos sociais que interagem com aplicativos externos (<http://developers.facebook.com/docs>). A primeira API é a *Open Graph*, que permite que aplicativos externos leiam e gravem objetos de conteúdo como fotos, amigos, e as conexões entre eles no *Facebook*. A API é simples e permite que sejam acessados objetos de conteúdo via URLs. Por exemplo, com a devida autorização, acessando a URL <http://graph.facebook.com/brunoc.vasconcellos>, obtém-se informações básicas do usuário em questão (Ko *et al.*, 2010). Na tabela 3, o resultado dessa consulta.

Tabela 3 - Retorno de informações básicas a partir da *Graph* API.

```
"id": "1746776665",
"name": "Bruno Vasconcellos",
"first_name": "Bruno",
"last_name": "Vasconcellos",
"link": "http://www.facebook.com/brunoc.vasconcellos",
"username": "brunoc.vasconcellos",
"gender": "male",
"locale": "pt_BR"
```

De acordo com Ho (2010), para usar a *Facebook* API, uma instância da API é necessária e uma autenticação para que funções sejam utilizadas. A *Graph* API do *Facebook* fornece visualização do grafo social simples e consistente, uniformemente chamado de objetos (curtir pessoas, fotos, eventos e páginas) e conexões entre eles (relacionamentos, curtir e foto tags).

Facebook também provê mecanismo de autenticação para permitir que uma aplicação interaja com a *Graph* API e conseqüentemente usuários do *Facebook*, acessando de forma *single-sign* – forma de realizar *login* apenas uma vez no aplicativo, evitando que necessite fazer várias vezes *login* no Facebook - através da WEB, *mobile*, e aplicativos *desktop*. Estas informações podem ser acessadas por meio da *Facebook* API pelo FQL (*Facebook Query Language*).

A *Graph* API é a primeira forma de acesso para receber dados e enviar dados do *Facebook social graph*. Permite gravar e ler dados de acordo com a permissão do usuário

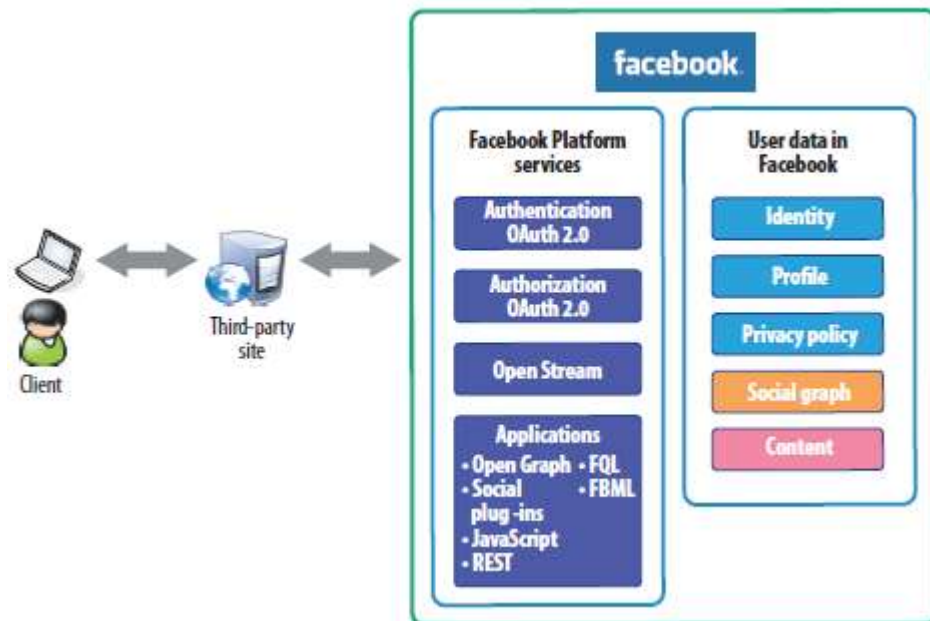
nos aplicativos. Para usar os recursos da API além da permissão, é preciso obter um *token* de acesso, composto pelos dados de acesso ao aplicativo, permissões e *URL* de retorno.

Outra forma de acessar dados é a FQL (Facebook query language), linguagem semelhante ao SQL para acessar dados da Graph API.

Open Graph, permite publicar histórias em nome do usuário e lê-las posteriormente, transformando URL em objetos, criando ações customizadas ou utilizando as existentes.

A figura 8 mostra o que é a arquitetura básica de um aplicativo *Facebook*, os serviços disponíveis para autenticação, o acesso de dados e quais dados estão disponíveis para o acesso do aplicativo.

Figura 8 - Arquitetura básica de um aplicativo Facebook.



Fonte: Ko *et al.* (2010)

O desenvolvimento de aplicativos no *Facebook* é um assunto bastante extenso, que contém uma vasta documentação e dispõe de recursos para acesso aos dados (APIs) de usuários da rede social. Estes dados podem ser compartilhados o que contribui para a divulgação do Projeto IGUAL. Além disso, no capítulo 3 também foi apresentado conhecimento fundamental sobre os recursos que o *Facebook* dispõe para criar e integrar aplicativos na rede social, sendo vitais para o recolhimento de dados para a recomendação de objetos de aprendizagem.

4. RECOMENDAÇÃO DE OBJETOS DE APRENDIZAGEM

Serão mostrados neste capítulo conceitos de filtragem colaborativa e sua relação com sistemas de recomendação, cálculos de medidas de similaridade entre usuários e itens, exemplos práticos de projetos relacionados à filtragem colaborativa e ainda bibliotecas disponíveis para implementação de filtragem colaborativa.

4.1 Filtragem Colaborativa

Filtragem colaborativa é o processo de filtragem de informação ou de padrões, usando técnicas que envolvem a colaboração entre diversos agentes, pontos de vista, fontes de dados, etc. Aplicações da filtragem colaborativa normalmente envolvem conjuntos de dados muito grandes. Este método tem sido utilizado para diversos tipos de dados (Cabré, 2011).

Filtragem colaborativa é um método de fazer previsões automáticas (*filtragem*) sobre os interesses de um usuário, coletando informações sobre o gosto de muitos usuários (*colaboradores*). O pressuposto dessa abordagem é que aqueles usuários que concordaram no passado tendem a concordar novamente no futuro: se João gosta de A e B, e Gabriel gosta de A, B e C, então é provável que João goste de C. É por isso que em filtragem colaborativa são calculadas similaridades entre os usuários ou os itens (Cabré, 2011).

4.2 Item-based versus User-based

É comum recomendarmos algo para alguém, seja um filme, uma viagem ou um restaurante. Para existir uma recomendação é necessário haver preferência por algum item ou alguma relação entre as pessoas. As redes sociais, tais como, o *Facebook*, através de suas funcionalidades e popularidade, contribuem muito com o levantamento dessas informações.

Segundo Zacharski (2012), existem duas formas de distinguir preferências de usuário, sendo elas: *explicit ratings* ou *implicit ratings*. As *explicit ratings* são construídas a partir de informações explicitamente fornecidas pelo usuário como é o caso do IGUALS. Por exemplo, notas dadas por usuários a produtos comprados são consideradas como preferências explícitas, assim como informações de gostar ou não gostar de algo. Já as *implicit ratings* não pedem informação ao usuário e realizam um estudo do seu comportamento do usuário. Como exemplo de preferências implícitas podemos considerar o acesso do usuário a um determinado item, ou a marcação de um item por parte do usuário em uma coleção pessoal (bookmarks).

