

# RELATÓRIO DO AP2

Universidade de Aveiro

(102817) Rui Lameiras, (104090) Tomás  
Rodrigues,(103552) Alexandre Martins



# Índice

<b>1</b>	<b>Introdução do Trabalho proposto</b>	<b>1</b>
<b>2</b>	<b>Interface Web</b>	<b>2</b>
<b>3</b>	<b>Aplicação Web</b>	<b>3</b>
<b>4</b>	<b>Processador de Imagens</b>	<b>4</b>
<b>5</b>	<b>Erros no site</b>	<b>5</b>

## Capítulo 1

# Introdução do Trabalho proposto

Foi pedido neste projeto para criar uma plataforma digital na qual seja possível colecionar imagens. Para diferentes papéis desempenhados dentro da plataforma, para além da coleção será possível realizar o upload das imagens e da marca da água.

Na plataforma web, após o *login/registro* um utilizador normal (coleccionista) pode aceder a imagens da sua propriedade tanto como da propriedade de outro utilizador porém estas estarão marcadas com uma marca de água. Um coleccionista tem também a possibilidade de ceder a propriedade das suas imagens a outros coleccionistas, porém nunca terá acesso às imagens sem marca de água e todas as transações ficarão registradas.

Um curador não só tem acesso às imagens sem marcas de água como é o responsável por fazer upload destas.

## Capítulo 2

# Interface Web

A primeira página apresenta uma caixa de login com um link adicional para uma página dedicada ao registo de novos utilizadores.

Após um login com sucesso o utilizador tem acesso, na segunda página, a uma lista de coleções de imagens e à opção de criar uma nova coleção ou ver as imagens presentes nas coleções.

Optando por criar uma nova coleção, na terceira página, deverá proceder ao envio de uma ou mais imagens para o servidor. Para este efeito deverá enviar uma imagem assim como a marca de agua de cada vez através do seu browser.

Acedendo a uma coleção, na quarta página é possível ver as imagens da mesma coleção que tenham um proprietário. Todas as imagens contêm uma marca de água.

A quinta página deverá conter informação sobre os alunos e o repositório utilizado. Todas as páginas deverão obter informação através de pedidos à aplicação Web, que deverá devolver um objeto *JavaScript Object Notation* (JSON)[4]. Excecetuando a página inicial e de registo, todas as páginas deverão estar protegidas por um *"access token"* que não é mais do que uma string random que representa um utilizador autenticado.

## Capítulo 3

# Aplicação Web

A aplicação web consiste na conexão de todas as páginas web e fornece suporte à database e métodos *python* como, por exemplo, *pil*.

O *cherrypy* está dividido nas classes *rot*, onde se encontram as páginas base como login, sign e index, sendo este último a main page para operações como adicionar foto. *Log* contém todas as operações de suporte ao *log/sign* e como o *lofE.html* que é a página que aparece caso o *log* tenha falhado, us que é a class que contém tudo relacionado com qualquer usuário do site e image que, como o nome indica, contém tudo o que está relacionado com as marcas de água e a adição de imagens.

Temos de ter em conta, para além das classes que devedem o *cherrypy*, das funções exteriores ao *cherrypy* que fornecem ajuda para os relativos objetivos das classes.

## Capítulo 4

# Processador de Imagens

O processo de imagens é diferente caso seja a adição de uma imagem que já está na database ou uma nova.

Uma nova será solicitado na interface web duas imagens, uma que será a foto a ser adicionado e outra a marca de água e o utilizador quer que, cada das suas imagens, tenha.

Se for uma foto já decente será, simplesmente, solicitado ao user a marca de água.

Quer nos dois, o processo é semelhante a *i1* será a imagem e a *i2* a marca de água, duas imagens que serão adicionadas à pasta uploads( caso *i1* exista já na database será apenas *i2*) e que, o seu resultado, será adicionado a pasta de img que irá conter todas as imagens que foram processadas pelo server.

Depois da imagem ter sido processada pelo servidor o nome da imagem será adicionada há tabela pic como o nome do cliente que a adicionou e a sua password.

## Capítulo 5

# Erros no site

No final do projeto encontramos incompatibilidades com o template implementado, em particular na pagina de upload de imagens nao conseguindo realizar a visão estética que tínhamos em mente.

Incapazes de detectar o erro, decidimos priorizar a fucionalidade à estética. Na figura X encontra-se uma print do que pretendíamos que a pagina de upload de imagens fosse.



Figura 5.1: Exemplo de uma imagem da matriz representativa.

Outros aspetos fazem mostrar incopatibilidade com o template, o que era pretendido é no *index* que poderá ocorrer um bug na interface web mostrando as fotos dos utilizadores com diferentes tamanhos.

No *cherry.py* nas funções relativas da demotração das imagens no *index.html*, era suposto ser só mostrado uma imagem de cada utilizador que tem imagens na database, mas, por erros que também fomos incapazes de encontrar, as funções: *sendinfo()* e *sendinfo2()*, não enviam só uma imagem de cada utilizador causado com que apareça no *index* mais que uma vez a coleção de um utilizador se esse tiver mais que uma imagem no server

No guião não é exigida a utilização de *linux*, pelo que optámos pela utilização de windows na realização do projeto. Só verificamos se o projeto estava funcional em *linux* no final do projeto, no entanto devido a problemas na utilização da maquina virtual não conseguimos testar o código em sistema *linux*. Erros que ocorram em *linux* devem, em principio incidir no path.



# Contribuições dos autores

Rui Lameiras 50% Tomás Rodrigues 40% Alex Martins 10%