



Universidade do Minho  
Mestrado Integrado em Engenharia Informática  
4ºano - 1º Semestre

## **Desenvolvimento de Aplicações WEB**

### **Plataforma de Gestão e Disponibilização de Recursos Educativos**

```
console.log(team_name)
```



a84430 – Bruno Miguel Fernandes Teixeira  
a84752 - Diogo Miguel Pinto Rio  
a85334 – Pedro Paulo Fontes Delgado

7 de fevereiro de 2021

## Conteúdo

<b>1</b>	<b>Introdução</b>	<b>2</b>
<b>2</b>	<b>Implementação</b>	<b>2</b>
<b>3</b>	<b>Utilizadores</b>	<b>2</b>
3.1	Modelo . . . . .	3
3.1.1	Departamentos . . . . .	3
3.2	Autenticação . . . . .	3
3.3	Administração . . . . .	4
<b>4</b>	<b>Recursos</b>	<b>4</b>
4.1	Modelo . . . . .	5
4.1.1	Tipos . . . . .	5
4.2	Upload/Download . . . . .	5
4.2.1	Manifesto . . . . .	6
4.2.2	Armazenamento . . . . .	6
<b>5</b>	<b>Notícias</b>	<b>6</b>
5.1	Modelo . . . . .	6
<b>6</b>	<b>Conclusão</b>	<b>7</b>

## Lista de Figuras

1	Exemplo da informação de um utilizador . . . . .	3
2	Exemplo da informação de um departamento . . . . .	3
3	Exemplo de uma sessão antes de se efetuar o login . . . . .	3
4	Exemplo de uma sessão após efetuar login . . . . .	4
5	Exemplo da informação de um recurso . . . . .	5
6	Exemplo de tipos de recursos . . . . .	5
7	Exemplo de SIP e manifesto . . . . .	6
8	Exemplo da informação de uma noticia . . . . .	6

# 1 Introdução

No âmbito da UC de *Desenvolvimento de Aplicações WEB*, foi-nos proposta a realização de um trabalho que consiste na elaboração de uma plataforma de gestão e disponibilização de recursos.

Esta aplicação consiste num servidor aplicacional que recebe e responde a pedidos tendo como principais funções o registo e *login* de utilizadores, o *upload* e *download* de recursos educativos assim como a visualização de um recurso e um vasto conjunto de operações sobre o mesmo, que serão posteriormente indicadas.

Neste relatório é apresentada e fundamentada de forma sucinta a implementação da aplicação desenvolvida assim como algumas das suas funções.

## 2 Implementação

Para a implementação desta aplicação foi utilizada a *framework Express* indicada para desenvolvimento de aplicações WEB, assim como uma base de dados em *MongoDB* para fazer o armazenamento de parte dos dados da aplicação.

Um requisito para a implementação desta aplicação consiste na mesma seguir um modelo **Open Archival Information System**.

Deste modo foram implementados 3 níveis de acesso à aplicação (Administrador, Produtor e Consumidor) assim como um modelo de autenticação. O Administrador consiste num utilizador com permissão para efetuar todas as operações no sistema, entre elas a remoção de utilizadores e de recursos e acesso a estatísticas acerca da informação no sistema. Um Produtor consiste num utilizador que após o seu registo tem acesso a todos os recursos públicos, pode efetuar o *upload* e *download* de recursos, pode comentar e classificar recursos, entre outros. Já um Consumidor consiste num utilizador exatamente com as mesmas permissões de um Produtor, contudo este opta por não fazer *upload* de recursos apesar de ter permissão para tal.

Relativamente aos recursos, para que seja possível realizar operações de *upload*, armazenamento e *download*, existem 3 tipos de pacotes: *Submission Information Package (SIP)*, *Archival Information Package (AIP)* e *Dissemination Information Package (DIP)*. Além disso cada recurso possui um conjunto de informações associadas que serão posteriormente indicadas.

## 3 Utilizadores

Um utilizador é uma das entidades principais e, como anteriormente referido, existem 3 níveis de acesso à aplicação (Administrador, Produtor e Consumidor). Relacionado com um utilizador foram implementadas as várias operações, entre as quais:

- Registo de um utilizador
- Login de um utilizador
- Logout de um utilizador
- Listagem dos utilizadores registados na plataforma
- Edição de um utilizador (por parte do utilizador)

## 3.1 Modelo

```
{
  "_id" : "ObjectId('601d63a3ea4d6b2e8e511739')",
  "username" : "teste123",
  "email" : "teste@teste.com",
  "name" : "teste",
  "type" : "user",
  "regDate" : "2021-02-05",
  "hash" : "b3ea1325e3c4f9032515b7503495fa4932caf5e174bffa6766e75d45859e2d798f90dc5b861c3b6f394d3c445f3b568f704e01f38573facd40f9fd7535b718d1f",
  "salt" : "b432ac56154e82cfeefa53a8f091edb40cfe8cec5763e9c5a89538edf11c9d98",
  "location" : "Braga",
  "department" : "D000000",
  "course" : "C000004",
}
```

