CENTRO UNIVERSITÁRIO LUTERANO DE PALMAS
Recredenciado pela Portaria Ministerial nº 1.162, de 13/10/16, D.O.U nº 198, de 14/10/2016
ASSOCIAÇÃO EDUCACIONAL LUTERANA DO BRASIL

Emílio Gomes Santana

CATALOGAÇÃO DE VÍDEOS DO YOUTUBE VOLTADOS PARA A COMPUTAÇÃO

Palmas – TO

2017

Emílio Gomes Santana

CATALOGAÇÃO DE VÍDEOS DO YOUTUBE VOLTADOS PARA A COMPUTAÇÃO

Trabalho de Conclusão de Curso (TCC) II elaborado e apresentado como requisito parcial para obtenção do título de bacharel em Sistemas de Informação pelo Centro Universitário Luterano de Palmas (CEULP/ULBRA).

Orientador: Prof. M.Sc Fernando Luiz de Oliveira.

Palmas – TO

2017

Emílio Gomes Santana

CATALOGAÇÃO DE VÍDEOS DO YOUTUBE VOLTADOS PARA A COMPUTAÇÃO

Trabalho de Conclusão de Curso (TCC) II elaborado e apresentado como requisito parcial para obtenção do título de bacharel em Sistemas de Informação pelo Centro Universitário Luterano de Palmas (CEULP/ULBRA).

Orientador: Prof. M.Sc Fernando Luiz de Oliveira.

Aprovado em: _____/_____/_____

BANCA EXAMINADORA

_____

Fernando Luiz de Oliveira

Orientador

Centro Universitário Luterano de Palmas – CEULP

_____

Prof. M.Sc. Jackson Gomes de Souza

Centro Universitário Luterano de Palmas – CEULP

_____

Prof. M.Sc. Parcilene Fernandes de Brito

Centro Universitário Luterano de Palmas – CEULP

Palmas – TO

2017

RESUMO

SANTANA, Emílio Gomes. **Catalogação de Vídeos do Youtube Voltados para a Computação.** 2017. 33 f. Trabalho de Conclusão de Curso (Graduação) – Curso de Sistemas de Informação, Centro Universitário Luterano de Palmas, Palmas/TO, 2017.

Em virtude do crescimento contínuo da quantidade de informação disponível na *web*, problemas relacionados a organização e a classificação de conteúdo tornam-se constantes. Diante disso, ferramentas que auxiliam na busca e classificação de informação relevante é de fundamental importância. Tendo como foco o conteúdo de vídeos relacionados a computação, o presente trabalho tem por objetivo o desenvolvimento de uma ferramenta que permita a categorização e visualização de vídeos coletados do *Youtube* de acordo com uma taxonomia da computação existente. Para isto, foi definido a taxonomia ACM 2012 da Computação como base hierárquica a ser seguida. Para o desenvolvimento da ferramenta foi utilizada a linguagem de programação PHP e a API de dados do *Youtube,* e para armazenamento o SGBD "MySQL". Para entender o processo de desenvolvimento da ferramenta, este trabalho aborda os conceitos relacionados à Recuperação de Informação, processo ETL e web crawler.

Palavras-chave: Classificação, Recuperação da Informação, *Youtube.*

LISTA DE FIGURAS

LISTA DE ABREVIATURAS

ACM - Association for Computing Machinery

API - Application Programming Interface

DW - Data Warehouse

ETL - Extract, Transform, Load

RI - Recuperação da Informação

SGDB - Sistema de Gerenciamento de Banco de Dados

SQL - Structured Query Language

URL – Uniform Resource Locator

WWW - World Wide Web

SUMÁRIO

1 INTRODUÇÃO

Plataformas de vídeo têm experimentado um crescimento exponencial em termos de conteúdo e interações entre seus usuários. Esta percepção é clara na plataforma *Youtube*[1], que é a rede social de vídeos mais popular hoje no mundo com quase cinco bilhões de vídeos assistidos por dia (STATISTIC BRAIN, 2016, online). Sua popularidade, em grande parte, deve-se à capacidade de integração com outras redes, suporte a vídeos de alta qualidade e disponibilidade de legendas em vários idiomas.

Todo conteúdo publicado no *Youtube* é criado pelos próprios usuários, assim como as informações que identificam cada vídeo em uma busca textual, tais como título, *tag*, descrição. O fato do usuário ter a liberdade de postar quaisquer dados para identificar um vídeo gera resultados inconsistentes na busca por conteúdo.

É importante ressaltar que o *Youtube* organiza os vídeos em categorias como alternativa para otimizar o processo de indexação e busca dos vídeos. Atualmente, existem quinze categorias estabelecidas pelo *Youtube*, são elas: Filmes e desenhos, Automóveis, Musica, Animais, Esportes, Viagens e Eventos, Jogos, Pessoas e Blogs, Comedia, Entretenimento, Notícias e Política, Guias e Estilo, Educação, Ciência e Tecnologia e Sem fins Lucrativos. Porém, este tipo de catalogação não é suficiente porque cada categoria possui inúmeras subdivisões.

Nesse contexto, alternativas que auxiliem o acesso organizado às informações é de suma importância para a indexação e pesquisa, uma vez que podem contribuir para reduzir o esforço do usuário na busca por uma informação relevante ao seu interesse. Para isso, técnicas e métodos de recuperação da informação (RI) são aplicadas para que o resultado apresentado ao usuário seja o mais próximo do desejável. Desta forma, a proposta do presente trabalho é prover uma classificação hierarquizada dos vídeos voltados ao contexto da área da Computação a partir de uma taxonomia da área previamente definida. Para atingir esse propósito, serão utilizadas as *tags* e os títulos associadas aos vídeos para classificar os vídeos de acordo com os elementos definidos na referida taxonomia. Assim, ao final, espera-se que seja concebida uma ferramenta para coleta, categorização e recuperação de vídeos do *Youtube* relacionados a área da computação.

[1] Plataforma Youtube – disponível em: https://www.youtube.com/

2 REFERENCIAL TEÓRICO

Nesta seção serão abordados os conceitos necessários para o desenvolvimento da ferramenta de catalogação de vídeos proposta neste trabalho. Para tanto, é necessário entender sobre o Processo ETL, Recuperação de Informação e Web Crawler.

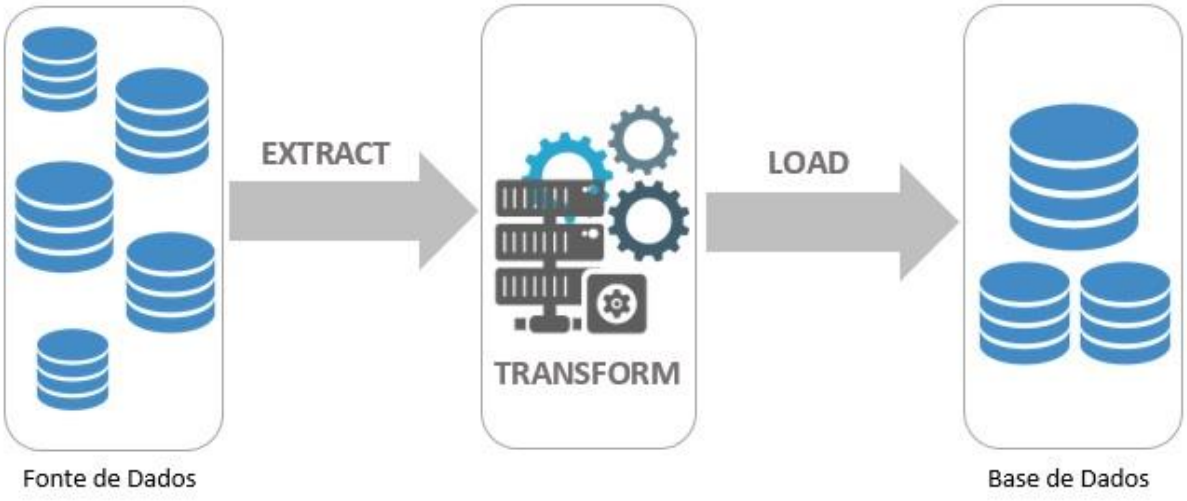
2.1 PROCESSO ETL

O ETL (*Extract Transform Load*) é um processo baseado em ferramentas de software que se destinam a extração, transformação e carga de dados. Estes dados podem ser originados de uma ou mais bases de dados, bem como o destino destes dados podem ser para um ou mais bancos de dados de sistemas de informação ou *data warehouse* (ABREU, 2010).

