



UNIVERSIDADE DO MINHO

DESENVOLVIMENTO DE APLICAÇÕES WEB

---

## **Trabalho Prático**

---

RELATÓRIO DE DESENVOLVIMENTO

---

Mestrado Integrado em Engenharia Informática

***Realizado por:***

Hugo André Coelho Cardoso, a85006  
Válter Ferreira Picas Carvalho, a84464

7 de fevereiro de 2021

## Conteúdo

<b>1</b>	<b>Introdução</b>	<b>2</b>
<b>2</b>	<b>Arquitetura</b>	<b>3</b>
<b>3</b>	<b>Funcionalidades</b>	<b>4</b>
3.1	Utilizadores . . . . .	4
3.2	Autenticação . . . . .	4
3.3	Recursos . . . . .	4
3.4	Publicações . . . . .	5
3.5	Notícias . . . . .	5
3.6	BagIt . . . . .	5
<b>4</b>	<b>Conclusão</b>	<b>7</b>

## Lista de Figuras

1	Arquitetura final da aplicação . . . . .	3
2	Geração do ficheiro em formato zip BagIt no browser do cliente . . . . .	6



## 1 Introdução

No âmbito da Unidade Curricular Opcional *Desenvolvimento de Aplicações WEB*, lecionada no Mestrado Integrado em Engenharia Informática da Universidade do Minho, foi proposta a resolução do projeto prático **Open Archival Information System**, utilizando uma aplicação Web.

Este projeto consiste no desenvolvimento de uma plataforma que, resumidamente, disponibiliza vários recursos para utilizadores fazerem *download* ou *upload*, assim como publicações e notificações sobre os mesmos.

A solução obtida, com base nesta arquitetura, tira partido das ferramentas lecionadas em contexto de aula, nomeadamente Node.JS, Express, W3.CSS e PUG, assim como ferramentas extra, tais como Bootstrap e Mongo Compass.

De seguida será explicitado a arquitetura utilizada assim como as características da mesma.