Na filtragem colaborativa, existem duas categorias: *item-based* e *user-based*. A filtragem colaborativa *user-based* realiza a predição de acordo com situações ocorridas de forma semelhante entre usuários, como por exemplo, compra de um mesmo produto num site de *e-commerce* e calcula a similaridade entre os produtos e usuários. Os principais problemas da *user-based* são escalabilidade e esparsividade de dados. Escalabilidade, porque a tendência é que o número de usuários cresça muito, isto significa calcular similaridade para milhões de usuários, o que requer um poder computacional grande. A esparsividade se dá porque as avaliações podem ser dadas apenas para alguns itens, fazendo com que o cálculo de similaridade falhe nos casos onde não há uma avaliação.

Na filtragem colaborativa *item-based*, as recomendações são feitas a partir de avaliações de itens. Uma matriz com as avaliações é gerada entre itens e usuários, a partir disso uma similaridade (vizinhança) é buscada entre os itens avaliados e recomendados para o usuário ativo.

4.3 Medidas de similaridade

As medidas de similaridade buscam uma semelhança entre dois itens por meio de modelos matemáticos, como distância entre dois pontos e distância entre ângulos.

4.3.1 Pearson

Coefficiente de correlação de *Pearson* é uma medida de similaridade baseada na distância entre duas séries de números. Esta faz uma comparação matemática entre os valores avaliados pelos usuários, usando uma escala, resultando em um intervalo entre 1 e 0. Quanto mais próximo ao 1, mais alta é a similaridade e quanto mais próximo de 0, mais baixa é a similaridade entre os itens comparados.

O cálculo da medida de similaridade entre usuários e itens deve ser dado pela fórmula na figura 9.

Figura 9 - Fórmula da correlação de Pearson.

$$r = \frac{1}{n-1} \sum \left(\frac{x_i - \bar{X}}{s_x} \right) \left(\frac{y_i - \bar{Y}}{s_y} \right)$$

Na matriz da figura 10, é possível ver uma demonstração prática do uso de *Pearson*. Uma matriz de usuários x itens com 5 usuários e 3 itens. De acordo com a coluna "Correlation with user 1", os usuários 1 e 5 têm maior correlação porque os

usuários avaliaram o mesmo item e também porque as avaliações foram com escalas próximas.

Figura 10 - Matriz de Avaliações.

	Item 101	Item 102	Item 103	Correlation with user 1
User 1	5.0	3.0	2.5	1.000
User 2	2.0	2.5	5.0	-0.764
User 3	2.5	-	-	-
User 4	5.0	-	3.0	1.000
User 5	4.0	3.0	2.0	0.945

Fonte: Owen *et al.* (2011)

4.3.2 Distância Euclidiana

É a distância entre dois pontos pelo menos bidimensionais num plano cartesiano. Quanto menor a distância entre os itens de preferências do usuário, maior é a similaridade.

A distância euclidiana entre dois pontos é calculada com a fórmula da figura 11.

Figura 11 - Fórmula distância euclidiana

$$d_{ii'} = \left[\sum_{j=1}^p (X_{ij} - X_{i'j})^2 \right]^{\frac{1}{2}}$$

Na figura 12 é mostrada uma matriz com avaliações entre 5 usuários e 3 itens. De acordo com a coluna “Distance”, a maior similaridade está entre o usuário 1 e o 4, devido as suas distâncias menores.

Figura 12 - Matriz da distância euclidiana.

	Item 101	Item 102	Item 103	Distance	Similarity to user 1
User 1	5.0	3.0	2.5	0.000	1.000
User 2	2.0	2.5	5.0	3.937	0.203
User 3	2.5	-	-	2.500	0.286
User 4	5.0	-	3.0	0.500	0.667
User 5	4.0	3.0	2.0	1.118	0.472

Fonte: (Owen *et al.*, 2011)

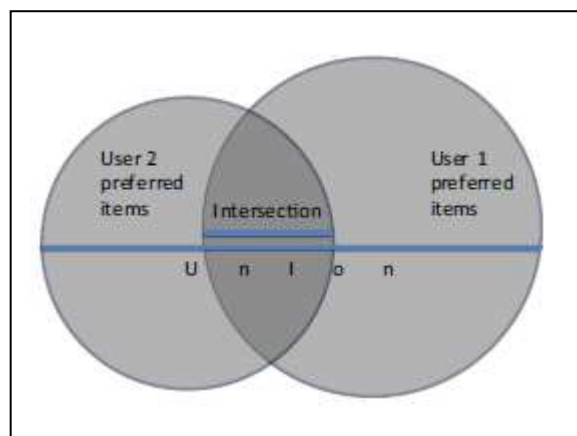
4.3.3 Tanimoto

Métrica que avalia apenas valor booleanos (0 e 1), isto é, se o usuário gostou ou não de um item. Verifica o número em comum existente de itens entre dois usuários, relacionado com o total de itens externos e o total de itens que o usuário deu preferência.

O aplicativo proposto não mantém dados sobre avaliações de objetos de aprendizagem em nível de escala. Esta métrica de similaridade utiliza dados disponíveis de forma booleana, sendo assim vem a ser a forma correta de elaborar a recomendação no IGUALS, tendo em vista que as informações de preferências no *Facebook* estão no formato booleano, i.e., o usuário clicou em “Gostar” ou não clicou.

Na figura 13 a área escura demonstra a interseção entre as preferências de dois usuários. Dados estes que são avaliados pela fórmula de *Tanimoto*.

Figura 13 - Interseção entre dois usuários.



Fonte: Owen *et al.* (2011)

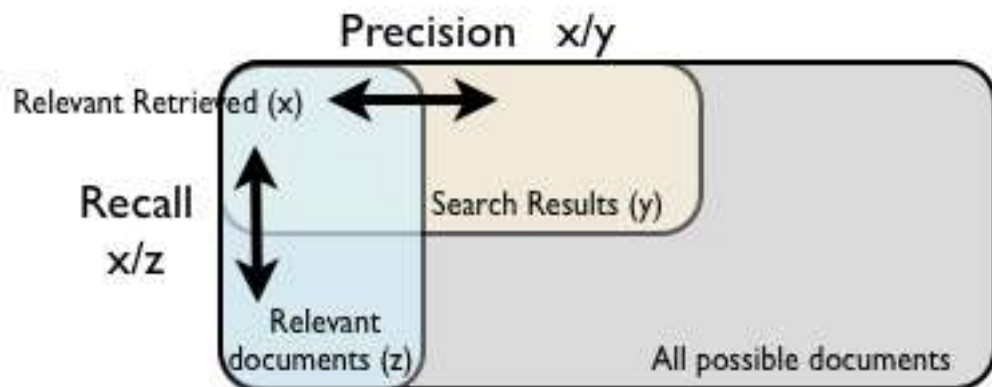
Representado pela fórmula $\tau = NAB / NA + NB - NAB$. Onde NAB representa o número de itens com as mesmas preferências em A e B, NA as preferências em A e NB as preferências em B.

4.4 Métricas para avaliação de resultados

Em um sistema de recomendação, é necessário avaliar a qualidade das recomendações, como por exemplo, comparar com recomendações reais e identificar as diferenças. Para desempenhar este papel existem métricas como *precision*, *recall*.

A métrica *Precision* é obtida pela razão entre os itens relevantes e os itens do resultado, no caso da figura 14, X/Y, onde X são itens relevantes e Y, os resultados da busca. Já a métrica *recall* estabelece a relação entre os itens relevantes e o total de itens relevantes. Como mostra a figura 14, X/Z, onde X itens retornados e Z itens relevantes.

Figura 14 - Relação entre as métricas Precision e Recall.



Fonte: Caraciolo (2011)

4.5 Exemplos de soluções de filtragem colaborativa

Movielens é um site de recomendação de filmes que utiliza filtragem colaborativa para gerar as recomendações. O site é mantido pelo grupo de pesquisa *GroupLens* do Departamento de Ciência da Computação e Engenharia da Universidade de Minnesota, que visa o estudo de sistemas de recomendação e outras áreas.

O *Movielens* é uma plataforma experimental que através de situações reais, auxilia nos estudos em sistemas de recomendação do grupo (Movielens, 2013).