Processos de ETL estabelecem etapas com o objetivo de consolidar dados de diferentes fontes, sendo que as fontes podem ser de diferentes formatos de arquivos. Existem ferramentas capazes de auxiliar a realização deste processo, ainda assim é uma atividade trabalhosa, complexa e também muito detalhada.

No processo ETL, é imprescindível a extração e carga dos dados, ficando a transformação e a limpeza dos dados de origem opcionais. A decisão de não aplicar a transformação ou limpeza dos dados a serem carregados deve ser adotada somente se os dados de origem estiverem em conformidade com as necessidades da aplicação de destino.

Figura 1 - Processo ETL



Fonte: Elaborada pelo autor

A Figura 1 apresenta as etapas do processo ETL. No primeiro momento, é necessário identificar o tipo, forma de armazenamento, estrutura e modelagem dos dados a serem extraídos. Tal acesso pode ocorrer em diferentes bases de dados e em diferentes tipos de formato de arquivos, dessa forma, gerando a necessidade de distintas formas de extração, resultando em um maior esforço desta etapa.

Segundo KIMBALL (1998 apud ABREU, 2007), somente a extração dos dados leva mais ou menos 60% das horas de desenvolvimento de um DW.

Após a extração, tem-se as informações necessárias para iniciar a etapa de transformação. Nessa fase é realizada a eliminação, o tratamento e a recuperação dos dados que apresentam algum tipo de erro ou inconsistência. Como exemplo, a identificação do sexo de um cadastro de usuário, que pode estar identificado no banco de dados com as letras: “M” ou “F”, para masculino e feminino respectivamente, e já em outro banco de dados estes podem estar representados pelo sentido literal da palavra, através de números ou por qualquer outro tipo de codificação específica do banco de dados de origem. Antes de levar estes dados ao banco de dados de destino, deve-se realizar uma padronização das informações, para garantir a confiabilidade do processo.

A etapa de carga ocorre em sequência com a de transformação. Assim que são efetuados todos os tratamentos necessários para que as informações estejam de acordo com os requisitos do banco de destino, a carga dos dados se inicia de fato. Dependendo da modelagem do sistema, a forma como será realizado o processo poderá variar. Esta etapa pode ser executada em uma única vez ou de forma periódica para atualização dos dados. Algumas informações podem ser substituídas por informações cumulativas ou novas informações podem ser adicionadas frequentemente, por exemplo, diariamente, semanalmente ou mensalmente.

Além disso, é importante verificar a integridade dos dados, no qual é preciso certificar se os campos que são chaves estrangeiras com suas respectivas tabelas estão de acordo com a chave primaria da outra tabela.

Durante o processo de carga, técnicas para otimizar o processo podem ser utilizadas, tais como evitar a geração de *log* durante o processo, criar índices e agregar dados. Muitas destas técnicas podem ser aplicadas aos bancos de dados via *scripts* ou através de ferramentas de organização de dados.

Com isso, o conjunto de informações armazenadas, é possível a visualização dos dados através de ferramentas, que podem submeter a análise qualitativa e quantitativa de padrões dos dados.

2.2 RECUPERAÇÃO DE INFORMAÇÃO

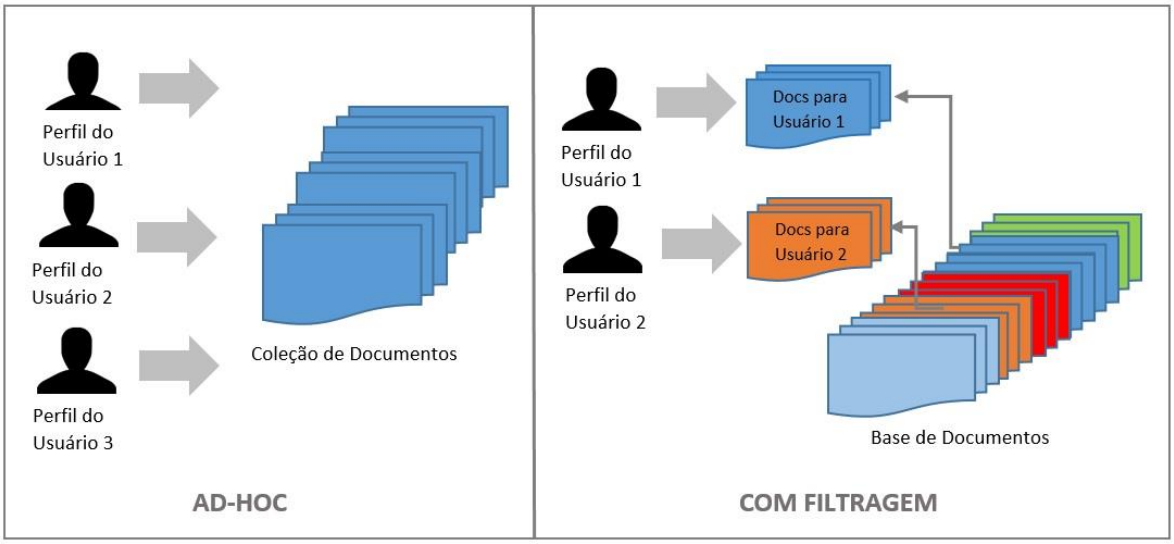
De acordo com Gonzalez & Lima (2001, p. 2), “a essência da Recuperação da Informação (RI) consiste na busca de documentos relevantes a uma dada consulta que expressa a necessidade de informação do usuário”. Neste sentido, o processo de recuperação de informação vai além de uma simples pesquisa de documentos, espera-se que os sistemas de recuperação de informação sejam flexíveis o suficiente, de tal maneira que o usuário possa adaptar o processo de busca de informação à sua necessidade.

Desta maneira a RI possui grande importância para a área da computação, pois não consiste apenas em técnicas e métodos que envolvem armazenamento e algoritmos de recuperação, mas também em adaptar os sistemas no comportamento do usuário, entendendo desta maneira, como é a construção da informação e das instruções para a recuperação (CONEGLIAN & FUSCO, 2015).

Em geral, o conteúdo recuperado é apresentado em forma de texto, como por exemplo, documentos diversos e páginas *web*, embora possa ser constituído de outros tipos, tais como imagens, áudios e gráficos. Anteriormente o conteúdo é organizado e indexado, o que torna o acesso à informação mais eficiente.

