

Representação e Processamento de Conhecimento na Web

Repositório de Recursos Didáticos

Relatório de Desenvolvimento

Gustavo Araújo Lourenço
(pg47229)

Martim Bento da Rocha de Almeida
(pg47514)

19 de junho de 2022

Conteúdo

1	Introdução	2
2	Arquitetura	3
3	Funcionalidades	4
3.1	Utilizadores	4
3.2	Autenticação	4
3.2.1	Nível de acesso no token	4
3.3	BagIt	5
3.4	Recursos	5
3.5	Notícias/Logs	6
4	Extra	7
5	Conclusão e Trabalho Futuro	8

Capítulo 1

Introdução

No âmbito da UC Representação e Processamento de Conhecimento na Web, foi proposto o desenvolvimento de uma aplicação Web cujo serviço consiste num repositório de recursos didáticos.

A aplicação em si terá as funcionalidades usuais de uma aplicação web:

- Autenticação de Utilizadores;
- *Upload*, *Download* e edição de recursos, sendo possível que os recursos tenham visibilidade pública ou privada;
- Realizar comentários nos recursos presentes no sistema;

Para isso, a solução representada deste projeto, que será apresentada neste relatório é desenvolvida através das várias tecnologias utilizadas nas aulas leccionadas, tais como, *Node.js*, *Express*, *MongoDB* e *Docker*.

Capítulo 2

Arquitetura

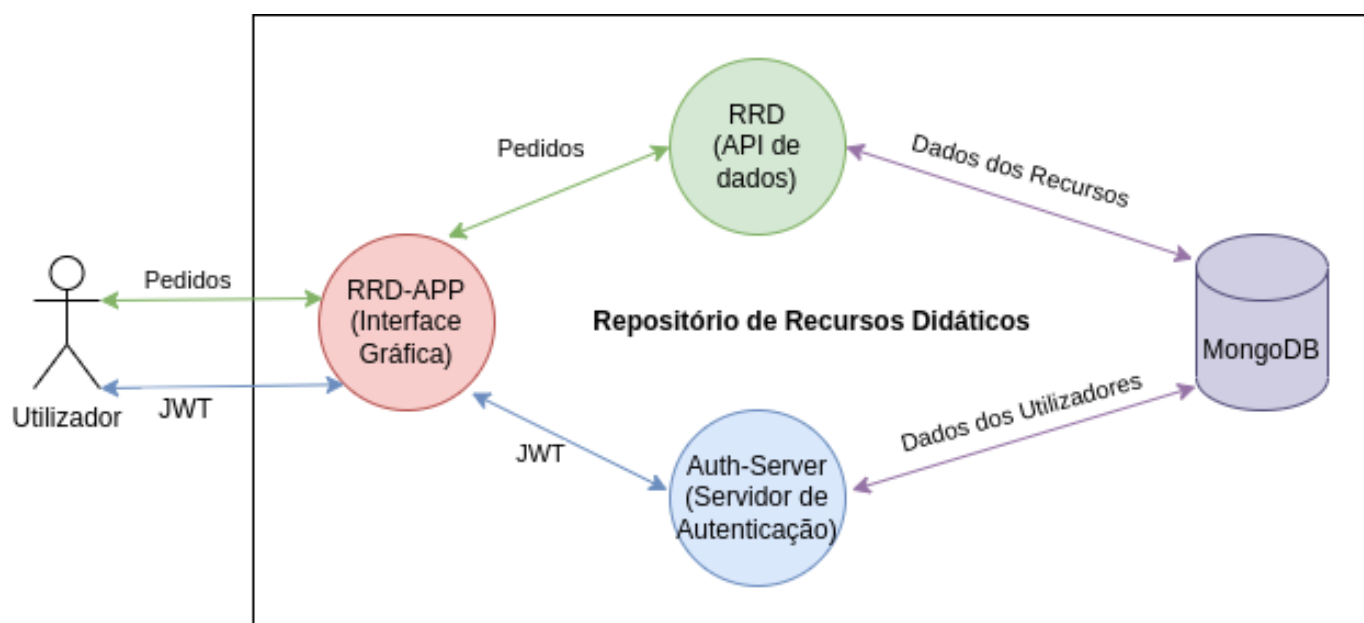


Figura 2.1: Relatório gerado

Capítulo 3

Funcionalidades

3.1 Utilizadores

Dado a natureza do modelo *OAIS* utilizado como orientação no desenvolvimento deste projeto, temos 3 tipos de utilizadores baseados nos três atores presentes no modelo, tendo cada um nível de acesso à plataforma distintos

- **Administrador:** Utilizador que é o Administrador do Sistema. É o utilizador com maior nível no sistema, estando desprovido de restrição de utilização no sistema;
- **Produtor:** Utilizador do Sistema com nível intermédio no sistema. Um produtor tem a funcionalidade de depositar recursos no Sistema bem como editar recursos existentes depositados pelo mesmo;
- **Consumidor:** Utilizador do Sistema com nível mais baixo. Devido a isto, é o único utilizador à qual não necessita de ter registo no sistema. Um consumidor pode visualizar ou realizar o *download* dos recursos presentes no sistema.