## 2 Arquitetura

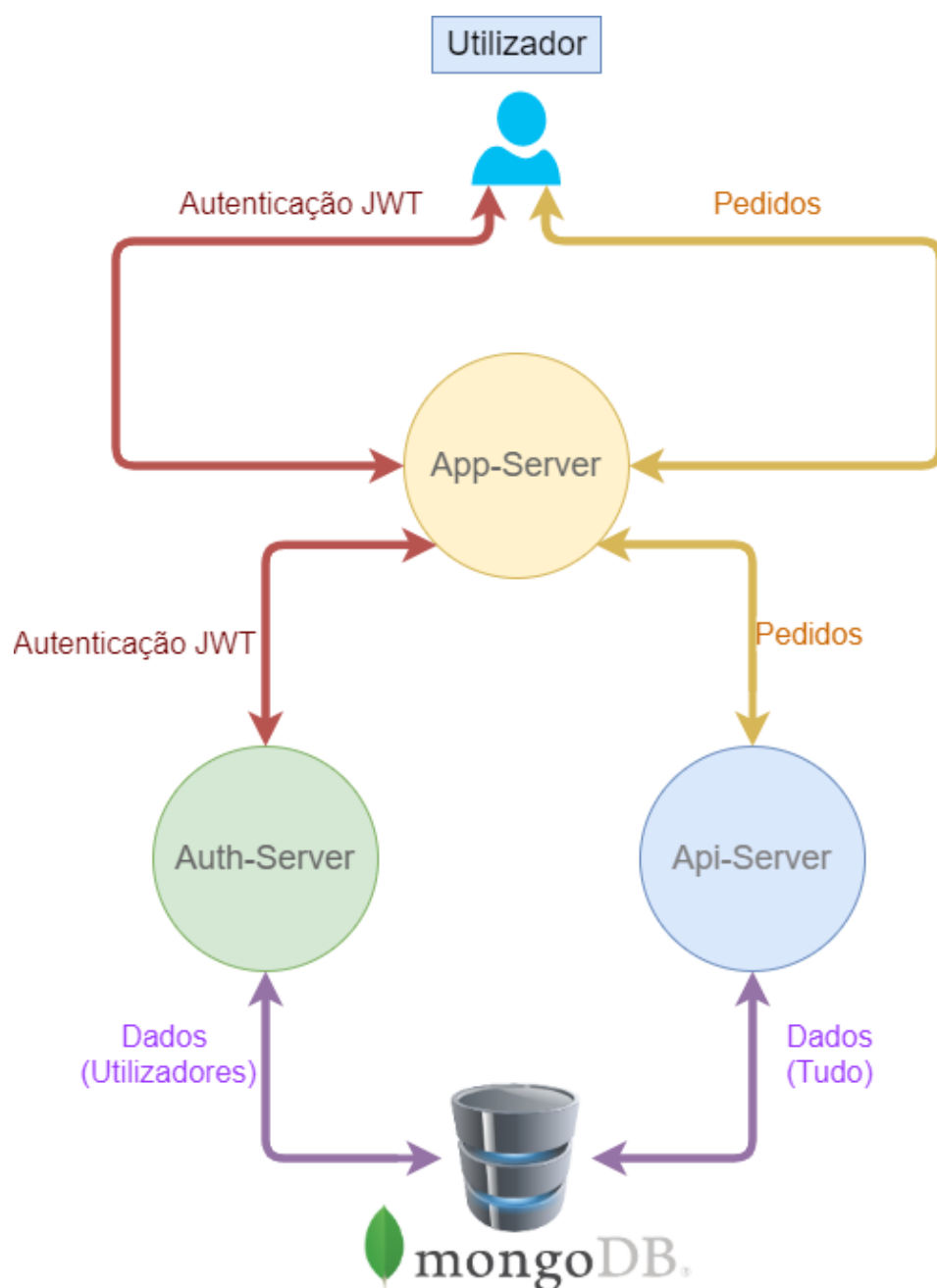


Figura 1: Arquitetura final da aplicação



## 3 Funcionalidades

### 3.1 Utilizadores

Os utilizadores têm um perfil único com o seu histórico de atividade na plataforma, nomeadamente os recursos que disponibilizaram e as publicações criadas.

Cada utilizador tem apenas um único nível de acesso à plataforma, isto é, tem apenas permissões para realizar algumas operações:

- **Consumidor:** nível mais baixo, não exige registo na plataforma. Não lhe é permitido editar, remover ou adicionar recursos, apenas visualizar ou fazer *download* e não consegue realizar nem comentar publicações.
- **Produtor:** nível intermédio, exige registo na plataforma. Consegue editar o seu perfil, adicionar recursos e editar ou remover os seus recursos já adicionados previamente, assim como criar publicações sobre os vários recursos da plataforma ou comentar nelas.
- **Administrador:** nível mais elevado, não tem restrições. Pode também bloquear o acesso de um utilizador diretamente a partir do seu perfil.

### 3.2 Autenticação

Ao realizar o *login*, é atribuído aos utilizadores um JSONWebToken (**JWT**) e adicionado aos seus cookies um campo *token* com o mesmo, que mantém a informação pertinente do utilizador tal como o seu nome, identificador, email, nível, etc.

O nível depende do utilizador registado, se for um administrador é-lhe atribuído o nível de **administrador**. Caso contrário, é-lhe atribuído de **produtor**.

Se o utilizador não realizar login, é-lhes atribuído por defeito um *token* com nível de **consumidor**, sem informação extra sobre o utilizador.

### 3.3 Recursos

Os recursos são o foco principal do projeto e, como tal, são disponibilizadas ao utilizador várias operações sobre os mesmos.

É possível submeter (**upload**) recursos na plataforma, que são constituídos por um conjunto de 1 ou mais ficheiros, assim como os metadados relevantes tais como título, data de criação, entre outros. Estes recursos são entregues no formato **BagIt** a ser analisado mais à frente.

Cada recurso pode ser marcado como público ou privado, pelo que apenas os públicos são exibidos na listagem de recursos disponíveis para *download* para outros utilizadores, que não o autor do recurso.

O produtor que submeteu o recurso tem a possibilidade de o editar futuramente, removendo ou adicionando ficheiros bem como alterando os seus meta-dados.

Todos os utilizadores podem visualizar o conteúdo dos ficheiros contidos no recurso e descarregar (**download**) o recurso ou um conjunto deles, desde que sejam públicos - o ficheiro gerado também é entregue no formato **BagIt**.

Na página de listagem de recursos, é possível ainda interagir com os recursos disponibilizados através de métodos de ordenação (ascendente ou descendente):

1. Classificação
2. Data de modificação
3. Número de publicações
4. Número de downloads

Assim como métodos de ordenação, a plataforma disponibiliza também filtros sobre os recursos:

1. Título, para obter recursos com um dado título
2. Tipo, para obter recursos de um dado tipo
3. Autor, para todos os recursos de um autor
4. Data, para obter todos os recursos criados num dado ano
5. Recursos disponibilizados pelo utilizador, que apenas mostra os recursos que o utilizador carregou para a plataforma

Tanto os métodos de ordenação como os filtros funcionam **simultaneamente**, isto é, o utilizador pode filtrar todos os recursos que disponibilizou de um certo tipo ordenados por classificação, por exemplo.

### 3.4 Publicações

Os produtores podem fazer publicações sobre um determinado recurso, onde outros produtores podem inserir comentários.

As publicações exprimem uma opinião ou local de discussão sobre um determinado recurso, muito semelhante ao *Reddit* - cada publicação contém uma descrição e um título, assim como uma lista dos comentários efetuados nela.

### 3.5 Notícias

Quando são efetuados **uploads** de recursos **públicos**, os utilizadores da plataforma são notificados desse mesmo acontecimento, de modo a saberem quando efetivamente aconteceu essa disponibilização do recurso.

Também são geradas notícias sobre um recurso quando um utilizador o edita, desde que continue público.

Cada notícia menciona o recurso disponibilizado, o seu tipo e o respetivo autor.

Tanto as notificações como publicações estão na página inicial da plataforma (*home*).