A RI pode ser de dois tipos: *ad-hoc* ou com filtragem (BAEZA-YATES, 1999). Na recuperação *ad-hoc* é executado um conjunto de novas requisições sobre uma coleção de documentos pré-estabelecidos, ou seja, é executado diferentes consultas sob uma mesma base de dados de possíveis documentos relevantes. Este tipo é o mais comum, o resultado das consultas é o mesmo, independente do usuário. Para a recuperação com filtragem, é necessário definir se um novo documento é relevante ao usuário. Assim, pode-se determinar quais documentos são relevantes e determinar a necessidade informacional do usuário. A Figura 2 apresenta ambas as possibilidades:

Figura 2 - Tipos de recuperação de informação: ad-hoc e com filtragem

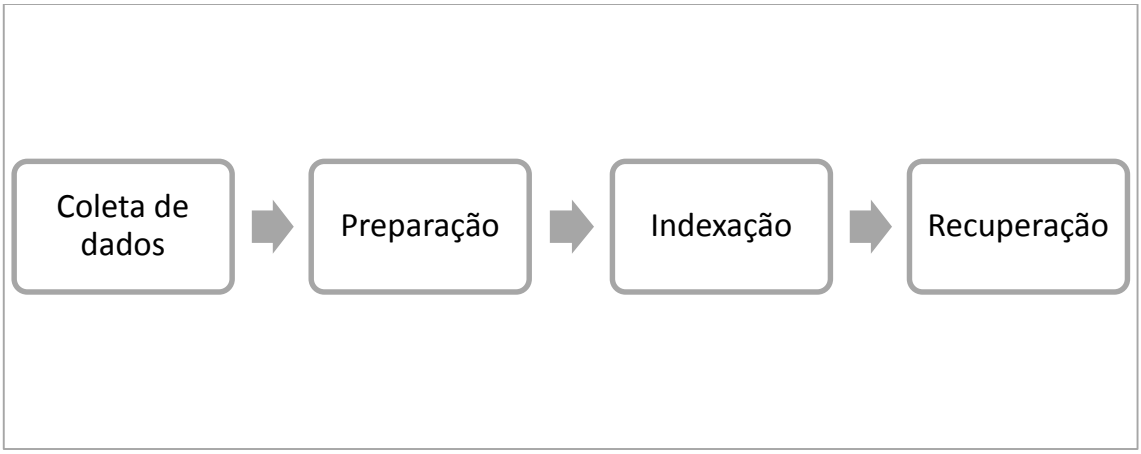


Fonte: SWEETS: um Sistema de Recomendação de Especialistas aplicado a Redes Sociais (SILVA, 2009, p. 24)

Como exemplo de recuperação do tipo *ad-hoc*, pode-se citar a função *trend topics* do *twitter*. Esta funcionalidade do *twitter* retorna uma lista de *hashtags* mais discutidas daquele momento. O *trend topics* do *twitter* é formado com base no conteúdo da própria plataforma. Neste contexto, os usuários recebem a mesma lista de *hashtags* recuperada de uma mesma coleção de documentos, consolidando assim um tipo *ad-hoc*. Seguindo esta ideia, é possível o usuário personalizar o *trend topics* de sua conta do *twitter*, além do conteúdo geral que é gerado para todos usuários, ele pode definir os resultados do *trend topics*, determinando o resultado por região. Essa função de filtrar o conteúdo relevante para um determinado usuário é atribuída ao tipo de recuperação de informação com filtragem, conforme o contexto da Figura 2.

Com base nos conhecimentos de sistema de Recuperação da Informação, estes podem seguir uma estrutura de etapas bem definidas, são elas:

Figura 3 - Processo de recuperação de informação



Fonte: Desenvolvimento de um aplicativo de recomendação de artigos científicos para materiais didáticos (JESUS, 2009)

- **Coleta de dados:** a coleta pode ser realizada em uma fonte especifica ou em diversas fontes. Em ambos os casos há a necessidade dos dados serem acessados e submetidos a um tratamento posterior. Além disto, esta coleta pode ocorrer de forma manual, quando se faz o acesso direto no local onde os dados estão armazenados ou coleta automática do conteúdo de páginas com a utilização de *crawlers*.
- **Preparação:** Na preparação os dados são transformados, ou seja, é executado uma limpeza de todo conteúdo indesejado. Rezende (2003, p. 315), "técnicas de limpeza devem ser aplicadas aos dados a fim de garantir sua qualidade". Isto é necessário para que a filtragem da informação ocorra de forma mais precisa, visto que uma parte do conteúdo de uma mensagem pode ser dispensável no momento da busca.
- **Indexação:** Para Vieira e Corrêa (2010, p. 03), a etapa de indexação é responsável pela "construção da representação do conteúdo dos documentos através da atribuição de termos (palavras-chave) ou códigos de indexação que serão úteis posteriormente na recuperação desses documentos". Esses termos são responsáveis por referenciar documentos, funcionando como índices que auxiliar.
- **Recuperação:** Após a indexação, os dados são armazenados em bancos de dados para que os usuários possam recuperar a informação. Esta etapa consiste na utilização de técnicas que permitam o usuário obter um resultado mais aproximado do desejado. De acordo com Ferneda (2003, p.18), "a eficiência de

> um sistema de RI está diretamente ligada ao modelo que o mesmo utiliza". Assim, deve-se definir um modelo que garanta a eficiência de todo o processo.

Para que estas etapas tenham desempenho satisfatório, faz-se necessária a adoção de modelos específicos de recuperação da informação Assim, de acordo com Aires (2005), um modelo de RI apresenta e explica o que um usuário irá considerar relevante dada sua consulta. São três modelos clássicos seguidos por sistemas de RI para determinar a relevância de documentos: Booleano (Lógico), Vetorial e Probabilístico.

O modelo booleano é baseado na teoria dos conjuntos e álgebra booleana. As consultas são especificadas como expressões booleanas (BAEZA-YATES & RIBEIRO-NETO, 1999, p.25). Portanto, essas buscas podem ser formuladas por meio de operadores lógicos *or*, *and* e *not* para especificar os documentos a serem recuperados, baseados nas restrições logicas da expressão de busca. Seu método de recuperação é baseado em um sistema binário, ou seja, os documentos são analisados e considerados relevantes ou não relevantes sem nenhuma outra forma de ordenação.

O modelo espaço vetorial ou simplesmente vetorial utiliza o conceito de vetores para representar documentos (Rezende, 2003). Neste modelo, cada documento é representado por um vetor de termos e cada termo possui uma relevância, pode-se calcular o grau de similaridade e definir um *ranking* que indica o grau de importância do termo no documento. Para determinar a similaridade são utilizadas algumas medidas de similaridades, dentre elas que as palavras apareçam ao menos duas vezes e determinar o grau de similaridade, com vistas a construir um ranking (Souza, 2006).

O modelo probabilístico considera pesos binários que representam a presença ou ausência de termos em documentos (Cardoso, 2004). O vetor gerado é o resultado do cálculo da probabilidade de que um documento é relevante para uma consulta. Este, é criado através de sucessivas iterações com o usuário, busca-se aproximar cada vez mais de um conjunto de documentos ideal, por meio da análise dos documentos considerados pertinentes ao usuário.

As etapas de um processo de RI são essenciais para um melhor desempenho de busca elaborada pelo usuário, já que envolve desde mecanismos de coleta de dados, até a etapa final que é a recuperação (busca) destes dados. Em ambientes web, o mecanismo de coleta de dados pode ser realizado através de um *web crawler*. - assunto abordado na Seção 2.3.

