

TRABALHO DE PARADIGMAS PROLOG

Alunos: Rafael Augusto de Rezende Neto
Wayne Nascimento Souza

Introdução

Este trabalho tem como objetivo apresentar uma breve pesquisa sobre paradigmas de programação, além da aplicação e o desenvolvimento de um sistema com uso de dois paradigmas de programação distintos, sendo eles, paradigma lógico e orientado a objetos. Este trabalho se justifica como uma atividade prática da disciplina de Paradigmas de Linguagens, ofertada pelo professor Bruno Ferreira, no curso de Ciência da Computação. Por fim, utilizando as linguagens de programação Prolog e Java, além dos frameworks Spring e Angular, e bibliotecas de integração entre as linguagens, pode-se obter como resultado, um sistema que utilize o melhor de dois paradigmas de programação diferentes.

Objetivos

Objetivo geral

O trabalho consiste em usar dois paradigmas de programação diferentes. O paradigma orientado a objetos será usado para criar interfaces, manipular planilhas e arquivos de texto, além de fornecer uma base de dados formatada. O paradigma lógico será usado para buscar informações em uma base de conhecimento.

Objetivos específicos

1. Manipular planilhas (XLSX) para extrair o conteúdo e armazenar em um arquivo de texto.
2. Construir uma interface que proporcione formas para que o usuário possa fazer perguntas a uma determinada base de conhecimento.

Materiais

1. Spring Framework (Java)
2. Angular (HTML, CSS e Typescript)
3. SWI Prolog
4. Sistema Operacional (Windows/Linux)
5. IntelliJ IDEA
6. Github

Spring (Java)

De acordo com Andrade (2021), o Spring é um framework Java criado com o objetivo de facilitar o desenvolvimento de aplicações, explorando, para isso, os conceitos de Inversão de Controle e Injeção de Dependências. Dessa forma, ao adotá-lo, temos à nossa disposição uma tecnologia que nos fornece não apenas recursos necessários à grande parte das aplicações, como módulos para persistência de dados, integração, segurança, testes, desenvolvimento web, como também um conceito a seguir que nos permite criar soluções menos acopladas, mais coesas e, conseqüentemente, mais fáceis de compreender e manter.

Angular (HTML, CSS e Typescript)

Segundo Branas (2014), o AngularJS foi criado por Miško Hevery e Adam Abron em 2009, sendo este um framework JavaScript de código aberto que é executado do lado do cliente (client-side), e que auxilia na produtividade do desenvolvimento Web. Angular (2020) descreve o Angular como um framework de código aberto para criação de SPAs, criado e mantido pelo Google. Além disso, o Angular explora aspectos do desenvolvimento backend no frontend, e que a biblioteca foi construída com o design pattern MVC, o que torna os projetos construídos com angular extensíveis, de fácil manutenção, testáveis e padronizados.

SWI Prolog

Segundo a própria documentação do SWI Prolog, pode ser considerada uma implementação versátil da linguagem Prolog. Embora tenha ganhado popularidade principalmente na educação, seu desenvolvimento é impulsionado pelas necessidades de desenvolvimento de aplicativos. Isso é facilitado por uma interface rica para outros componentes de TI, suportando muitos tipos de documentos e protocolos (de rede), bem como uma interface abrangente de baixo nível para C, que é a base para interfaces de alto nível, como Java e Python.

Sistema Operacional

Segundo Tanenbaum (2003), os sistemas operacionais realizam duas funções

essencialmente não relacionadas: fornecer a programadores de aplicativos (e programas aplicativos, claro) um conjunto de recursos abstratos limpo em vez de recursos confusos de hardware, e gerenciar esses recursos de hardware. Microsoft Windows é uma família de sistemas operacionais desenvolvidos, comercializados e vendidos pela Microsoft, já o Linux é um sistema operacional, assim como o Windows e o Mac OS, que possibilita a execução de programas em um computador e outros dispositivos. Linux pode ser livremente modificado e distribuído (SIMIONI, 2021).

IntelliJ

O IntelliJ IDEA é um ambiente de desenvolvimento integrado escrito em Java para o desenvolvimento de software de computador. Ele é desenvolvido pela JetBrains e está disponível como uma edição da comunidade licenciada do Apache 2 e em uma edição comercial proprietária. O IntelliJ IDEA foi projetado para maximizar a produtividade do desenvolvedor. Juntos, a assistência para codificação inteligente e o design ergonômico tornam o desenvolvimento não apenas produtivo, mas também agradável.

Github

De acordo com o LONGEN (2022), o GitHub é um serviço baseado em nuvem que hospeda um sistema de controle de versão chamado Git. Ele permite que os desenvolvedores colaborem e façam mudanças em projetos compartilhados enquanto mantêm um registro detalhado do seu progresso.

Métodos

Para que o objetivo do trabalho fosse realizado, primeiramente, foram feitas algumas pesquisas com o intuito de encontrar uma biblioteca que fizesse a conexão do Prolog com o Java. Após as pesquisas, ficou decidido que a biblioteca utilizada seria a JPL, que de acordo sua documentação, é um conjunto de classes Java e funções C que fornecem uma interface bidirecional entre Java e Prolog, porém, a versão utilizada do JPL só funciona com o SWI Prolog.

Diante disso, uma codificação básica para entender e aprender as funções do JPL foi iniciada. Logo após ter adquirido alguma experiência, foi iniciada a

implementação do objetivo do trabalho utilizando o framework Spring, por ambos os integrantes do grupo já possuírem experiência e todo o versionamento de código foi feito através do Github.