Outro exemplo é o IMDB, que é um site da *Amazon Company*, disponibiliza um banco de dados de vídeos, TV e conteúdo sobre celebridades. Este site utiliza sistemas de recomendação baseados nas avaliações concedidas pelos usuários (IMDB, 2013).

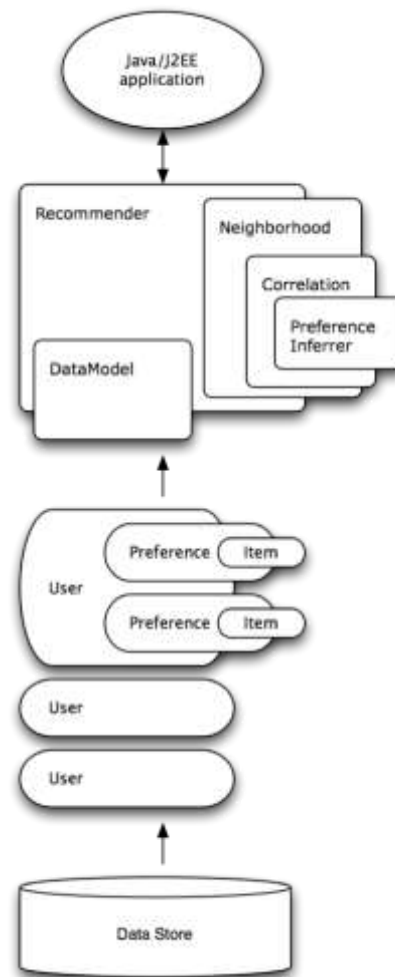
4.6 Mahout Apache

Lançado em 2008, o *Mahout* é uma biblioteca *open source*, criada pela *Apache*, escrita em *Java*, e com recursos para aprendizagem de máquina, classificação, *clustering* e recomendação. Esta suporta diversos algoritmos, entre eles, filtragem colaborativa *user-based* e *item-based*. De acordo com Owen *et al.* (2011), o *Mahout* é também altamente escalável, isto é, permite trabalhar com grandes *datasets* distribuídos em diversos computadores ou apenas com um único servidor. Para realizar escalabilidade é utilizado o *Apache Hadoop – framework* utilizado para realizar processamento distribuído em conjunto de dados de grande porte por meio de *cluster* de computadores usando modelos de programação simples (Hadoop , 2013).

Como o *Apache Mahout* é uma biblioteca, é integrável a aplicações *Java* de forma fácil, uma delas através do *Maven*, que é um software para automação de compilação de aplicações *Java*, facilitando e mantendo a integridade de dependências de pacotes.

Na figura 15 é mostrada a arquitetura de integração entre os pacotes de recomendação disponíveis.

Figura 15 - Integração entre componentes Mahout.



Fonte: Drost (2011)

O *Mahout* tem suporte a diversos SGDBs (Sistemas Gerenciadores de Banco de Dados) atuais tais como *MySQL* e *PostgreSQL*, e atualmente está na versão 0.7. É constantemente atualizado pela *Apache* e por outros desenvolvedores *open source*.

4.6.1 Easyrec

Engine para utilização de recomendação em WEB Sites, *open source*, atualmente em versão Beta. Escrita em Java, é disponibilizada e integrada à aplicação por meio de uma REST API (Easyrec, 2013). As avaliações são enviadas para um servidor *Easyrec* que processa as avaliações e retorna recomendações entre itens. Isto torna o processamento menos flexível e disponibiliza menos recursos para o desenvolvedor.

Para as recomendações de objetos de aprendizagem do IGUALS, foi utilizado o algoritmo de similaridade *Tanimoto* disponível na biblioteca *Apache*. Utilizamos o tipo de filtragem colaborativo denominado *user-based*.

É possível ver que há diversas técnicas para implementar filtragem colaborativa, variando de acordo com os dados disponíveis, contexto e capacidade computacional. A implementação prática dos capítulos vistos até agora estão no próximo capítulo.

5. IGUALS

O capítulo 5 entra em detalhes do uso dos conceitos visto até o momento. São descritas as tecnologias utilizadas na implementação do IGUALS e exemplos de código-fonte desenvolvidos durante o trabalho.

5.1 Trabalhos Correlatos

Aqui serão apresentados trabalhos com características educacionais e uso no *Facebook*. Foram pesquisados trabalhos com certa semelhança ao desenvolvido neste trabalho.

5.2 Izzui

Aplicativo *IZZUI* permite usuários publicarem apresentações e cursos sobre qualquer assunto no *Facebook*. Ainda conta com um sistema de recomendação para os usuários.

Na figura 16 a tela inicial do aplicativo, onde é mostrada uma busca por cursos, descrição ou autor de cursos.

Figura 16 - Tela inicial de busca.



Fonte: Izzui (2013)

5.3 Integrar Moodle ao Facebook

De acordo com (Braz *et al.*, 2011), o trabalho tem como objetivo agregar o *Facebook* para utilizá-lo como plataforma de comunicação entre alunos e professores e para explorar o seu potencial de atrair e engajar os estudantes. Os autores comentam também que a integração é feita por meio de um plugin desenvolvido para o Moodle. O professor deve habilitar o uso, e preferencialmente vincular sua conta do *Moodle* ao do *Facebook*. Estudantes devem fazer o mesmo. A partir disso, quando uma informação é gerada no *Moodle*, ela é automaticamente replicada no *Facebook*, disponibilizando a informação para discussão dos envolvidos. Ainda (Braz *et al.*, 2011), o mecanismo permite que os estudantes utilizem recursos do *Moodle* a partir do *Facebook*.

5.4 Uso do recurso grupos do *Facebook* como um LMS

Os autores exploraram o recurso Grupos do *Facebook* como um LMS (learning managing system). A rede social é utilizada num estudo em dois cursos para realizar anúncios, compartilhar recursos, organização de tutoriais semanais e conduzir discussões *online*, na formação de professor em Singapura (Wang *et al.*, 2012).

O autor conclui que os recursos acima citados podem ser utilizados nos objetivos traçados, ainda que não tenha suporte a *upload* de arquivos diretamente e as discussões não sejam bem estruturadas.

5.5 eFront

LMS com versão *open source*. Contém um *plugin* que permite a integração da plataforma com o *Facebook*. Ao utilizar o LMS integrado ao *Facebook*, algumas informações do *Facebook* como o *status* e *avatar* são vinculadas (Gogos, 2013).

5.6 Introdução

O IGUALS, desenvolvido neste trabalho, permite a integração ao *Facebook* das atividades de aprendizagem (cursos e lições) e seus objetos de aprendizagem relacionados. Isto dá uma visibilidade ao projeto IGUAL, fazendo com que estudantes possam divulgar para seus amigos e aprender quando e como quiserem, bastando ter uma conta no *Facebook* e o aplicativo instalado. Para visualizar as atividades de aprendizagem, foi criada uma forma de realizar uma busca pela descrição da atividade. Cada atividade contém uma descrição, um título e opções de “curtir”, “comentar”, “ver comentários”, “compartilhar” e “baixar”.

As informações cadastradas são inseridas em um banco de dados do aplicativo e inseridas no *Facebook* automaticamente, na forma de opções “comentário”, “curtir” e “compartilhar” e “baixar”.

5.7 Tecnologias utilizadas

O IGUALS foi desenvolvido, utilizando a linguagem de programação Java EE sendo executada sobre o servidor de aplicações *GlassFish*. Com a finalidade de abstrair o desenvolvimento da interface gráfica, foi utilizado o framework MVC (*model view controller*) *Java Server Faces*, somado a um conjunto de componentes de interface gráfica chamado *Prime Faces*. Para a camada de persistência, utilizamos o *Hibernate*, que é um

framework para mapeamento objeto-relacional em Java – juntamente com o banco de dados PostgreSQL.