2.3 WEB CRAWLER

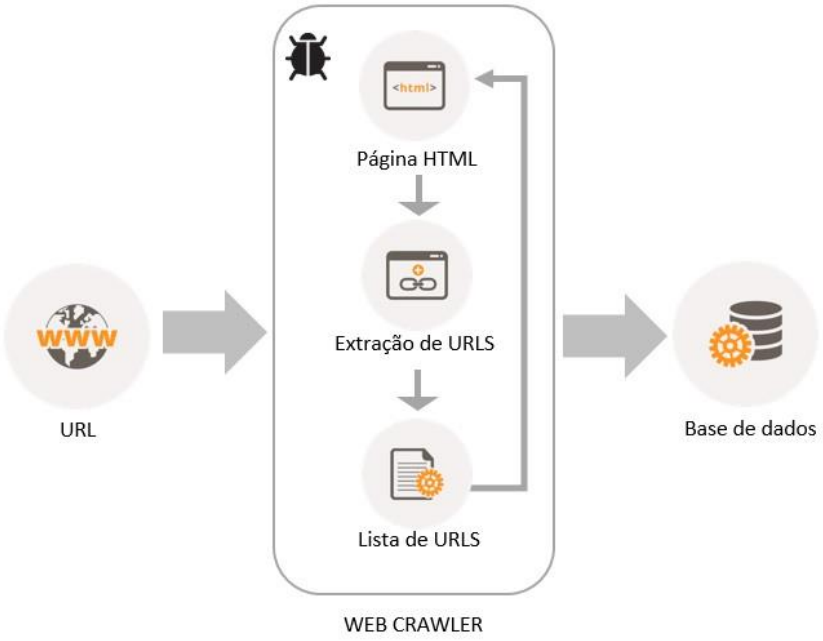
Um *web crawler* é uma ferramenta que automatiza a tarefa de navegar entre páginas da web, com a finalidade de indexar informações relevantes ao conteúdo pesquisado

(PINKERTON, 2009). Esta ferramenta faz uma varredura sistemática na internet e compacta o conteúdo original, além de arquivar *urls*, que possibilitam a descoberta de novas páginas. O objetivo de um *crawler* eficiente é coletar o máximo de páginas úteis no menor tempo possível (MANNING, RAGHAVAN e SCHÜTZE, 2009, p. 443).

Muitos sites, em particular os motores de busca, usam *crawlers* para manter uma base de dados atualizada. O *crawler* armazena uma cópia de todas as páginas visitadas para que o motor de busca possa indexar estas páginas e prover buscas mais eficientes. Além disto, pode ser utilizado para localizar informações especificas, como encontrar endereços de e-mails e depois serem usados em listas de *spam*. Ainda mais, é capaz executar tarefas automatizadas de manutenção em web sites, por exemplo, verificar *links* e validar um código *html*.

Um *web crawler* inicia o seu processo com uma ou mais *urls*, que constituem um conjunto de sementes (MANNING, RAGHAVAN e SCHÜTZE, 2009, p. 444). A medida que o *crawler* visita as *urls*, ele identifica todos os *hiperlinks* na página e armazena em uma lista de *links* a serem visitadas. Neste caso, Figura 2.

Figura 4 - Processo de captura de um Web Crawler



Conforme apresentado na Figura 4, o *web crawler* inicia seu processo a partir de um conjunto de endereços da web, no próximo passo, ele acessa as páginas web e executa uma varredura no código *html*, em que identifica o conteúdo desejado e extrai *links* de novas páginas. Depois, os *links* extraídos são salvos em uma lista de *urls*. Esta lista pode ser atualizada muitas vezes, até que se chegue no objetivo, isto é, ter coletado o conjunto de

dados esperado para o dado contexto. Por fim, as *urls* são armazenadas em um banco de dados.

Como exemplo de aplicação que faz uso de *web crawlers*, pode-se citar os *search engine* ou motores de busca. Os motores de busca para a internet são programas que, informadas determinadas palavras-chave ou expressões, desenvolvem uma lista de hiper-ligações para documentos onde essas palavras existem. (Pereira,2004).

3 MATERIAIS E MÉTODOS

Nesta seção serão apresentados os materiais e métodos utilizados para desenvolver o presente trabalho que, juntamente com as orientações, permitiram a conclusão do mesmo.

3.1 POPULAÇÃO E AMOSTRA

Para este trabalho considerou-se como população a totalidade de vídeos do *Youtube* coletados a partir da busca pelo termo "*computing*". Desta população foram recuperadas informações como as *urls* dos vídeos e palavras chave ou *tags*. A pesquisa foi realizada na língua inglesa, pois o objetivo era utilizar parte da taxonomia ACM 2012 da computação[2] para a catalogação destes vídeos.

3.2 MATERIAIS

Para a coleta de vídeos foi utilizado o módulo de extração de dados de vídeos do *Youtube* desenvolvido no trabalho de Estágio Supervisionado de Santana (2017), que faz parte da pesquisa deste trabalho.

No desenvolvimento da ferramenta web foi utilizada a API de dados do *Youtube*, a qual permite a incorporação de funções normalmente executadas no site do *Youtube* em uma aplicação baseada na web (GOOGLE DEVELOPERS, 2016). Para utilizá-la é preciso ter uma autorização, concebida através de uma chave de API, que pode ser obtida no site: <https://console.developers.google.com/apis/>.

Para o desenvolvimento do código fonte da ferramenta web de catalogação de vídeos foi utilizado a linguagem de programação PHP, que é uma linguagem *open source* de conteúdo dinâmico e aplicada na *World Wide Web* (WWW). Em conjunto foi utilizado o framework *Bootstrap*, que é um framework *front-end*, ou seja, usado para criar projetos *web* responsivos utilizando o *html* e *css*.

Como ambiente de desenvolvimento foi utilizado o *PhpStorm* (JETBRAINS, 2000). Esta ferramenta foi escolhida por ser gratuita, de fácil aprendizado e possuir recursos que facilitam a edição do código fonte.

Como forma de armazenamento dos dados foi utilizado o sistema de gerenciamento de banco de dados (SGBD) MySQL.

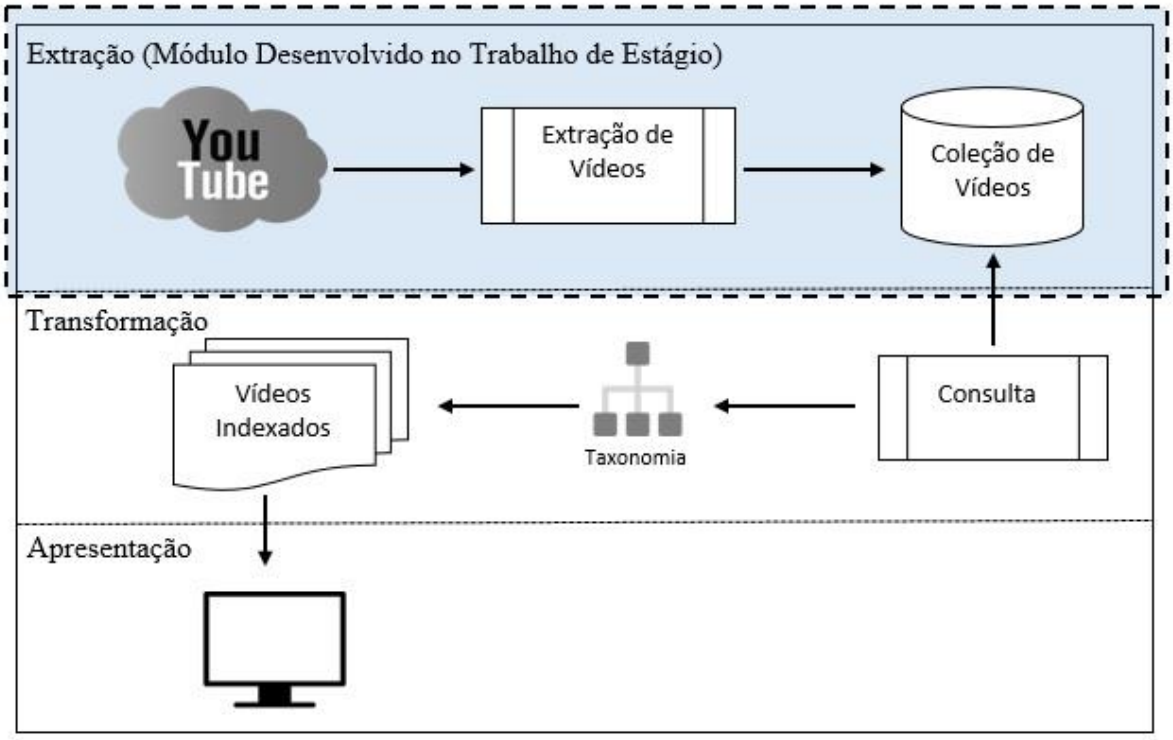
[2] https://www.acm.org/publications/class-2012

3.3 PROCEDIMENTOS

Para o entendimento conceitual, necessário para o desenvolvimento do trabalho, foi realizado um levantamento bibliográfico, que buscou abordar e esclarecer o processo ETL (*Extract Transform Load*), a Recuperação da Informação (RI) e descrever a estrutura e uso de *web crawlers*. Este estudo possibilitou uma maior compreensão das etapas existentes na recuperação de informação e quais processos e ferramentas foram utilizadas de fato.

