

Universidade Federal do Rio Grande do Sul - UFRGS

Instituto de Informática da UFRGS

Classificação e Pesquisa de Dados – INF 01124 – Turma A

RELATÓRIO DO TRABALHO FINAL DA DISCIPLINA

Ana Ribeiro

Fábio Petkowicz

Pablo Martins da Silva

Porto Alegre, 30 de junho de 2013.

Relatório do Trabalho Final de CPD

1. Identificação do trabalho e dos alunos

Extrator de currículos XML. Aplicativo em janela simples, desenvolvido através do GUI-Builder do NetBeans. Retorna às informações, propostas na definição do trabalho final 2013-1, a partir da seleção do currículo desejado.

Alunos:

Ana Paula Ribeiro 174045, Fábio Petkowicz 118627, Pablo Martins da Silva 213989.

2. Descrição do problema e da solução proposta

O problema proposto tem como principais pontos serem superados:

- Abrir os currículos em XML.
- Selecionar os conteúdos de interesse: (1) Nome do pesquisador e (2) Publicações (em Conferências e em Periódicos), contido em cada um dos currículos.
- Extrair esses dados dos currículos salvando em algum tipo de estrutura externa.
- Abrir os documentos CSV (*Comma-separated values*), `periodicos.csv` e `conferencias.csv`, para extrair a classificação QUALIS referente as Publicações encontradas nos currículos em XML.
- Relacionar ou agregar os QUALIS encontrados com as Publicações dos Pesquisadores aplicando e/ou relacionando os conceitos trabalhados em aula.
- Criar uma Interface Gráfica de Usuário (GUI) amigável.

Desta forma foram questões norteadoras iniciais: (1) Como realizar a leitura e extração de dados de um arquivo XML?; (2) Como realizar a leitura e extração de dados de um arquivo CSV?; (3) Como criar uma GUI amigável para o aplicativo proposto?; (4) Como aplicar as estruturas avançadas trabalhadas em aula?

Soluções:

- Qualquer documento XML pode ser representado como uma árvore de análise sintática (parse-tree). Portanto, se utilizarmos um analisador sintático (parser), podemos acessar os dados que nos interessam. Java disponibiliza vários *parsers* para XML. Todos eles seguem duas especificações (1) *Simple Access to XML* (SAX) e (2) *Document Object Model* (DOM). Os que seguem a especificação DOM nos permitem percorrer os nodos da árvore. Para saber quais nodos precisamos acessar temos dois recursos: (1) a Definição de Tipo de Documento (DTD) e (2) Abrir o Documento em um Editor de Planilhas, o primeiro é mais elegante, porém menos prático. A DTD, do currículo lattes pode ser encontrada na plataforma de mesmo nome. Java oferece muitas APIs para o trabalho com XML. Uma das mais conhecidas é a Java API for XML Processing (JAXP). Usamos a interface *Document* presente na DOM. Em geral mais fácil de utilizar, mas pouco prático. Exige muitos laços para percorrer os nodos da árvore. Ela pertence ao pacote *org.w3c.dom*.

Após obter todas as informações de cada documento XML. Convertemos as informações em uma lista de objetos pesquisadores.

- A leitura e extração dos dados em CSV utilizando Java foi muito simples. Exigiu apenas uma estrutura de controle tipo laço para leitura das linhas e método *split* da classe String. Um fragmentador de strings. O *split* recebe uma string separadora. Neste caso o ponto e vírgula uma vez que o documento foi gerado em uma planilha em língua portuguesa (números de ponto flutuante são separados por vírgula pelos falantes deste idioma). O método retorna um array de string.
- A criação da GUI exigiu bastante preparação. Nenhum integrante conhecia o GUI-Builder do NetBeans, nem qualquer outro GUI-Builder.
- A classe ArrayList, nos pareceu uma solução razoável para o tipo de aplicativo, uma vez que ela faz a realocação de memória, em caso de necessidade. Sabemos que isso é feito de forma bem custosa, em termos de tempo, pois faz o transporte de uma área grande de memória. Mas ao tipo de aplicativo que chegamos ela resolve o problema de maneira satisfatória.

3. Descrição da contribuição do trabalho

Foram desenvolvidos códigos para a extração dos dados solicitados. Esses códigos envolvem laços para percorrer a árvore de análise do XML.

Elaboração de novas estruturas de dados para o armazenamento em memória das informações coletadas.

- Classe Pesquisador;
- Classe Publicacao;

4. Manual do usuário da aplicação

Para rodar o aplicativo é necessário ter a Java Virtual Machine instalada no computador. Além disso é importante que a extensão .JAR, não esteja associada a outro software, isso pode acarretar em não funcionamento do buscador.

Dar duplo clique no executável. Selecionar o currículo no formato XML, através da barra de navegação.

Clicar no botão: Buscar XML.

5. Considerações Finais

O grupo procurou aplicar o princípio básico de dividir para conquistar a partir dos pontos detalhados no item 2 deste relatório. Entretanto, possuía uma expectativa inicial muito grande para programadores sem experiência em Java. Desta forma o tempo despendido no aprendizado da foi demasiado. Sendo necessárias adequações não só no cronograma de trabalho, mas também nas metas.

Embora tenhamos optado por uma estrutura pronta foi evidente que as técnicas e estruturas estudadas na disciplina facilitaram na compreensão geral do trabalho e consequentemente a busca de alternativas para os problemas enfrentados.

6. Código-Fonte: Em documento com extensão .JAVA.