

# Projeto 2

Universidade de Aveiro

André Silva, André Alves, Rafael Santos,  
Leonardo Francisco



# Projeto 2

DETI

Universidade de Aveiro

(98651) andrecastrsilva@ua.pt, (98466) rafaelmsantos@ua.pt,  
(94190) afa@ua.pt, (97772) l.fiuza@ua.pt

Junho 2020

## **Resumo**

O objetivo deste projeto consistiu na criação de um sistema que permita inserir, visualizar e comentar imagens enviadas para uma aplicação, replicando funcionalidades que se encontram noutros serviços.

### **Agradecimentos**

Queremos agradecer aos nossos professores da unidade curricular de Laboratórios de Informática, André Ventura Zúquete, António Manuel Adrego da Rocha, João Paulo Barraca por nos ter guiado durante o projeto e incentivado a realizá-lo da melhor forma possível. Aprendemos mais um bocado sobre cada matéria à medida que fomos avançando no trabalho, demonstrando assim que este último trabalho prático é indispensável na cadeira de LABI.

# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>1</b>
<b>2</b>	<b>Interface Web</b>	<b>2</b>
2.1	Páginas Criadas . . . . .	2
<b>3</b>	<b>Aplicação Web</b>	<b>4</b>
<b>4</b>	<b>Persistência</b>	<b>5</b>
4.1	Base de dados . . . . .	5
<b>5</b>	<b>Processador de imagens</b>	<b>6</b>
<b>6</b>	<b>Conclusões</b>	<b>7</b>

# Capítulo 1

## Introdução

Este relatório aborda a maneira . Este documento está dividido em 4 capítulos principais, no Capítulo 2 é onde se fornece o único interface para interação com o sistema. No Capítulo 3 é onde apresentamos métodos que permitem a navegação entre os diversos componentes. O Capítulo 4 é composto por métodos que permitem o registo de informação numa base de dados relacional e a obtenção de informação da mesma. O Capítulo 5 que se executa como parte da Aplicação Web, é onde terá métodos para lidar com imagens enviadas para o sistema, ou a obtenção das mesmas.

Finalmente, no Capítulo 6 são apresentadas as conclusões retiradas do trabalho.

## Capítulo 2

# Interface Web

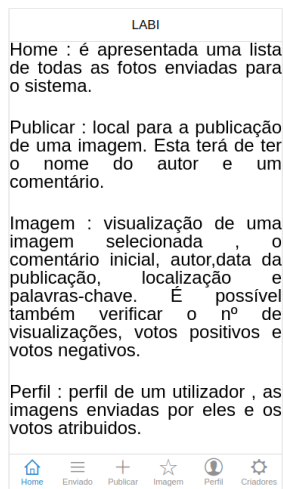
### 2.1 Páginas Criadas

No que toca à interface web foi usada uma template Ratchet( acessível em Ratchet Components).

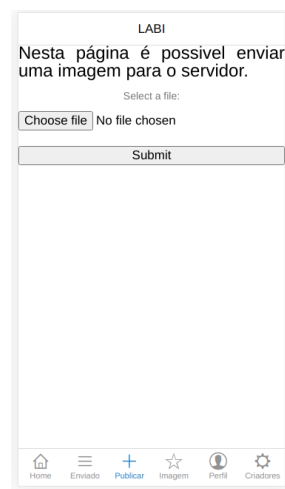
O que é o Ratchet? O Ratchet é uma framework com HTML, CSS e JavaScript permitindo criar páginas web que podem ser adaptadas a vários aparelhos diferentes.

Foram criadas 5 páginas html : uma Home , uma onde teria o feed de imagens , uma onde era possível publicar uma imagem, uma onde era possível ver o perfil da pessoa e por fim uma que fala do projeto e dos seus criadores.

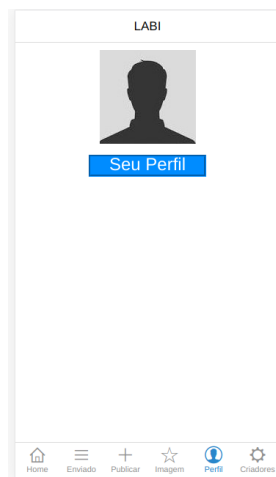




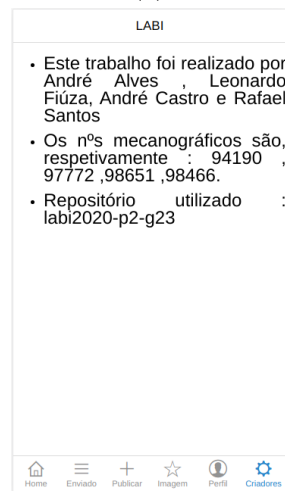
(a)



(b)



(c)



(d)

Figura 2.1: Todas as páginas a funcionar, menos a Perfil e a Publicar.