O desenvolvimento do trabalho contempla as seguintes etapas conforme apresentadas na Figura 5.

Figura 5 - Etapas do projeto



Conforme apresentado na Figura 5, o processo está divido em três etapas. A primeira etapa do processo (Extração), foi realizada pelo módulo de extração de dados de vídeos do *Youtube* relacionados a área da computação. Conforme já mencionando, este módulo foi desenvolvido no Estágio Supervisionado de Santana (2017), e foi adicionado como parte do desenvolvimento da ferramenta deste trabalho.

A segunda etapa corresponde a transformação. Inicialmente, foi realizada uma consulta aos dados extraídos na etapa anterior. A partir da consulta à coleção de vídeos, foi executado o processo de indexação. Nesta etapa, os dados serão indexados conforme a definição hierárquica da taxonomia ACM da computação. Esta indexação permitirá a identificação dos vídeos e sua recuperação de acordo com sua posição hierárquica da

taxonomia. Utilizou-se uma pequena parte da taxonomia ACM 2012 da área da computação, por se tratar de uma taxonomia extensa, com 84 categorias e mais de 6.000 (seis mil) palavras devidamente hierarquizadas.

Após a indexação, o próximo passo (terceira etapa) é a apresentação da catalogação dos vídeos em uma página web. Para apresentar a catalogação dos vídeos, é utilizada a própria hierarquia da taxonomia em forma de árvore, sendo que, cada elemento da taxonomia representa um nó.

Apresentados os materiais utilizados bem como a metodologia adotada neste trabalho, a seção a seguir descreverá os resultados obtidos.

4 RESULTADOS E DISCUSSÃO

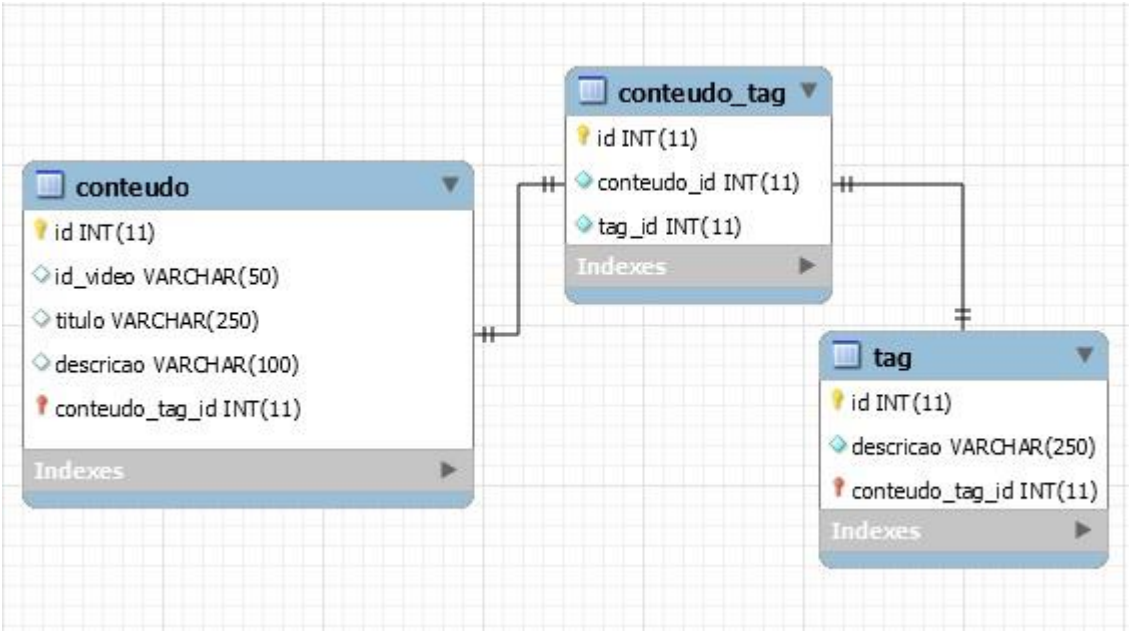
Nesta seção são apresentados os resultados obtidos no desenvolvimento da ferramenta proposta. Deste modo, a divisão deste capítulo segue as três etapas de divisão do desenvolvimento do trabalho, cada qual apresentando seus resultados específicos e discussões. Na seção a seguir é apresentada a extração dos vídeos, na 4.2, a transformação dos vídeos coletados, e na seção 4.3 a etapa de apresentação dos vídeos.

4.1 EXTRAÇÃO

A primeira etapa para realização do trabalho foi a coleta dos dados a serem classificados. Foi definido a extração dos títulos e *tags* dos vídeos existentes no *Youtube*. Para isto utilizou-se o processo *Extract, Transform and Load* (ETL), que permitiu a extração de 984 títulos de vídeos e 1037 *tags*.

Esta etapa em específico foi realizada através do módulo de extração de dados do *Youtube* desenvolvido no trabalho de estágio supervisionado, do próprio autor deste trabalho. Este módulo permitiu pesquisar e extrair dados de vídeos relacionados a computação. Para isto, a palavra "*computing*" foi utilizada como termo de pesquisa para retornar vídeos referente a computação. Para facilitar a análise, os dados dos vídeos foram armazenados em um banco de dados MySql. A Figura 6 apresenta a estrutura do banco de dados do módulo de extração.

Figura 6 - Estrutura do banco de dados do módulo de extração



A estrutura do banco de dados (Figura 6) é formada pela tabela conteúdo que armazena o *id* do vídeo, responsável pela identificação do vídeo no *Youtube*, além de armazenar o título e a descrição de cada vídeo. A tabela *tag* é responsável por armazenar as *tags* extraídas. Ainda foi criada a tabela de ligação *conteudo_tag*, que possui duas chaves estrangeiras. Este tipo de associação entre as tabelas otimiza o desempenho da consulta ao banco de dados e evita duplicidade de dados. Na sequência, será apresentada a etapa de transformação, que é a parte central deste trabalho.

4.2 TRANSFORMAÇÃO

O objetivo principal da transformação é permitir a classificação dos vídeos, através da indexação. Para isto, utilizou-se uma parte da taxonomia ACM *Computing Classification System* 2012. Nas próximas subseções são apresentados os processos envolvidos nesta etapa.

4.2.1 TAXONOMIA

A taxonomia escolhida para ser utilizada no trabalho é a ACM *Computing Classification System* 2012, criada por uma comunidade científica e educacional dedicada a computação, que permanece em constante revisão desde sua criação em 1982. Assim, foi definida a utilização de uma parte da referida taxonomia, no caso a parte referente a *computer system organization* que possui 4 níveis e 59 nós.

Para facilitar a associação, foi criada uma tabela no banco de dados com o intuito de armazenar os elementos da taxonomia. Esta tabela tem como objetivo facilitar a busca dos elementos da taxonomia e principalmente permitir a associação das palavras da taxonomia e os dados dos vídeos armazenados no banco.

Na figura 7, a seguir, é apresentado como os elementos da taxonomia foram inseridos na tabela do banco de dados.