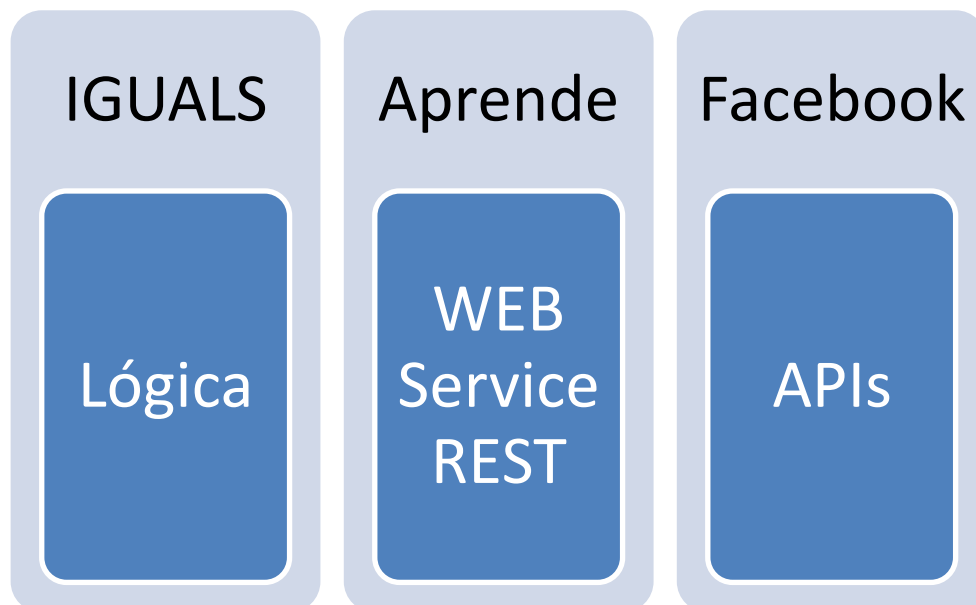
De acordo com RestFB (2013), a RESTFB é um cliente simples e flexível, escrito em Java, para acessar a Graph API Facebook. Está em um nível maduro e permite o uso de diversos dados do Facebook de forma padronizada, simplificando o desenvolvimento de aplicativos para a rede social. Utilizando este mecanismo os objetos de aprendizagem podem receber comentários, opções de “curtir” e serem compartilhados.

Para a recomendação, foi utilizado o *Apache Mahout*, cujo objetivo é implementar técnicas de aprendizado de máquina de forma escalável (Mahout, 2013), por meio de seus algoritmos implementados na linguagem de programação *Java*.

5.8 Arquitetura

A arquitetura é composta pelo IGUALS, *Facebook* e o *Aprende*. O último disponibiliza um WEB Service REST para acesso aos objetos de aprendizagem. Pelo aplicativo é feito o acesso a esse WEB Service e realizadas chamadas às APIs do Facebook, armazenando e disponibilizando os dados no mural do usuário. Estes dados são recolhidos e utilizados para o sistema de recomendação. Na figura 17 é possível visualizar a arquitetura do sistema.

Figura 17 - Representação gráfica da arquitetura do IGUALS.



Fonte: Arquivo Pessoal.

5.9 Implementação

Esta seção apresenta a implementação efetiva do IGUALS.

5.9.1 Persistência de dados

Os dados utilizados por meio do aplicativo são armazenados em servidor de banco de dados local com PostgreSQL instalado nas suas configurações padrão.

Utilizada a técnica de ORM (*object-relation mapping*) para mapear as tabelas do banco de dados e transformá-las em objetos Java.

5.9.2 Projeto físico do Banco de Dados

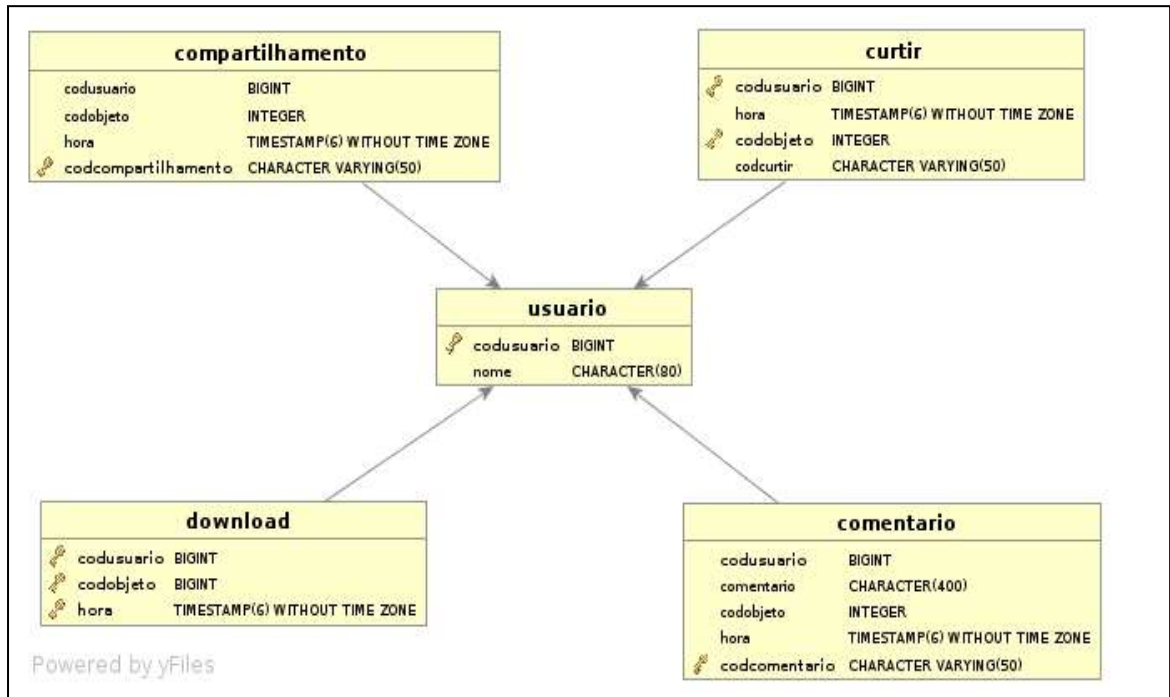
O projeto físico do banco de dados foi criado para manter os dados oriundos das ações de cada usuário do IGUALS, pode-se citar: “curtir”, “compartilhamento”, “usuario”, “comentario” e “download”. A tabela 4 consta a descrição das tabelas criadas.

Tabela 4 - Descrição das tabelas do banco de dados.

Tabela	Descrição
usuario	Armazena o código e nome referente ao usuário conectado ao Facebook.
curtir	armazena o código do usuário que curtiu o objeto de aprendizagem, hora do ocorrido, o código do objeto de aprendizagem e o código gerado pelo Facebook.
compartilhamento	armazena o código do usuário que compartilhou o objeto de aprendizagem, o código do objeto de aprendizagem, hora do ocorrido, e o código do compartilhamento gerado pelo Facebook
download	armazena o código do objeto de aprendizagem, código do usuário e hora do ocorrido
comentario	armazena código do usuário que comenta, o comentário, código do objeto curtido, hora do ocorrido e código do comentário gerado pelo Facebook.

O modelo físico resultou nas 5 tabelas dispostas na figura 18.

Figura 18 - Modelo físico de banco de dados.



A responsabilidade do armazenamento de objetos de aprendizagem é feita pelo Aprende e acessada pelo IGUALS por meio do WEB Service REST disponível.

5.9.2.1 APIs do Facebook utilizadas

Na implementação do aplicativo foram utilizados todos os recursos disponíveis para ler e gravar dados no Facebook. Foram feitas chamadas pela Graph API, publicações com Open Graph, leitura de dados com FQL e autenticação e autorização através da OAuth Dialog.