Figura 1: Exemplo da informação de um utilizador

### 3.1.1 Departamentos

Um dos "campos" presentes num utilizador consiste no Departamento e no Curso do mesmo. De seguida segue-se um exemplo de um Departamento e dos Cursos associados a esse departamento. Foi explorada a hipótese de adicionar um novo Departamento e de lhe associar novos Cursos, contudo essas operações são apenas permitidas ao administrador.

```
{
  "_id" : "ObjectId('601abacc09b83c5912a455cd')",
  "id" : "D000004",
  "designation" : "Statistics Dept.",
  "courses" : [
    {
      "_id" : "ObjectId('601abacc09b83c5912a455ce')",
      "id" : "C000027",
      "designation" : "Modern Statistics and Statistical Machine Learning"
    },
    {
      "_id" : "ObjectId('601abacc09b83c5912a455cf')",
      "id" : "C000028",
      "designation" : "Statistical Science"
    },
    {
      "_id" : "ObjectId('601abacc09b83c5912a455d0')",
      "id" : "C000029",
      "designation" : "Statistics"
    }
  ],
}
```

Figura 2: Exemplo da informação de um departamento

## 3.2 Autenticação

A estratégia escolhida para a autenticação de utilizadores fez uso do módulo Passport-local e é baseada em sessões geridas pelo servidor. Para cada ligação estabelecida com o servidor, é guardada uma sessão. Mesmo para um mesmo utilizador, se este se ligar através de um browser diferente, o servidor guardará duas sessões distintas. Quando uma ligação é autenticada através de um login, que faz uso dos campos username e password, a sessão passa a ficar associada a um utilizador. Por forma a identificar a sessão, o cliente que estabelece a ligação deve enviar um cookie para que o servidor reconheça a sessão. Este cookie é gerado pelo servidor na primeira vez que o cliente estabelece a ligação, ou seja, na primeira mensagem enviada e coincide com a criação da sessão pelo servidor. A partir deste momento, deverá ser enviado em todas as mensagens pelo cliente. A duração máxima de uma sessão foi estabelecida em 1 hora e pretende evitar que utilizadores ausentes se tornem vulneráveis. No final, a sessão é destruída pelo servidor.

```
> db.sessions.find().pretty()
{
  "_id" : "0jn4pDtOE9gC-V9enjRLKdKMcCj1yRNT",
  "expires" : ISODate("2021-02-05T23:59:34.813Z"),
  "session" : "{\"cookie\":{\"originalMaxAge\":3600000,\"expires\":\"2021-02-05T23:59:34.668Z\",\"httpOnly\":true,\"path\":\"/\"}}",
}
```

Figura 3: Exemplo de uma sessão antes de se efetuar o login

```
> db.sessions.find().pretty()
{
  "_id" : "0jn4pDt0E9gC-V9enjRLKdKMcCj1yRNT",
  "expires" : ISODate("2021-02-06T00:00:42.974Z"),
  "session" : "{\"cookie\":{\"originalMaxAge\":3600000,\"expires\":\"2021-02-06T00:00:42.774Z\",\"httpOnly\":true,\"path\":\"/\"},\"passport\":{\"user\":\"601d4d663291a31b723c179c\"}}"
```

Figura 4: Exemplo de uma sessão após efetuar login

### 3.3 Administração

O administrador do sistema possui permissão para efetuar todas as operações enumeradas ao longo deste relatório, nomeadamente o conjunto das operações CRUD sobre utilizadores e recursos. Além dessas operações o administrador possui uma série de estatísticas sobre os dados do sistema e pode alterar as permissões de um utilizador de modo que este tenha ou não permissões de administrador.

## 4 Recursos

Um recurso é a entidade principal da aplicação e consiste, como o nome indica, num recurso académico. Relacionado com um recurso foram implementadas as várias operações, entre as quais:

- Registo de um recurso
- Edição de um recurso (por parte do autor do recurso)
- Remoção de um recurso (por parte do autor do recurso)
- Listagem dos recursos públicos disponíveis na plataforma
- Comentar um recurso
- Remover comentário de um recurso (por parte do autor do comentário ou do recurso)
- Classificar um recurso
- Visualizar a página de um recurso
- Procurar um recurso pelo título e tipo
- Procurar todos os recursos de um determinado utilizador
- Ordenar recursos por título, data de criação e pontuação
- Upload e Download de um recurso

## 4.1 Modelo

```
{
  "_id" : "ObjectId('601d647aea4d6b2e8e511878')",
  "typeR" : "Report",
  "title" : "testerecurso",
  "creDate" : "2021-02-05",
  "regDate" : "2021-02-05T15:30",
  "visibility" : "Public",
  "nameR" : "example.zip",
  "author" : "teste123",
  "points" : [
    {
      "_id" : "ObjectId('601d649dea4d6b2e8e51187c')",
      "author" : "teste123",
      "point" : 4
    }
  ],
  "totalP" : 4,
  "path" : "/public/fileStore/teste123/example2021-02-05T15:30",
  "description" : "Isto é um teste",
  "comments" : [
    {
      "_id" : "ObjectId('601d648dea4d6b2e8e51187a')",
      "date" : "2021-02-05T15:30",
      "author" : "teste123",
      "description" : "teste comentário "
    },
    {
      "_id" : "ObjectId('601d649aea4d6b2e8e51187b')",
      "date" : "2021-02-05T15:30",
      "author" : "teste123",
      "description" : "outro comentário"
    }
  ]
}
```