Figura 7 - Tabela taxonomia

id	idPai	descricao
1	NULL	Architectures
2	1	Serial architectures
3	2	Reduced instruction set computing
4	2	Complex instruction set computing
5	2	Superscalar architectures
6	2	Pipeline computing
7	2	Stack machines
8	1	Parallel architectures
9	8	Very long instruction word
10	8	Interconnection architectures
11	8	Multiple instruction, multiple data
12	8	Cellular architectures
13	8	Multiple instruction, single data
14	8	Single instruction, multiple data
15	8	Systolic arrays
16	8	Multicore architectures
17	1	Distributed architectures
18	17	Cloud computing

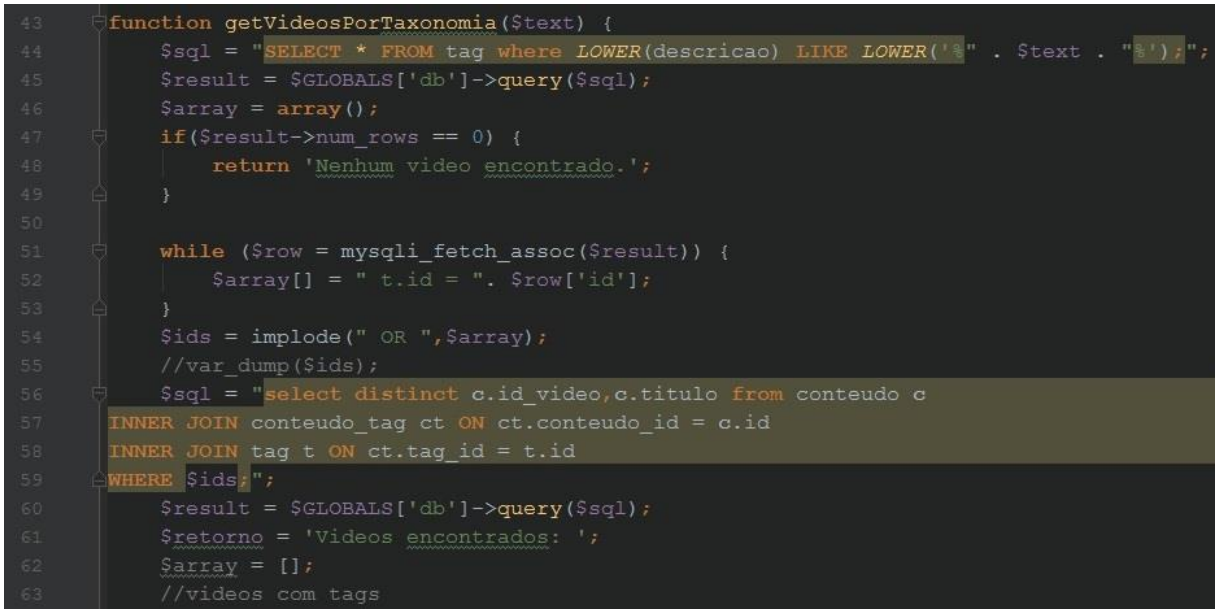
Na Figura 7 são apresentados os elementos inseridos na tabela taxonomia do banco de dados. Nesta tabela foi criado uma coluna idPai, para que fosse possível criar uma hierarquia de nós de forma que a ordem da taxonomia permanecesse. Assim, foi criada a relação entre um nó folha (último nó da hierarquia) e o nó raiz (nó inicial da hierarquia).

Logo depois, os elementos da taxonomia são utilizados no processo de indexação, este processo é apresentado na seção a seguir.

4.2.1 INDEXAÇÃO

O objetivo desta etapa é fazer a relação entre os dados extraídos do módulo de extração e os elementos da taxonomia, permitindo a identificação dos vídeos de acordo com a taxonomia. Para realização desta etapa utilizou-se uma consulta SQL, na qual foi criada uma função de busca, apresentada na imagem a seguir:

Figura 8 - Classificação de vídeos por tag

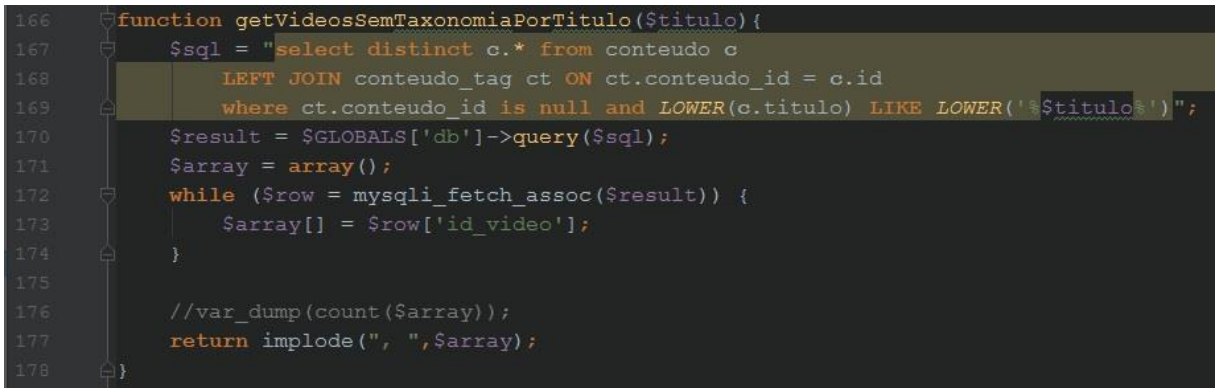


```
43  function getVideosPorTaxonomia($text) {
44      $sql = "SELECT * FROM tag where LOWER(descricao) LIKE LOWER('%" . $text . "%');";
45      $result = $GLOBALS['db']->query($sql);
46      $array = array();
47      if($result->num_rows == 0) {
48          return 'Nenhum video encontrado.';
49      }
50
51      while ($row = mysqli_fetch_assoc($result)) {
52          $array[] = " t.id = ". $row['id'];
53      }
54      $ids = implode(" OR ",$array);
55      //var_dump($ids);
56      $sql = "select distinct c.id_video,c.titulo from conteudo c
57  INNER JOIN conteudo_tag ct ON ct.conteudo_id = c.id
58  INNER JOIN tag t ON ct.tag_id = t.id
59  WHERE $ids;";
60      $result = $GLOBALS['db']->query($sql);
61      $retorno = 'Videos encontrados: ';
62      $array = [];
63      //videos com tags
```

Na Figura 8 é apresentada a função de busca por vídeos que pertencem a uma taxonomia através da comparação entre as palavras da taxonomia e as *tags* dos vídeos. Na linha 44 do código, uma consulta SQL faz a busca e a comparação entre as *tags* e os elementos da taxonomia informado através da variável *$text.*

Entretanto, alguns vídeos não possuem *tag*, neste caso uma outra função foi criada para ser feito a comparação e indexação pelo título do vídeo. A Figura 9 apresenta a função responsável pela busca dos vídeos relacionados com a taxonomia através do título.

Figura 9 - Classificação de vídeos pelo título



```
166 function getVideosSemTaxonomiaPorTitulo($titulo){
167     $sql = "select distinct c.* from conteudo c
168         LEFT JOIN conteudo_tag ct ON ct.conteudo_id = c.id
169         where ct.conteudo_id is null and LOWER(c.titulo) LIKE LOWER('%$titulo%')";
170     $result = $GLOBALS['db']->query($sql);
171     $array = array();
172     while ($row = mysqli_fetch_assoc($result)) {
173         $array[] = $row['id_video'];
174     }
175
176     //var_dump(count($array));
177     return implode(", ",$array);
178 }
```

Na Figura 9 é apresentada a função de busca por vídeos que pertencem a uma taxonomia através da comparação entre as palavras da taxonomia e o título dos vídeos. Na linha 167 do código, uma consulta SQL faz a busca pelos vídeos que não possuem *tags* através a identificação destes vídeos é feito a associação dos elementos da taxonomia e o título dos

vídeos. Nesta etapa, apenas os vídeos sem *tags* foram utilizados na comparação, já que os vídeos com *tags* foram catalogados previamente.