5.9.2.2 Autenticação Facebook

Para realizar a autenticação com permissões para o usuário, foi feita uma chamada a *OAuth Dialog* via protocolo *OAuth 2.0*.

A autenticação do *Facebook* é feita através do método `getLoginRedirectURL`, que contém a URL de acesso e as permissões necessárias para o IGUALS. A tabela 5 mostra o método que redireciona à URL de autenticação ao *Facebook*.

Tabela 5 - Método responsável por redirecionar para URL de autenticação.

```
public static String getLoginRedirectURL() {
    return "https://www.facebook.com/dialog/oauth?client_id=" + MY_APP_ID +
"&state=full&display=page&redirect_uri=" + redirect_uri +
"&scope=read_friendlists,user_activities,user_likes,publish_stream,email,publish_actions,user_actions:igualapp"; }
```

5.9.2.3 Curtir

O Open Graph contém uma opção de *like* (curtir), com ela é possível criar botões (componente de interface gráfica) integrado a aplicativos (Facebook, 2013c).

Open Graph, é criada uma ação do tipo “og.like” que registra o curtir no objeto de aprendizagem especificado, sendo realizadas chamadas individuais para cada atividade de aprendizado. A função em Javascript é mostrada na tabela 6.

Tabela 6 - Função que chama Open Graph na opção curtir.

```
function Curtir(objeto, id_usuario, token) {
  FB.api(
    'https://graph.facebook.com/me/og.likes',
    'post',
    { object: 'https://apps.facebook.com/appigual/objeto/'+objeto,
      access_token : token,
      privacy : {'value':'EVERYONE'}
    },
    function(response) {
      if (response.error) {
        PrimeFaces.cw('Growl','widget_frmBusca_lista_mensagem',
          {id:'frmBusca:lista_mensagem'
            ,sticky:false
            ,life:2000
            ,escape:true,msgs:[{summary:"Curtir",detail:"",severity:'error'}]}
        );
      }
      else {
        curtirServlet(objeto, id_usuario, response.id);
        PrimeFaces.cw('Growl','widget_frmBusca_lista_mensagem',
          {id:'frmBusca:lista_mensagem'
            ,sticky:false
            ,life:2000
            ,escape:true,msgs:[{summary:"Curtir",detail:"",severity:'info'}]}
        );
      }
    }
  );
}

```

5.9.2.4 Compartilhar

Na opção compartilhar, realizou uma chamada ao *feed Dialog*, interface gráfica padrão do *Facebook*, disponível para compartilhamento e inserção de comentário por parte do usuário, para recuperar os dados foi utilizada a linguagem FQL.

5.9.2.5 Comentar

Na opção comentar, realizamos uma chamada apela *Graph API*, por meio da *RestFB*, incluindo um comentário novo para a atividade de aprendizagem em questão.

Tabela 7 - Consta a chamada com a biblioteca RESTFB.

```

FacebookClient fb = new DefaultFacebookClient(token);
FacebookType retornoComentario =
fb.publish("me/feed", FacebookType.class,
Parameter.with("message", "IGUALS - Objeto de aprendizagem "+titulo+"-
https://apps.facebook.com/appigual/faces/index.xhtml "+this.getComentariounovo()));

```

5.9.3 Serviços do projeto IGUAL utilizados

Com o WEB Service REST do Aprende, foram mapeados os dados retornados nas consultas dos métodos *eQueryService*, *queryByIdService* e *getListService*. Estes métodos têm como retorno padrão, XML.

No método *eQueryService*, busca objetos de aprendizagem a partir da descrição. Este método recebe os parâmetros de valor a ser buscado e que tipo de busca será feita, são elas: “*all, title, description, author, owner_id*”. Aqui foi configurado como “*description*”.

Na tabela 8 é mostrado o código fonte utilizado para efetuar os objetos de aprendizado.

Tabela 8 - Código fonte para chamar WEB Service.

```

public void busca() {
    try {
        FormDataMultiPart form = new FormDataMultiPart().field("entity","LA");
        form.field("input_1", this.getBuscaObjeto().toUpperCase());
        form.field("type_input_1","description");

        Unmarshaller.ObjetosXML response = webResource.path("e_query")
            .accept(MediaType.APPLICATION_XML)
            .type(MediaType.MULTIPART_FORM_DATA).post( Unmarshaller.ObjetosXML.class, form);
        this.setRetorno(response.getLaList());
    } catch (Exception e){
        FacesContext.getCurrentInstance().addMessage(null, new
        FacesMessage(FacesMessage.SEVERITY_ERROR,e.toString(), "Erro!"));
    }
}

```

O método *queryByIdService* retorna os objetos de aprendizagem selecionados. Recebendo, como parâmetro, o tipo e os códigos das atividades. Um XML com uma lista de objetos de aprendizagem é retornado. O código fonte do método é mostrado na tabela 9.

Tabela 9 - Código fonte que retorna atividades buscadas pela recomendação.

```

public List<La> buscaID(){
    try {

```

```

FormDataMultiPart form = new FormDataMultiPart();
    form.field("type", "LA");
form.field("ids",this.getIdobjeto());

    Unmarshaller.ObjetosXML response = webResource.path("queryById")
        .type(MediaType.MULTIPART_FORM_DATA).post( Unmarshaller.ObjetosXML.class,
form);
        return response.getLaList();
    } catch (Exception e){
        FacesContext.getCurrentInstance().addMessage(null, new
FacesMessage(FacesMessage.SEVERITY_ERROR,e.toString(), "Erro!"));
    }

    return null;
}

```

O método *getListService* retorna todos os objetos de aprendizagem cadastrados. Recebendo como parâmetro a *string* “LA”, que indica para o método que deve trazer apenas os objetos de aprendizagem.

5.9.4 Recomendador

Durante o desenvolvimento deste trabalho, foi implementado um sistema de recomendação baseado nos algoritmos disponíveis no Apache Mahout.

De acordo com Lamm (2013), na filtragem colaborativa, temos alguns algoritmos que utilizam dados booleanos. Isto significa que um usuário pode ou não gostar de algum item. A medida de similaridade para este caso é *Tanimoto*. Baseado nisso, o dado utilizado para realização da recomendação foi à opção curtir, isto porque o usuário do aplicativo só poderá curtir ou não o objeto de aprendizagem.

Os dados utilizados, ainda em ambiente de testes, foram os da figura 19. Onde constam usuário que curtiu o objeto de aprendizagem, hora e o objeto de aprendizagem.

Figura 19 - Armazenamento de dados para recomendação.

codusuario [PK] bigint	hora timestamp without time zone	codobjeto [PK] integer
1	2013-06-11 20:12:00.581363	100
1	2013-06-11 20:12:00.581363	101
1	2013-06-11 20:12:00.581363	104
1	2013-06-11 20:12:00.581363	105
1	2013-06-11 20:12:00.581363	106
1	2013-06-11 20:12:00.581363	107
17467766	2013-05-27 14:39:51.206638	100
17467766	2013-05-27 14:39:51.206638	101
17467766	2013-05-27 14:39:51.206638	104
17467766	2013-06-05 23:44:11.431	105
17467766	2013-05-27 14:39:51.206638	108
17467766	2013-06-05 23:09:18.011	140
17467766	2013-05-27 14:39:51.206638	152
10000393	2013-05-26 14:39:51.206638	10
10000393	2013-06-05 00:31:42.666	11
10000393	2013-05-25 14:39:51.206638	100
10000393	2013-05-25 14:30:51.206638	101
10000393	2013-05-29 14:20:51.206638	102
10000393	2013-05-27 11:53:47.028	105
10000393	2013-06-05 00:30:29.241	106
10000393	2013-06-05 23:54:53.134	107
10000393	2013-05-27 14:39:51.206638	108
10000393	2013-06-05 23:51:38.165	109
10000393	2013-05-24 16:36:50.803	110
10000393	2013-05-27 14:39:51.206638	144
10000393	2013-05-27 14:39:51.206638	146

A fim de ser realizada a implementação do recomendador no *Apache Mahout*, é necessário um modelo de dados, que irá armazenar as preferências dos usuários, tais como: item, código do usuário, e hora; número de usuários para identificar a vizinhança entre eles, e um coeficiente de similaridade, neste caso, *Tanimoto*. O código fonte

implementado utilizando o *Apache Mahout*, está na tabela 10. Lá é possível ver o uso da fonte de dados, os campos disponíveis no banco de dados e o uso do coeficiente de *Tanimoto*.