Figura 5: Exemplo da informação de um recurso

### 4.1.1 Tipos

De seguida seguem-se alguns tipos de recursos disponíveis na aplicação. Foi explorada a hipótese de adicionar novos tipos de recursos, sendo que essa operação é apenas permitida para o administrador do sistema por motivos de incongruência por parte dos utilizadores aquando do registo do recurso.

```
{ "_id" : "ObjectId('601d635aea4d6b2e8e511733')", "type" : "Report", "_v" : 0 }
{ "_id" : "ObjectId('601d635aea4d6b2e8e511734')", "type" : "Thesis", "_v" : 0 }
{ "_id" : "ObjectId('601d635aea4d6b2e8e511735')", "type" : "Application", "_v" : 0 }
{ "_id" : "ObjectId('601d635aea4d6b2e8e511736')", "type" : "Slides", "_v" : 0 }
```

Figura 6: Exemplo de tipos de recursos

## 4.2 Upload/Download

O requisito de *upload/download* de um recurso foi implementado, e para tal são utilizados 3 tipos de pacotes como anteriormente referido (AIP, SIP e DIP). Optamos por considerar que um pacote SIP é muito semelhante a um pacote DIP, em que um pacote SIP consiste num ficheiro **ZIP** no qual estão presentes uma pasta **data** e um ficheiro **manifesto.json** o qual é utilizado para fazer a verificação do conteúdo do pacote SIP. Relativamente ao pacote DIP é igualmente um ficheiro **ZIP** onde dentro do mesmo está presente exatamente o mesmo conteúdo de um pacote DIP, contudo os ficheiros não se encontram dentro de uma pasta **data**. De seguida é apresentado um exemplo de um pacote SIP e de um manifesto ilustrativo do mesmo com o qual é feita a verificação do *type* e *title* de cada ficheiro.

### 4.2.1 Manifesto

```
example2021-02-05T15:30/
├── data
│   ├── 1.txt
│   ├── 2.txt
│   └── 3.txt
└── manifesto.json

{
  "files": [
    {
      "title": "1",
      "type": "txt"
    },
    {
      "title": "2",
      "type": "txt"
    },
    {
      "title": "3",
      "type": "txt"
    }
  ],
  "folders": [
  ]
}

"manifesto.json" [dos] 21L, 327C
```

Figura 7: Exemplo de SIP e manifesto

### 4.2.2 Armazenamento

Relativamente ao armazenamento dos recursos optou-se por uma politica de armazenamento simples e, do nosso ponto de vista, adequada à dimensão do problema que consiste em armazenar os recursos (AIP) numa pasta *fileStore* dentro da diretoria *public*. Houve ainda o cuidado de evitar o problema de exceder o número máximo de arquivos dentro de uma pasta e para tal fez-se a divisão dos recursos por o username do utilizador, ou seja, dentro da pasta *fileStore* são criadas diretorias com o username do utilizador sendo os recursos armazenados nas mesmas. Referir ainda que cada recurso, quando registado, guarda o seu *path* que é utilizado quando se efetua o seu *download*

## 5 Notícias

Outro dos requisitos para o trabalho consiste em sempre que um utilizador fizesse um registo de um recurso fosse possível gerar uma noticia/notificação acerca desse registo. Para tal foi criada a entidade Noticia que serve para isso mesmo, sempre que é registado um recurso é atualizada a lista de noticias na página principal da aplicação.

### 5.1 Modelo

```
{
  "_id" : "ObjectId('601d647aea4d6b2e8e511879')",
  "author" : "teste123",
  "date" : "2021-02-05",
  "body" : "The user teste123 published an article called testerecurso!",
  "title" : "New article by teste123",
}
```

Figura 8: Exemplo da informação de uma noticia

## 6 Conclusão

A elaboração deste trabalho prático permitiu cimentar conhecimentos e competências adquiridas nas aulas da UC de Desenvolvimento de Aplicações WEB, assim como explorar outras não tão aprofundadas. Durante o desenvolvimento do projeto, o grupo ultrapassou vários desafios até atingir os objetivos a que se propuseram.

Alguns dos desafios mais interessantes que foram superados destacam-se a autenticação dos utilizadores, a definição de um conjunto completo de características das entidades do sistema, o *upload*, *download* e armazenamento de recursos, a zona de administração com total permissão no sistema, a privacidade de dados dos utilizadores, assim como o desenvolvimento de uma interface responsiva e relativamente interessante.

Em suma, o grupo considera que os objetivos foram atingidos, tanto ao nível do exercício proposto como ao nível do desenvolvimento de competências, apresentando assim resultados satisfatórios.