3.2 Autenticação

A autenticação presente no sistema é realizada através do uso de *JSONWebToken*, ou seja, ao ser realizado o login de um utilizador, é atribuído um JWT a esse utilizador. Para além disso, esse JWT é adicionado a um campo *token* presente nos seus *cookies*.

O *token* presente na *cookie* contém informações de forma a ser possível identificar a que utilizador o token pertence, tais como:

- O username do utilizador;
- O nível de acesso do utilizador.

3.2.1 Nível de acesso no token

O nível de acesso presente no *token* é atribuído de forma diferente para os vários utilizadores do sistema:

- Como um consumidor não necessita de fazer *login* no sistema, terá um *token* com o nível *"consumer"* atribuído por defeito;
- Todo o utilizador que realiza o registo no sistema terá o token com o nível *"producer"*, permitindo assim que possa realizar o depósito de recursos no sistema;
- Para o administrador, o nível é atribuído sendo colocado manualmente através de uma *query* à base de dados do sistema.

3.3 BagIt

Os recursos que são disponibilizados na plataforma são depositados e enviados no formato *BagIt*, um requisito pedido. O BagIt consiste num ficheiro *zip* com o seguinte formato:

- **data:** Pasta que contém os vários ficheiros do recurso a ser submetido;
- **RRD-SIP.json/RRD-DIP.json:** Ficheiro que contém o *checksum*, gerado pelo algoritmo SHA512 de cada um dos ficheiros presentes na pasta *data*;
- **metadata.json:** Ficheiro que contém toda a metainformação presente na base de dados associada ao recurso.

Quando um recurso é depositado no sistema é realizado um *BagIt* sobre o mesmo recurso. Após a criação do *bagit*, este mesmo é validado, verificando se todo o conteúdo presente no recurso corresponde ao recurso depositado pelo Utilizador. Posteriormente os recursos são armazenados numa zona do *file system* e com uma *path* associada ao SHA512 gerado do nome do recurso.

Quando é realizado um download sobre um pedido, é recriado um *zip*, em tudo igual ao *BagIt* com os ficheiros associados ao recurso.

3.4 Recursos

Os recursos consistem na matéria-prima apresentada no sistema. Tal como visto pela secção anterior, os recursos a serem depositados e descarregados contém um conjunto de metainformação, que estão presentes no *BagIt*. Assim que o recurso se encontra no sistema, este pode ser visualizado, dependendo do nível de visibilidade imposta pelo produtor que depositou o recurso, onde obtemos informação sobre o recurso tais como:

- Nome do recurso;
- Tipo de recurso;
- Ficheiros presentes no recurso;
- Secção de comentários onde os produtores e administradores do sistema podem postar um comentário;

3.5 Notícias/Logs

As notícias do sistema foi implementado como uma página de *logs* somente visível para o administrador.

A página de logs consiste em todo o histórico de alterações realizadas nos recursos, tais como, *upload* de um recurso, edição de campos de um recurso, edição e remoção de ficheiros associado a um recursos, realizar um comentário num recurso.

Capítulo 4

Extra

De forma a facilitar o *deployment* desta aplicação, foram gerados os *container Docker* do vários serviços que compõem a arquitetura do sistema, estando disponíveis no repositório *Docker Hub*.

Foi também gerado um *docker-compose* para orquestrar todos os *containers* do serviços da aplicação. A página dos *containers* da aplicação apresenta um *docker-compose* exemplo para realizar o *deployment* da aplicação, que pode ser visto aqui

Capítulo 5

Conclusão e Trabalho Futuro

O desenvolvimento desta aplicação permitiu consolidar toda a matéria lecionada e aumentar a complexidade de utilização das ferramentas e tecnologias abordadas nas aulas, uma vez que colocamos estas tecnologias em práticas num aplicação com contexto real.

No geral, consideramos este trabalho como um trabalho bem conseguido, resultando numa aplicação complexa com um manutenção e utilização simples. Dada a natureza do trabalho ser uma aplicação web, existem várias ideias e funcionalidades que possam ser modificadas e atualizadas bem como a utilização de outras tecnologias para além das que foram lecionadas nas aulas. Por exemplo, várias funcionalidades que podiam ser adicionadas seriam um processo de classificação dos recursos e contagem de visualizações e *downloads* de um recurso.