Tabela 10 - Código fonte para o recomendador utilizando *Tanimoto Coefficient*.

```

pgmodel = new PostgreSQLBooleanPrefJDBCDataModel(datasource,"curtir","codusuario","codobjeto","hora");
tanimoto = new TanimotoCoefficientSimilarity(pgmodel);
neighborhood = new NearestNUserNeighborhood(100, tanimoto, pgmodel);

Recommender recommender = new GenericUserBasedRecommender(pgmodel, neighborhood, tanimoto);

itensList = recommender.recommend(Long.valueOf(this.getUserid()), 20);

```

5.10 Instalação

Para tornar funcional o IGUALS, é necessário instalá-lo. Para o correto funcionamento do aplicativo são exigidas algumas permissões para que o usuário possa utilizá-lo, tais como nome e publicar em seu nome.

Na figura 20 é possível ver o momento da instalação, o componente *Dialog* do *Facebook*, pedindo algumas permissões de acesso ao usuário.

Figura 20 - Instalação do IGUALS.



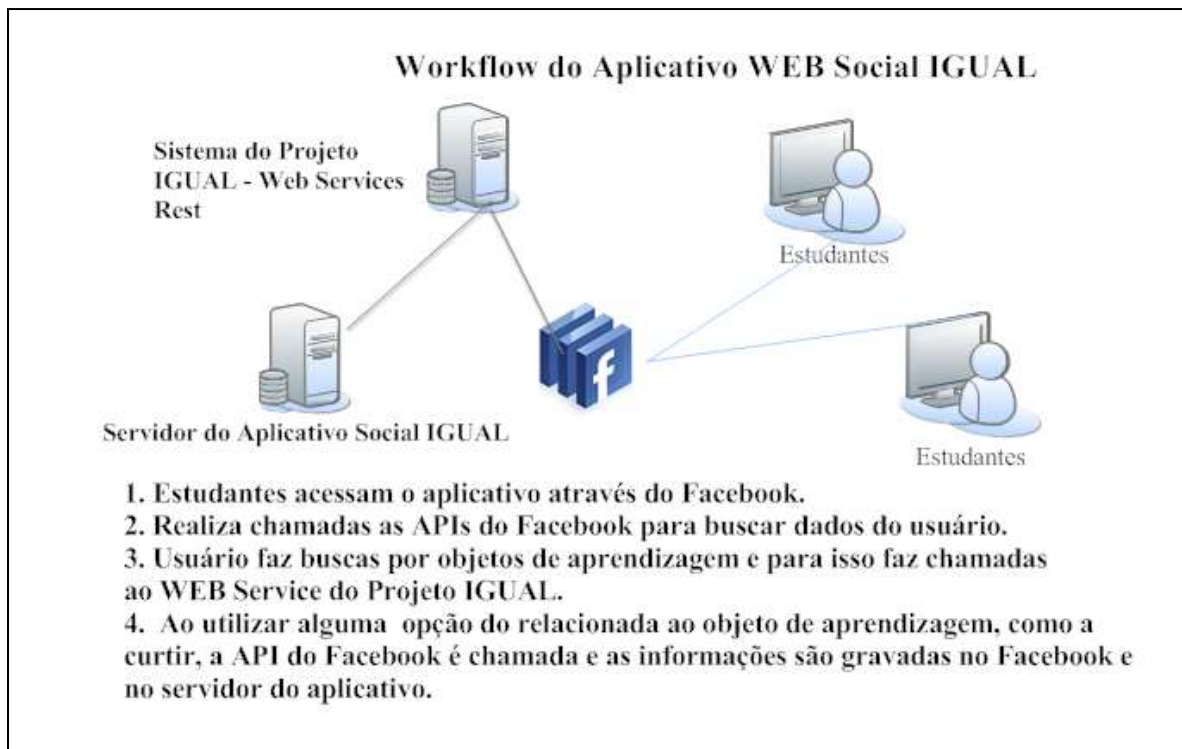
5.11 Exemplos de uso

A figura 21 mostra o workflow do uso do IGUALS, inclusive a integração entre o *Facebook* e o projeto IGUAL.

Na tela inicial, primeiramente é mostrada uma opção de busca pela descrição dos objetos de aprendizagem, se já houver utilização do aplicativo, sugestões serão mostradas logo abaixo da busca.

Em cada atividade de aprendizagem é possível avaliar com “curtir”, “compartilhar” e postar comentários sobre cada uma delas. Além disso é possível visualizar e realizar download de cada objeto de aprendizagem, caso este esteja cadastrado.

Figura 21 - Workflow do IGUALS.



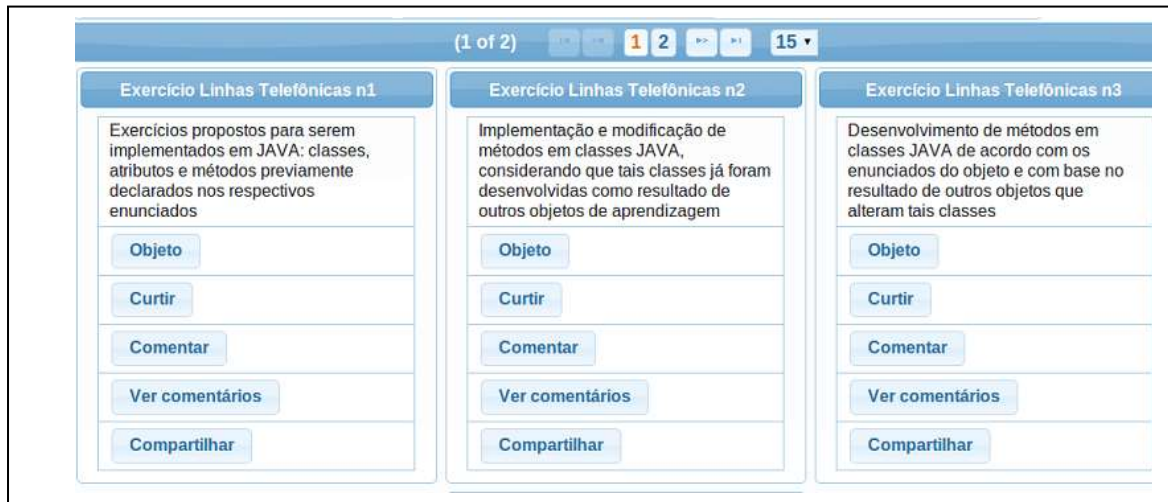
Na figura 22 a tela inicial do IGUALS pode ser visualizada.

Figura 22 - Tela inicial do IGUALS.



Na Figura 23, é mostrada a forma de realizar uma busca pela descrição da atividade de aprendizado, neste exemplo, a linguagem de programação Java.

Figura 23 - Tela de busca de objetos de aprendizagem.



As opções curtir, comentar, ver comentários, compartilhar e objeto são mostrados para cada atividade de aprendizado. Ao clicar em curtir, a opção é lançada no *Facebook*, como mostra a figura 25. Contendo o link para a atividade de aprendizado, disseminando o aplicativo, permitindo que a rede de amigos do usuário possa usar também o aplicativo.