### 3.6 BagIt

Os ficheiros que são disponibilizados na plataforma são recebidos e enviados num formato **BagIt** simplificado, exposto na imagem seguinte:

O ficheiro *zip* é constituído por 2 partes:

- **data**: pasta que contém os vários ficheiros do recurso a ser submetido
- **manifest-md5.txt**: ficheiro que contém o *checksum* de cada um dos ficheiros a serem submetidos, gerados com o algoritmo MD5.

O **browser** do cliente é responsável pela conversão dos vários ficheiros num formato **BagIt**, evitando assim que o utilizador tenha que gerar o seu próprio *zip* antes de enviar.

Este ficheiro é, depois, validado do lado do servidor caso seja um **upload** e, caso o manifesto seja verdadeiro para os vários ficheiros presentes no *zip*, é armazenado com sucesso o recurso e os ficheiros respetivos **individualmente**, garantindo assim que não são submetidos ficheiros adulterados/infechos na plataforma.

Caso seja pedido um download, é recriado um *zip* com os ficheiros de cada um dos recursos com a mesma estrutura. Caso sejam pedidos 2 ou mais recursos, a pasta **data** contém os *zip* dos vários recursos e o manifesto conterá o respetivo *checksum* dos *zip*.

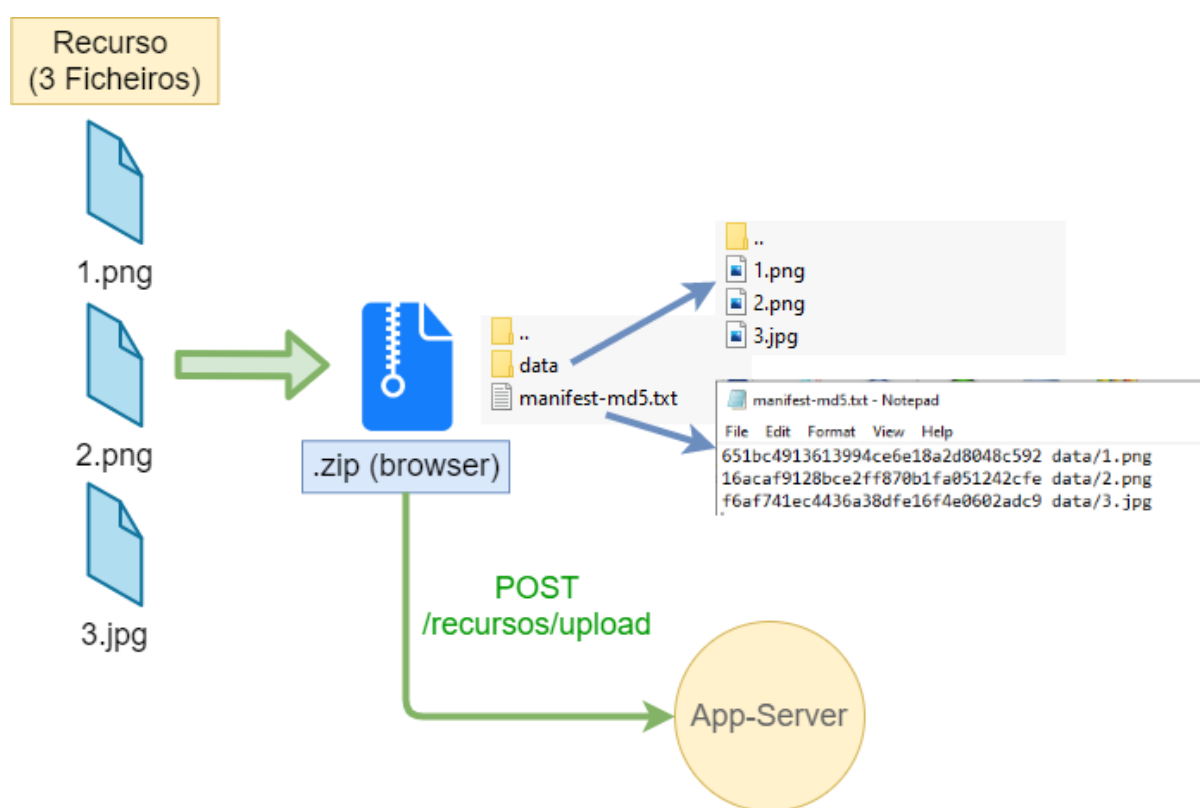


Figura 2: Geração do ficheiro em formato zip BagIt no browser do cliente



## 4 Conclusão

O desenvolvimento deste projeto permitiu a consolidação da matéria teórico-prática lecionada nas aulas da unidade curricular, nomeadamente da utilidade de ferramentas como o NodeJS, Express, PUG e JWT em contextos práticos.

O grupo está bastante satisfeito com o resultado final e considera que não só cumpriu os requisitos propostos no enunciado, como foi para além disso e desenvolveu uma solução escalável e dinâmica, uma vez que os módulos são independentes.

Concluindo, foi possível obter conhecimentos valiosos através da elaboração deste trabalho prático relativamente à programação por eventos, assim como conhecimentos transversais, nomeadamente trabalho em grupo e utilização de controladores de versões para uma melhor gestão de tarefas.