Foram desenvolvidos dois *endpoints* (URL onde seu serviço pode ser acessado por uma aplicação cliente) na parte de *backend* da aplicação, um para criar a base de conhecimento e o outro para realizar as pesquisas. Para construir um primeiro *endpoint* foi necessário utilizar a biblioteca Apache POI para manipular uma planilha, que segundo o Baeldung (2022) é uma biblioteca que pode ser usada para ler, escrever e modificar dinamicamente o conteúdo de uma planilha e fornecer uma maneira eficaz de integrar o Microsoft Excel em uma aplicação Java, com isso, conseguimos extrair as informações da planilha e ajustar para o formato correto armazenamento um arquivo Prolog como uma base de conhecimento. O segundo endpoint consiste em pegar a informação digitada pelo usuário e a regra correspondente e através do JPL realizar a consulta no arquivo Prolog que já foi previamente preenchido com o conteúdo da planilha, por fim, retornando o resultado da consulta para o usuário.

Para a interação do usuário, foi desenvolvido uma aplicação utilizando o framework Angular, por apresentar uma boa produtividade no desenvolvimento, além de fornecer interfaces com usuário agradáveis e uma boa experiência de uso. A escolha foi baseada na experiência de ambos integrantes do grupo com o framework. Dessa forma, foi criada uma camada de serviço, responsável por realizar as requisições ao servidor, um componente que é responsável por montar a interface e carregar as informações recebidas na tela de forma dinâmica.

Figura 1 - Interface de Usuário

Trabalho Prático - Paradigmas

Alunos: Rafael Neto e Wayne Souza

Paradigma Lógico

Caminho do arquivo XLSX

Preencher

Verifique se um produto qualquer foi comprado

Verificar

Verifique qual foi a quantidade total comprada de um produto

Verificar

Verifique qual foi o valor total comprada de um produto

Verificar

Verifique o que foi comprado na data

Verificar

Verifique qual o total de compras em uma loja

Verificar

Veja qual foi o produto mais comprado:

_1

Depois de tudo estar pronto, tanto o *backend* quanto o *frontend*, elaboramos as regras para que fosse possível responder as perguntas presentes no enunciado do trabalho.

Resultados e discussão

Foram obtidos bons resultados com a resolução do trabalho, visto que as três primeiras perguntas e a quinta pergunta do enunciado foram concluídas sem nenhum impedimento ou inconsistência. Já na quarta pergunta, percebemos alguns comportamentos distintos, provavelmente relacionados à questão do tipo *Date* (data) do próprio Excel. Em alguns casos que produtos foram comprados na mesma data, o retorno da consulta está correto e em outros apenas um produto é retornado, essa inconsistência não atrapalha no funcionamento da aplicação, porém, a resposta daquela determinada pergunta pode estar incompleta.

Um ponto negativo encontrado durante a construção do projeto foi que, mesmo usando Maven para a injeção e gerência de dependências, ainda foi necessário adicionar um arquivo `.jar` do JPL nas configurações de bibliotecas externas.

Conclusão

Neste trabalho foi apresentado e discutido brevemente sistemas com uso de dois paradigmas de programação distintos, sendo eles, paradigma lógico e orientado a objetos. Segundo as pesquisas realizadas e citadas acima, as tecnologias que mais foram adequadas para realizar tal integração, foi a utilização da linguagem Prolog, para o desenvolvimento dos fatos e regras lógicas, e a linguagem Java, com framework Spring, para o desenvolvimento das API orientadas a objetos. Por fim, o trabalho se mostra útil na aplicação a qual se destinou, pois foi possível responder a todas as perguntas propostas nas especificações, porém, foi encontrado dificuldades no desenvolvimento das regras que respondem a última pergunta. Mesmo assim, os resultados foram favoráveis, e demonstraram ser fiéis ao esperado, sendo apresentados em uma interface de usuário amigável.

Referências bibliográficas:

SIMIONI, Dionatan. Entenda como funciona o Linux. **Hostinger**, 2021. Disponível em: <<https://www.hostgator.com.br/blog/entenda-como-funciona-o-linux/>>. Acesso em: 14 de maio de 2022.

ANDRADE, Ana Paula de. O que é Spring?. **TreinaWeb**, 2021. Disponível em: <<https://www.treinaweb.com.br/blog/o-que-e-o-spring>>. Acesso em: 14 de maio de 2022.

SWI PROLOG. SWI Prologs features. Página inicial. Disponível em: <<https://www.swi-prolog.org/features.html>>. Acesso em: 14 de maio de 2022.

BAELDUNG. Working with Microsoft Excel in Java, 2022. Disponível em: <<https://www.baeldung.com/java-microsoft-excel>>. Acesso em: 15 de maio de 2022.



LONGEN, Andrei. O que é Github e como usá-lo. **Hostinger**, 2022. Disponível em: <<https://www.hostinger.com.br/tutoriais/o-que-github>>. Acesso em: 15 de maio de 2022.

TANENBAUM, A. S., **Sistemas Operacionais Modernos**. Segunda Edição, Prentice Hall, 2003. Bibliografias Complementares. GALVIN, S., Operating System Concepts.

BRANAS, Rodrigo. AngularJS essentials: **Design and construct reusable, maintainable, and modular web applications with AngularJS**. Birmingham, United Kingdom: Packt Publishing, 2014.