O aplicativo possui um mecanismo de recomendação baseada na opção curtir. Cada usuário recebe recomendações de acordo com as suas preferências informadas pelo aplicativo. Na figura 24 é mostrado um exemplo de recomendação.

Figura 24 - Recomendação de objetos de aprendizagem.

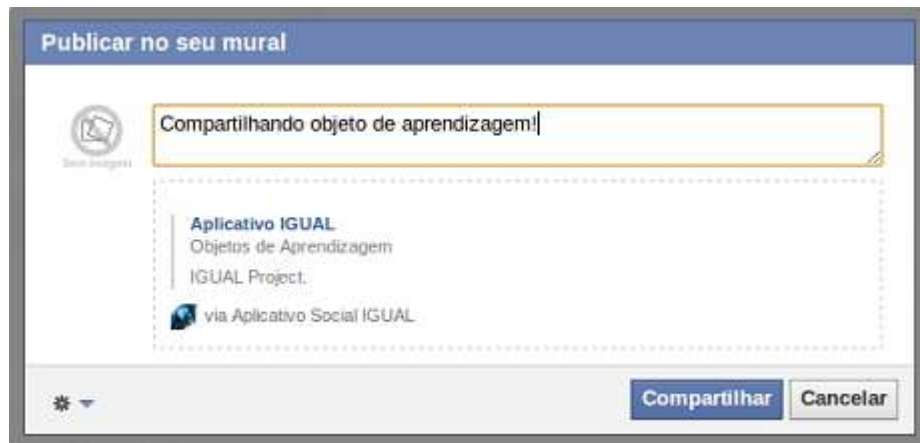


Figura 25 - Opção curtir lançada no Facebook.



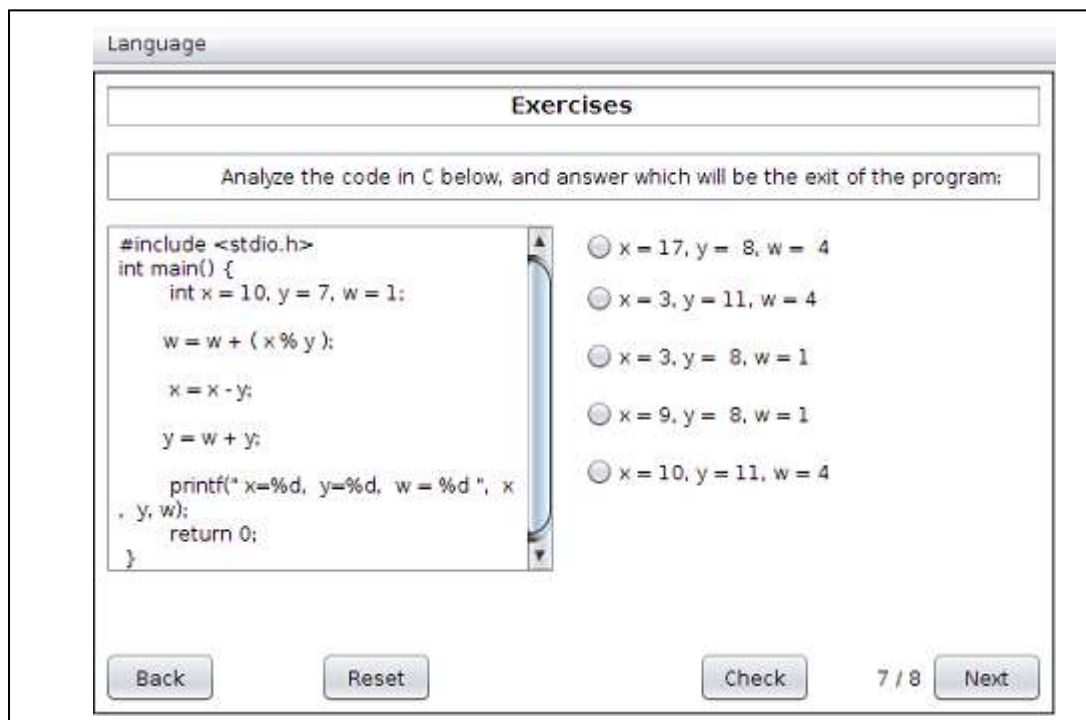
Ao clicar na opção de compartilhar, o usuário pode digitar uma mensagem e compartilhar diretamente no *Facebook*. A Figura 26 demonstra o procedimento.

Figura 26 - Compartilhar objeto de aprendizado.



Há a opção de “Objeto”, link no qual pode ser realizado o download ou visualização do objeto de aprendizagem. Na figura 27 um objeto de aprendizagem aberto pelo IGUALS.

Figura 27 - Exemplo de objeto de aprendizagem.



5.12 Avaliações dos resultados

Tecnicamente, o Aprende é dotado de uma complexidade muito grande. Esta complexidade o torna muito abrangente para a comunidade acadêmica. O sistema contém

um *WEB Service* muito bem definido, documentado e coeso, disponibilizando os dados para utilização dos objetos de aprendizagem.

O *Facebook* permite por meio de diversas linguagens de programação acessar informações dos usuários com restrições. Neste caso, foi utilizado a linguagem de programação *Java* através da biblioteca *RESTFB* e nativamente o *Javascript*. Os dados acessados dos usuários dentro do *IGUALS* estão armazenados em um banco de dados local apenas, não ficando disponível para o *Aprende*. Futuramente, estes dados deverão ser inseridos e depois recuperados por meio do *WEB Service REST* do Projeto.

Com as avaliações “curtir” recolhidas através do *Facebook*, foi possível implementar a recomendação usando o coeficiente *Tanimoto*.

6. CONCLUSÃO E TRABALHOS FUTUROS

Inicialmente o trabalho apresentava algumas limitações em relação ao uso das informações oriundas do *Facebook*. Com pesquisas constatou-se que poderíamos trabalhar com as informações locais sendo recolhidas pela rede.

A integração de aplicativos ao *Facebook* é plenamente viável, no entanto, as configurações exigidas pela plataforma da rede social são pouco documentadas e restritas.

O objetivo de unir o *Aprende* ao *Facebook* dissemina com mais facilidade a ideia geral do projeto e faz com que os objetivos deste sejam alcançados junto a um maior número de pessoas e de forma mais rápida.

O IGUALS mostra os objetos de aprendizagem com a recomendação criada com o *Apache Mahout* e utilizando o coeficiente de *Tanimoto*. Isto permite um uso mais personalizado para os estudantes, incentivando-os a usar. O aplicativo é implementado em *Java*, linguagem de programação amplamente utilizada e orientada a objetos, facilitando a continuação deste aplicativo por programadores.

O IGUALS não armazena os dados no *Aprende*. Isto deve ser implementado futuramente, garantindo inclusive o uso dos dados para outros fins pelo Projeto. O armazenamento destes de forma externa ao *Facebook* demonstra a facilidade de acesso a informações de usuários rede social, bastando a permissão destes.

O aplicativo não contém qualquer tipo de interface gráfica para administração, isto é, os dados não podem ser visualizados em nível de usuário de nenhuma forma. Este módulo deve ser implementado futuramente.

As informações coletadas sobre cada objeto de aprendizagem por meio do *Facebook* ficarão disponíveis, garantindo espaço a outros tipos de recomendação como a *item-based*, identificando as preferências dos estudantes e dando retorno para aumentar a qualidade do trabalho que está sendo feito no projeto IGUAL.

A documentação existente da implementação, além do banco de dados, é da ferramenta *JavaDoc*, responsável por gerar documentação a partir do código fonte de aplicações *Java*.