## Capítulo 3

# Aplicação Web

Para interligar as páginas e para que a aplicação funcionasse corretamente foi necessário utilizar uma framework.

Uma framework é um conjunto de código escrito que procura aumentar a eficiência do programador ao oferecer-lhe funções e funcionalidades de modo a que o programador não esteja constantemente a reescrevê-la para cada aplicação.

A framework utilizada neste projeto foi o cherrypy que não só serviu como framework como também funcionou como servidor web.

```
import os
import cherrypy

PATH = os.path.abspath(os.path.dirname(__file__))

class Root(object):
    def __init__(self):
        self.image = Image()
        self.comment = Comment()

    @cherrypy.expose
    def index(self):
        return open("index.html", "r").read()

class Image(object):
    @cherrypy.expose()
    pass

class Comment(object):
    @cherrypy.expose()
    @cherrypy.tools.json_out()
    pass

conf = {
    "/": {
        "tools.staticdir.on": True,
        "tools.staticdir.dir": os.path.join(PATH)
    }
}

cherrypy.server.socket_port = 10023
cherrypy.config.update({"server.socket_port": 10023})
cherrypy.server.socket_host = "0.0.0.0"
cherrypy.tree.mount(Root(), "/", conf)
cherrypy.engine.start()
cherrypy.engine.block()
```

Figura 3.1: Código aplicação

## Capítulo 4

# Persistência

### 4.1 Base de dados

A base de dados tem como objetivo o armazenamento de diversos tipos de informação relacionada, de tal forma que a sua atualização e consulta possa ser eficiente e efetuada num curto espaço de tempo e para facilitar a organização, manutenção e pesquisa de dados.

```
CREATE TABLE imagens (  
    image_id INTEGER PRIMARY KEY AUTOINCREMENT,  
    image_path TEXT,  
    creation_date INTEGER,  
    user_id INTEGER  
);  
  
CREATE TABLE comments (  
    comment_id INTEGER PRIMARY KEY AUTOINCREMENT,  
    comment_content TEXT,  
    image_id INTEGER  
);  
  
CREATE TABLE users (  
    user_id INTEGER PRIMARY KEY AUTOINCREMENT,  
    username TEXT  
);  
  
CREATE TABLE palavrachave (  
    palavrachave_id INTEGER PRIMARY KEY AUTOINCREMENT,  
    palavra TEXT  
);  
  
CREATE TABLE imagenspalavrachave (  
    id INTEGER PRIMARY KEY AUTOINCREMENT,  
    palavrachave_id INTEGER,  
    imagem_id INTEGER  
);
```

Figura 4.1: Esqueleto base de dados

## Capítulo 5

# Processador de imagens

No que toca ao processamento de imagens tivemos algumas dificuldades no envio da fotos para o sistema e no tratamento das mesmas.

Conseguimos realizar os efeitos , no entanto também não o conseguimos aplicar sendo o passo anterior necessário. Apesar disto , testamos os efeitos com imagens do nosso portátil e estão totalmente funcionáveis.

```
from PIL import Image
from PIL import ImageFilter
import sys

def main(fname):
    # Create an image object
    image = Image.open(fname)

    # Find the edges by applying the filter ImageFilter.FIND_EDGES
    image.filter(ImageFilter.FIND_EDGES)
    new_im = image.filter(ImageFilter.FIND_EDGES)
    new_im.save('edge'+fname)

if __name__ == '__main__':
    main(sys.argv[1])
```

(a) Efeito edge

```
from PIL import Image
from PIL import ImageFilter
import sys

def main(fname):
    im = Image.open(fname)
    blurred_img = im.filter(ImageFilter.BoxBlur(5))
    new_im = blurred_img.save('blur'+fname)

if __name__ == '__main__':
    main(sys.argv[1])
```

(b) Efeito blur

```
from PIL import Image
import sys

def main(fname):
    im = Image.open(fname)
    new_im = im.convert("L")
    new_im.save('bw'+fname)

if __name__ == '__main__':
    main(sys.argv[1])
```

(c) Efeito preto e branco

## Capítulo 6

# Conclusões

Apesar de não termos realizado o projeto na íntegra consideramos que este projeto nos ajudou a desenvolver capacidades e competências na área do desenvolvimento das aplicações web, manipulação de dados e base de dados. Concluindo, sabemos que não atingimos todos os objetivos mas enriqueceu o nosso percurso académico.

# Contribuições dos autores

Consideramos que houve uma participação de 25% atribuída a cada um dos autores.