Com a conclusão do processo de indexação, é feita a apresentação dos dados em uma página web. A etapa de apresentação dos dados é descrita na próxima seção.

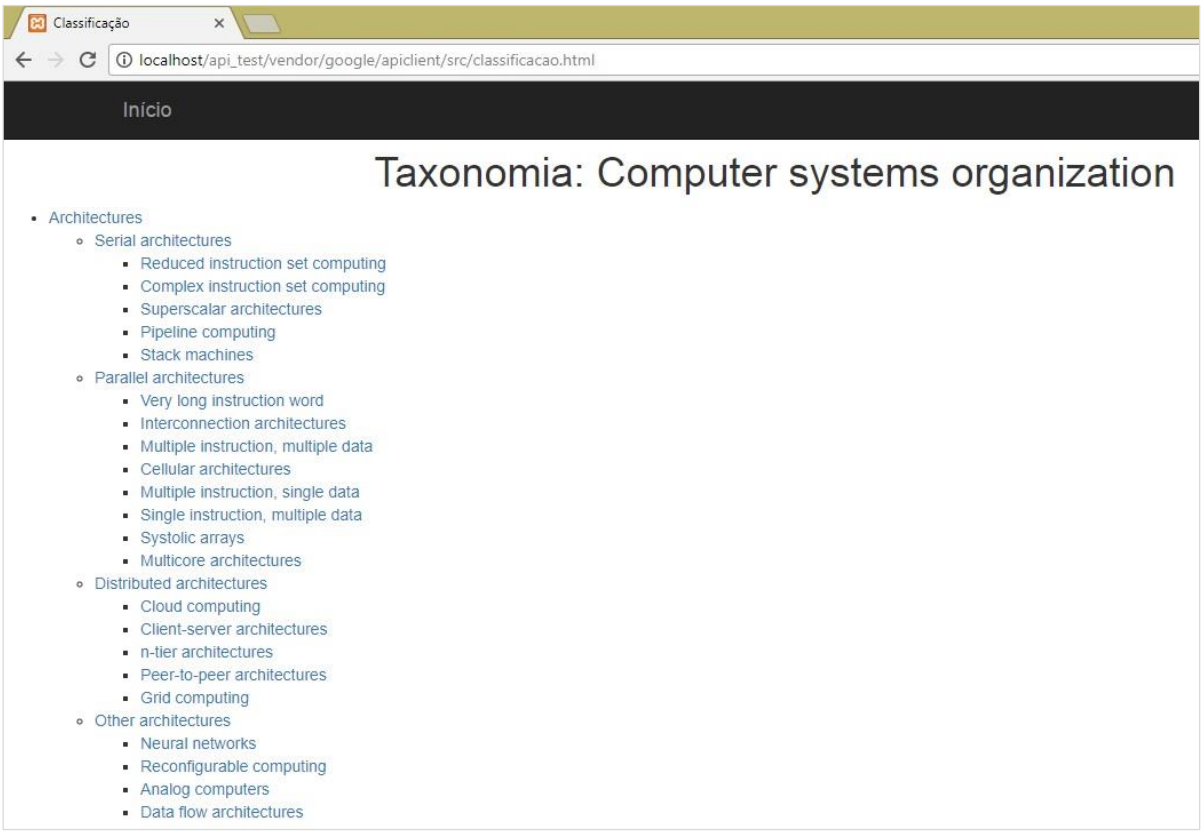
4.3 APRESENTAÇÃO

A etapa de apresentação tem o objetivo de listar os vídeos classificados e permitir que o usuário selecione os vídeos. Para isto, foi criada a estrutura de interface para apresentar os vídeos classificados na taxonomia. Esta estrutura foi criada a partir de um *plugin* jquery. Para tanto, foi utilizado o plugin jsTree, sua estrutura é apresentado na figura a seguir:

Figura 10 - Estrutura do plugin jsTree

Na Figura 10 é apresentado um exemplo da estrutura do plugin jsTree. Este plugin utiliza jQuery e foi customizado de acordo com o modelo hierárquico da taxonomia aplicada. O resultado de sua utilização é apresentado na figura a seguir:

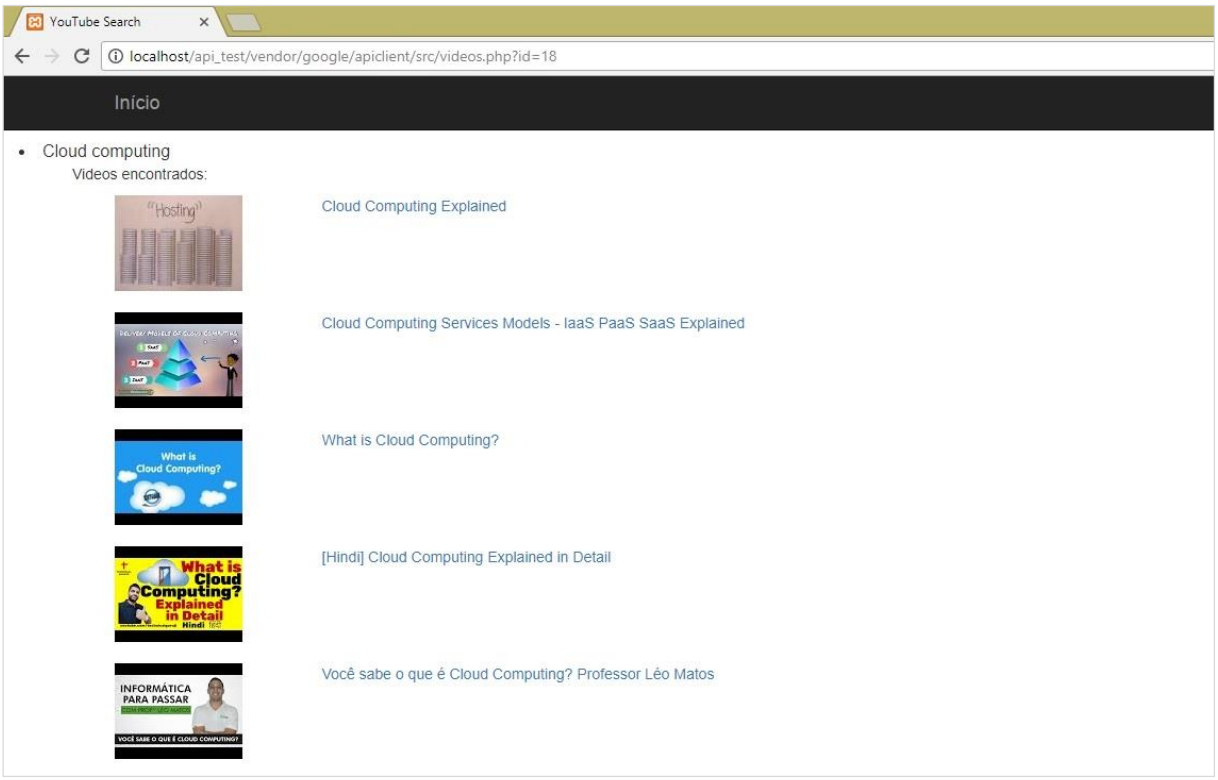
Figura 11 - Tela inicial da ferramenta



Na Figura 11 é apresentada a tela inicial da ferramenta de classificação. Quando o usuário selecionar o elemento da taxonomia, serão apresentados os vídeos referentes a sua taxonomia em uma nova janela.

Para demonstrar o processo de apresentação dos vídeos classificados será utilizado o exemplo a seguir:

Figura 12 - Tela de vídeos classificados



A Figura 12 apresenta o resultado da seleção do elemento *Distributed architectures*. Ao clicar sobre um dos vídeos listados, o vídeo é apresentado em uma nova janela diretamente no site do *Youtube*.