A implantação do IGUALS será feita na ESPOL (*Escuela Superior Politécnica del Litoral*), com sede em Guayaquil, Equador. Para isso é preciso, de forma remota com o auxílio de analistas da ESPOL, realizar todos os procedimentos necessários de instalação de servidor de aplicação, banco de dados e configurações. Além disso, ainda há uma

alteração a ser feita no IGUALS em razão das versões de bibliotecas que serão utilizadas no servidor da instituição. Como o aplicativo ainda não foi implantado não há justificativa do uso de métricas de avaliação. Futuramente as métricas serão imprescindíveis para avaliar a real relevância das recomendações. Após a implantação, os dados de uso do aplicativo serão recolhidos e poderão ser analisados posteriormente.

REFERÊNCIAS

BATALHA, G.; BOHADANA, E. **O uso do Ila como recurso auxiliar no processo de ensino-aprendizado de algoritmo.** 6º encontro de educação e tecnologia de informação e comunicação., p. 1–13, 2008.

BRAZ, L. M.; SERRÃO, T.; PINTO, S. C. C. S.; CLUNIE, G. **Um Mecanismo para a Integração entre o LMS Moodle e o Site de Redes Sociais Facebook.** , p. 904–913, 2011.

CABRÉ, G. J. **FILTRAGEM COLABORATIVA APLICADA À RECOMENDAÇÃO MUSICAL** Mestrado, 2011. Fundação Edson Queiroz Universidade de Fortaleza - CE.

CAPOBIANCO, L. **Comunicação e Literacia Digital na Internet Estudo Etnográfico e Análise Exploratória de Dados do Programa de Inclusão Digital AcessaSP - PONLINE,** 2010. Universidade de São Paulo - Escola de Comunicação e Artes.

CARACIOLO, M. **Artificial Intelligence in Motion: Evaluating Recommender Systems - Explaining F-Score, Recall and Precision using Real Data Set from Apontador.** Disponível em: <<http://aimotion.blogspot.com.br/2011/05/evaluating-recommender-systems.html>>. Acesso em: 29 abr. 2013.

CECHINEL, C.; CAMARGO, S.; PEREZ, C. C. **Uma Proposta para Localização Facilitada de Objetos de Aprendizagem.** XXII SBIE, p. 331–340, 2011.

DROST, I. **Recommender Documentation.** Disponível em: <<https://cwiki.apache.org/confluence/display/MAHOUT/Recommender+Documentation>>. Acesso em: 14 jun. 2013.

EASYREC. Disponível em: <<http://easyrec.org/recommendation-engine>>. Acesso em: 2 jun. 2013.

CONNECTING to School Communities with Facebook Tips to Help You Connect. Disponível em: <<http://www.thecoudrain.com/files/documents/education-FB-BestPrac-.pdf>>. Acesso em: 10 jun. 2013.

EXTEND Permissions - Desenvolvedores do Facebook. Disponível em: <<http://developers.facebook.com/docs/reference/login/extended-permissions/>>. Acesso em: 25/6/2013a.

OPEN Graph Permissions - Desenvolvedores Facebook. Disponível em: <<http://developers.facebook.com/docs/reference/login/open-graph-permissions/>>. Acesso em: 25 jun 2013.

LIKE - Desenvolvedores do Facebook. Disponível em: <<https://developers.facebook.com/docs/opengraph/guides/og.likes/>>. Acesso em: 6 jun 2013.

FORTUNE, E. Can Facebook Apps Help More Students Get Into College? Disponível em: <<http://www.impatientoptimists.org/Posts/2013/04/HackCollege-20-How-Hacking-Can-Help-Students-Get-a-College-Degree>>. Acesso em: 7 mai. 2013.

GOGOS, R. efront LMS integration with Facebook. Disponível em: <<http://blog.efrontlearning.net/2013/05/efront-lms-integration-with-facebook.html>>. Acesso em: 1 ago. 2013.

HADOOP, A. Welcome to Apache™ Hadoop®! Disponível em: <<http://hadoop.apache.org/>>. Acesso em: 2 mai. 2013.

HO, W. **Integrating social network service into learning management system.** e-Learning Week 2010: Conference, 2010.

IMDB. Disponível em: <<http://www.imdb.com/pressroom/about>>. Acesso em: 2 jun. 2013.

IZZUI. Disponível em: <<http://www.izzui.com/pt/>>. Acesso em: 10 ago. 2013.

KO, M. N.; CHEEK, G. P.; SHEHAB, M.; CAROLINA, N. **Social-Networks Connect Services.** Computing Now, p. 37–43, Aug. 2010.

LAMM, C. Background of collaborative filtering with mahout. Disponível em: <<http://blog.comsysto.com/2013/04/03/background-of-collaborative-filtering-with-mahout/>>. Acesso em: 11 jun. 2013.

MAHOUT. Apache Mahout: Scalable machine learning and data mining. Disponível em: <<https://cwiki.apache.org/confluence/display/MAHOUT/Overview>>. Acesso em: 15/5/2013.

MAZMAN, S. G.; USLU, Y. K. Modeling educational usage of Facebook. **Computers & Education**, v. 55, n. 2, p. 444–453, 2010. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/S0360131510000424>>. Acesso em: 2/3/2013.

MOVIELENS. movielens. Disponível em: <<http://www.movielens.org/html/tour/index.html>>. Acesso em: 2/6/2013.

MUÑOZ, C. L.; TOWNER, T. L. Opening Facebook : How to Use Facebook in the College Classroom. **Society for Information Technology and Teacher Education**, p. 1–13, 2009.

OAuth 2.0. Disponível em: <<http://oauth.net/2/>>. Acesso em: 15 mai. 2013.

OCHOA, X.; SUPERIOR, E.; CARRILLO, G.; ORTEGA, A.; VILLAVICENCIO, C. **Large-scale storage and retrieval of educational metadata using an rdf store.** D-Lib Magazine, 2012.

OWEN, S.; ANIL, R.; DUNNING, T.; FRIEDMAN, E. **Mahout in Action.** 1ed ed. 2011.

PEDAGOGY in Action. Disponível em: <<http://serc.carleton.edu/sp/library/media/why.html>>. Acesso em: 5/5/2013.

QIYUN WANG, HUAY LIT WOO, C. L. Q. **Approaches to Effective Use of Facebook for Teaching and Learning**. ,2011.

RACTHAM, P.; FIRPO, D. **Using Social Networking Technology to Enhance Learning in Higher Education: A Case Study Using Facebook**. 2011 44th Hawaii International Conference on System Sciences. Anais... p.1–10, 2011. IEEE.

RESTFB. RestFB - A Lightweight Java Facebook Graph API and Old REST API Clien. Disponível em: <<http://www.restfb.com/>>. Acesso em: 7 mai. 2013.

REZAC, D. 10 Reasons Facebook Fails Education. Disponível em: <<http://edreach.us/2013/02/24/10-reasons-facebook-fails-education/>>. Acesso em: 7 mai. 2013.

SOCIALBAKERS .Disponível em: <<http://www.socialbakers.com/facebook-statistics/?interval=last-month#chart-intervals>>. Acesso em: 6 mai. 2013.

UOL. Facebook chega a 1 bilhão de usuários com Brasil entre os 5 países mais conectados à rede. Disponível em: <<http://tecnologia.uol.com.br/noticias/redacao/2012/10/04/facebook-chega-a-1-bilhao-de-usuarios-com-brasil-entre-os-5-paises-mais-conectados-a-rede.htm>>. Acesso em: 7 mai. 2013.

WANG, Q.; WOO, H. L.; QUEK, C. L.; YANG, Y.; LIU, M. Using the Facebook group as a learning management system: An exploratory study. **British Journal of Educational Technology**, v. 43, n. 3, p. 428–438, 2012.

WORLD CAT. Disponível em: <<https://apps.facebook.com/worldcat/>>. Acesso em: 7 mai. 2013.

ZACHARSKI, R. **A Programmer's Guide to Data Mining**. ,2012.