



Relatório - 2º Trabalho Prático

Universidade de Évora

Sistemas Distribuídos 2022/2023

Bernardo Vitorino (148463) e Daniel Barreiros (148452)

I – Introdução

O relatório que se segue é alusivo ao 2º trabalho pratico da disciplina de sistemas distribuídos, lecionado pelo docente José Saias e tem como objetivo transmitir a nossa opinião critica sobre a realização deste trabalho.

De uma forma geral, foi implementado um sistema de arquitetura distribuída com abstração dos detalhes de comunicação sendo um serviço RESTful com uso do framework Spring Boot

Para realizar a interação com a base de dados foi usada a API JpaRepository.

II – Desenvolvimento

No desenvolvimento deste trabalho foram criados dois módulos, um servidor e um cliente. O módulo servidor contém todas as classes e interfaces que estabelecem o serviço. No modulo cliente existem duas classes, ClienteGeral e ClienteGestao, que efetuarão os pedidos HTTP ao servidor.

No módulo servidor criamos uma classe Ad.java que tem como anotação @Entity fazendo desta classe uma entidade da base de dados criada mais tarde. A classe Ad tem descritas todas as variáveis que fazem parte de um anúncio no contexto do trabalho, sendo estas o estado, localização, preço, género, data, anunciante, tipologia e tipo. Além destes atributos a classe Ad tem também uma lista de strings que irá guardar as mensagens. Esta lista tem a anotação @ElementCollection que fará com que seja criada uma outra tabela na base de dados para fazer a representação das mensagens.

No servidor está também presente uma interface AdRepository que estende JpaRepository. Nesta interface estão presentes os métodos que poderão ser chamados sobre a base de dados. Com o uso da interface JpaRepository não tivemos que desenvolver as queries para interagir com a base de dados. A ligação à base de dados também não foi desenvolvida de uma forma explicita visto que através de alguns parâmetros no application.properties o framework estabelece a ligação com a base de dados.

A classe `AdController` pertencente também ao módulo servidor com a anotação `@RestController` para que o framework spring assuma que esta classe é o controlador do serviço. No controlador é feita a associação de cada método com um URL, para que no cliente se consigam fazer pedidos HTTP através dos mesmos. A interação com a base de dados é feita através destes métodos.

No módulo cliente as classes nele presentes servem de interface de comunicação com o servidor, evitando a necessidade da utilização de um programa mais poderoso como um browser. Em ambas as classes são executados pedidos HTTP através de métodos disponíveis para a linguagem Java.

III - Considerações

No decorrer da concretização deste trabalho confrontámo-nos com algumas dificuldades, uma delas foi o uso das dependências maven que muitas vezes geravam conflitos.

O framework spring também causou algumas dificuldades dado que eram gerados vários erros quando tentávamos iniciar o servidor.

Devido às dificuldades encontradas e a alguma falta de tempo não conseguimos implementar a parte correspondente à valorização superior.

Durante o desenvolvimento deste trabalho, em maior parte das ocasiões que nos confrontamos com duvidas conseguimos solucioná-las através da observação das atividades realizadas na aula e da API Spring.

IV - Conclusão

Aprofundamos todos os temas deste trabalho de um modo que o pudéssemos enriquecer e de maneira a que ficássemos a compreender melhor a matéria lecionada nesta disciplina.

Do nosso ponto de vista, e embora não tenhamos conseguidos realizar todos os pontos, consideramos que o balanço foi positivo visto que, embora não tenhamos realizado a parte correspondente à valorização superior, conseguimos desenvolver a aplicação cliente – servidor como pedido no enunciado.