Esta seção abordou os resultados obtidos durante o desenvolvimento da ferramenta de catalogação de vídeos do *Youtube*. O desenvolvimento incluiu a apresentação do módulo de extração de dados, a indexação dos vídeos com a taxonomia e o desenvolvimento da apresentação hierárquica da catalogação dos vídeos. Foi apresentado também um exemplo de visualização gerada pela ferramenta.

5 CONSIDERAÇÕES FINAIS

No decorrer deste trabalho foram realizados estudos sobre Recuperação da Informação, Processo ETL e Web Crawler. A partir destes conceitos foi elaborado o referencial teórico no qual permitiu que fosse entendido os fundamentos para o desenvolvimento da ferramenta de catalogação de vídeos do *Youtube* relacionados à computação. Foram ainda apresentados a taxonomia ACM 2012 da computação e o *plugin* jsTree, aplicados no desenvolvimento da ferramenta.

Para atingir o objetivo do trabalho, foi utilizado uma base de dados de vídeos relacionados a computação. Para executar esta tarefa de extração, utilizou-se o módulo de extração de dados do *Youtube* desenvolvido pelo próprio autor deste trabalho durante a disciplina de Estágio Supervisionado. Com o módulo foi possível extrair os *ids*, títulos e *tags* de vídeos voltados para computação.

Durante o desenvolvimento deste trabalho, observou-se que o site do *Youtube* alterou sua apresentação dos resultados. A API de dados do *Youtube* utilizada no módulo de extração, passou a retornar os vídeos pesquisados em um formato de paginação do tipo *page token*, onde é gerado um código para cada página de vídeos retornados. Este tipo de paginação evita o envio desnecessário de dados ao cliente, consequentemente o número de vídeos coletados diminuiu comparado aos testes iniciais.

Constatou-se que alguns vídeos não possuíam *tags* ou não condiziam com o conteúdo do vídeo. Isto ocorre pelo fato de não ser obrigatório o preenchimento desta informação no momento da publicação do vídeo. Na intenção de melhorar precisão da classificação dos vídeos, os elementos da taxonomia foram comparados inicialmente pelo título, caso alguma palavra do título não pertencesse a uma taxonomia, a comparação foi feita com as *tags*.

Deste modo, a ferramenta desenvolvida, permitiu a organização e apresentação e dos vídeos de forma mais relevante. Entretanto, muitos vídeos não foram classificados. Assim, outro trabalho futuro poderia utilizar a taxonomia por completo, tornando maior o número de equivalências de termos entre a taxonomia e a base de dados.

Para trabalhos futuros, propõe-se fazer a tradução da taxonomia para a língua portuguesa, já que a taxonomia original da ACM é disponibilizada na língua inglesa, assim, o trabalho pode contribuir melhor com a comunidade acadêmica. Também, apresentar uma solução de ranqueamento, que apresente os resultados mais relevantes aos usuários da ferramenta.

6 REFERÊNCIAS

STATISTIC BRAIN. **Youtube Company Statistics**. Disponível em: <http://www.statisticbrain.com/youtube-statistics/>. Acessado em: 12 dez. 2016.

ABREU, Fábio Silva Gomes da Gama e. **Desmistificando o conceito de ETL**. Disponível em: < http://www.fsma.edu.br/si/Artigos/V2_Artigo1.pdf >. Acesso em: 17 out. 2016.

KIMBALL, Ralph. **Data Warehouse ETL Toolkit**. Indianapolis, Crosspoint Boulevard, 2004.

Gonzalez, Marco; Lima, Vera L. S. de. Recuperação de Informação e Expansão Automática de Consulta com Thesaurus: uma avaliação. **XXVII Conferencia Latinoamericana de Informática (CLEI'2001)**. Mérida, Venezuela, 2001. CD-ROM, ISBN 980-110527.

CONEGLIAN, Caio Saraiva; FUSCO, Elvis. Agente semântico de recuperação da informação aplicado a extração de artigos científicos**. Regrad Univem**, v. 8, n. 1, 2015. Disponível em: < file:///D:/Downloads/784-1-2533-1-10-20151123%20(1).pdf> Acesso em: 28 set. 2016.

Baeza-Yates, R., Ribeiro-Neto, B., Modern Information Retrieval, ACM Press, New York, USA, 1999.

Ferneda, E. **Recuperação da Informação: Análise sobre a contribuição da Ciência da Computação para a Ciência da Informação**. 2003. 147 f. Tese (Doutorado em Ciência da Informação) – Escola de Comunicação e Artes, Universidade de São Paulo, São Paulo.

SILVA, Edeilson Milhomen da. **SWEETS: um Sistema de Recomendação de Especialistas aplicado a Redes Sociais**. 2009.

JESUS, D. N. **Desenvolvimento de um aplicativo de recomendação de artigos científicos para materiais didáticos.** 2009.

VIEIRA, Jessica Monique de Lira; CORRÊA, Renato Fernandes. Recuperação De Informação Através De Recursos Visuais. In: Encontro Nacional De Estudantes De Biblioteconomia, Documentação, Gestão, E Ciência Da Informação, 33, 2010, João Pessoa, PB. **Anais**... João Pessoa: ENEBD, 2010. Disponível em: < http://dci.ccsa.ufpb.br/enebd/index.php/enebd/article/viewFile/19/22>. Acesso em: 28 set. 2016.

AIRES, Rachel V.X. **Uso de marcadores estilísticos para a busca na Web em português**. 2005. 202 p. Tese (Doutorado em Ciências de Computação e Matemática Computacional) – USP, São Carlos.

SOUZA, Renato Rocha. **Sistemas de recuperação de informações e mecanismos de busca na web: panorama atual e tendências**. 2006. Disponível em:

<portaldeperiodicos.eci.ufmg.br/index.php/pci/article/download/320/940>. Acesso em 10 nov. 2016.

CARDOSO, O. N. P. **Recuperação de Informação**. Disponível em: <www.dcc.ufla.br/infocomp/index.php/INFOCOMP/article/download/46/31>. Acesso em 10 nov. 2016.

PINKERTON, Brian. **WebCrawler: Finding What People Want**. Disponível em: < http://www.thinkpink.com/bp/Thesis/Thesis.pdf >. Acesso em: 01 nov. 2016.

Rezende, S. O., Sistemas Inteligentes: fundamentos e aplicações. Barueri, SP: Manole, 2003.

MANNING, C. D.; RAGHAVAN, P.; SCHÜTZE, H. **An Introduction to Information Retrieval**. England: Cambridge University Press Cambridge, 2009. 544p. Disponível em: <http://nlp.stanford.edu/IR-book/pdf/irbookprint.pdf>. Acesso em: 03 set. 2016.

PEREIRA, Vasco N. S. **Arquitectura de um motor de busca: exemplo do Google**. Disponível em: < https://eden.dei.uc.pt/~vasco/Papers_files/Google_v1.pdf >. Acesso em: 9 nov. 2016.

ACM. **The 2012 ACM Computing Classification System**. Disponível em: <https://www.acm.org/publications/class-2012>. Acesso em: 1 set 2016.

GOOGLE. **YouTube Data API**. 2016. Disponível em: < https://developers.google.com/youtube/> Acesso em: 28 set. 2016.

PHP. **php**. 2016. Disponível em: <http://php.net>. Acesso em: 28 set. 2016.

JETBRAINS. **PhpStorm IDE**. 2016. Disponível em: <https://www.jetbrains.com/phpstorm/>. Acesso em: 28 set. 2016.

ORACLE. **MySQL**. 2016. Disponível em: <http://www.oracle.com/br>. Acesso em: 28 set. 2016.

SANTANA, Emilio Gomes. **Módulo de extração de dados de vídeos relacionados a área da computação**. CEULP/ULBRA, 2017.