

Sistema de controle de vendas com Web Scraper

Thyall D'greville Santiago de Azevedo, Tiago Onofre Araújo

Instituto Metr pole Digital - Universidade Federal do Rio Grande do Norte (UFRN).

Disciplina: Linguagem de programa  o II

Docente: Jo o Carlos Xavier Junior

Abstract. *This article aims to develop and use a tool capable of extracting data and information from sales sites. For this purpose it was necessary to use the JAVA language and the IDE Eclipse development environment, with that. The graphic interface tool adopted was SWING with an MDI (multiple document interface) model. After passing the product to the program the user can choose to receive a file containing the 10 most expensive values, the 10 cheapest and the average price of the desired product.*

Resumo. *Este artigo tem como objetivo o desenvolvimento e o uso de uma ferramenta capaz de realizar a extra  o de dados e informa  es de sites de vendas. Para tal feito foi necess rio o uso da linguagem JAVA e o ambiente de desenvolvimento da IDE Eclipse, com isso . A ferramenta de interface gr fica adotada foi do SWING com um modelo de MDI (multiple document interface). Ap s passar o produto para o programa o usu rio poder  optar por receber um arquivo contendo os valores mais caros, os mais baratos e a m dia de pre o do produto desejado.*

1. Introdu  o

O problema a ser tratado   a varia  o nos pre os dos produtos que consumidores est o enfrentando nos  ltimos meses. Devido ao cen rio atual de pandemia os produtos v m apresentando uma mudan a muito r pida e abrupta nos seus valores principalmente os produtos eletr nicos. Obter um registro e ter a facilidade de pesquisar os valores do produto desejado ir  auxiliar o usu rio na praticidade e na an lise da compra.

2. Metodologia de Desenvolvimento

2.1 Descrição das etapas do projeto

Primeiro, em equipe, foi levantada uma discussão de qual seria o tema do projeto. Após a escolha do tema foi feito um levantamento de formas de abordar o tema (*Web Scraper*). Com a forma de abordagem decidida(listagem dos preços de algum produto) foi dividido entre os integrantes suas demandas e tarefas: Criação e integralização da interface, por Thyall, e o desenvolvimento do backend, por Tiago, sendo essa a busca, coleta e listagem dos produtos.

2.2 Descrição das classes

- **TelaPrincipal** → Classe responsável pela criação e gerenciamento da tela principal do projeto.
- **TelaProduto** → Classe responsável pela criação e gerenciamento da tela do produto, assim como nela é passado os valores para a requisição do produto e métodos de pesquisa.
- **TelaSobre** → Classe responsável pela criação e gerenciamento da tela ajuda, nela há um breve descrição sobre a forma de interação com a interface.
- **Banco** → Classe responsável por criar e armazenar uma lista de produtos, assim como nessa classe estão os métodos que chamam a forma de pesquisa e ordenação dos produtos.
- **ScraperController** → SuperClasse responsável por estruturar as futuras classes, visto que, pela diferença nas necessidades de parsing das respostas, um processador genérico não seria eficaz.
- **ScraperAmazonController** → Classe derivada que ficaria encarregada de parsear a resposta da amazon.
- **ScraperKalungaController** → Classe derivada que ficou encarregada de parsear a resposta da Kalunga.
- **Eletronic** → Classe derivada da interface produto, adaptada para acomodar produtos referentes a categoria de eletrônicos.
- **IProductInterface** → Esta classe é o que permite a ampliação do projetos para os variados tipos de produtos, mantendo um padrão para as características gerais presentes em um item comercializável(Nome, preço, local de venda).

3. Dificuldades Encontradas

As principais dificuldades encontradas se deram em alguns sites, pois enquanto uns usavam requisições, outros usavam páginas estáticas, dificultando na busca do produto. Outro problema apresentado foi como fazer o *Parse* nos valores do produto, pois como não havia um padrão, foi necessário um tratamento maior com a *String* de valor, gerando respostas muito complexas para alguns sites (Amazon). A ideia inicial do projeto era poder requisitar de 2 ou mais sites para que o usuário pudesse comparar os preços, devido a tais dificuldades optamos por trabalhar apenas com 1 site o da Kalunga. Por fim a maior dificuldade foi necessário separar os valores de taxa, vista etc. Por isso foi utilizado só busca pela tag “a vista”, com isso o poder de busca de produtos foi reduzido por nem todos produtos apresentarem essa tag.

4. Conclusão

Durante a execução e finalização do projeto concluímos que projetos com o uso de *Web Scraper* tem um grande leque de possibilidades para desenvolvimento de softwares, em contrapartida, a grande variedade nos meios de comunicação de um site atrelada a falta de experiência na área inviabilizaram uma busca e coleta otimizadas, fazendo com que o projeto tenha ficado limitado.

Referências

GeeksForGeeks - Java Guide

<https://www.geeksforgeeks.org/java/?ref=shm>

Wikipedia- Coleta de dados na web

https://pt.wikipedia.org/wiki/Coleta_de_dados_web

[Sahin, K.] (2017) "Introduction to Web scraping with Java"

<https://medium.com/hackernoon/introduction-to-web-scraping-with-java-3d8a8d0f250b>