

Proj2

MIECT

Mário Liberato, Jorge Oliveira



Proj2

DETI
MIECT

Mário Liberato, Jorge Oliveira
(84917) mliberato@ua.pt, (84983) jorge.am.oliveira@ua.pt

12-06-2017

Resumo

Para o 2º projeto, o grupo tinha um objetivo: criar uma aplicação móvel onde é possível modificar imagens, aplicando um efeito, adicionar texto ou transformar em meme enviando de seguida para um servidor onde também se pode visitar uma galeria com outras imagens submetidas. Este projeto foi realizado em Python tal como alguns módulos, sendo os mais usados CherryPy e Pillow. A aplicação necessitava de ter uma (frontend) onde era apresentado o conteúdo ao utilizador e uma (backend) onde esta um servidor, base de dados, aplicador de efeitos e gerador de memes. A aplicação pode ser acedida de duas forma: a mais convencional é através da página web on é apletivo e de uso direto; a segunda forma é a através de um terminal de escolha do utilizador, desta forma é possível interagir de forma mais profunda com a aplicação onde por exemplo poderá armazenar as imagens localmente invece de num servidor.

Conteúdo

1	Introdução	1
2	Metodologia	2
2.1	Testes	2
3	Descrição da aplicação	3
3.1	Página Web	3
3.1.1	Página Inicial	3
3.1.2	Submissão de Imagem	3
3.1.3	Galeria	3
3.2	Núcleo da Aplicação	4
3.2.1	Efeitos	4
3.2.2	Gerador de Memes	4
3.2.3	Backend	4
4	Resultados e Análise	8
4.1	Página Web	8
4.2	Núcleo	8
4.2.1	Tratamento de Imagem	8
4.2.2	Backend	8
5	Conclusões	9

Capítulo 1

Introdução

Neste projeto é pretendida a criação de uma aplicação móvel através da qual o utilizador possa enviar fotografias para um servidor, podendo ou não aplicar filtros. Também existe uma galeria contendo todas as imagens submetidas pelos grupos. O documento encontra-se dividido em quatro capítulos, sendo que no Capítulo 2 é apresentada a metodologia seguida para a criação da página e do programa em si. No Capítulo 4 são apresentados os resultados obtidos da aplicação e breve análise. Finalmente, no Capítulo 5 são apresentadas as conclusões do projeto.

Capítulo 2

Metodologia

O grupo optou por utilizar Python 3 como linguagem de programação principal usufruindo do módulo Pillow e CherryPy. Para desenvolver o programa o grupo foi desenvolvendo um pouco de cada módulo até à sua finalidade.

2.1 Testes

O grupo utilizou Python3 como a linguagem de programação principal. Foram usados diversos módulos, na imagem Pillow e OpenCV(2), para a *backend* maioritariamente CherryPy, e finalmente, para a *frontend* o módulo Ratchet.

Capítulo 3

Descrição da aplicação

3.1 Página Web

O objetivo do grupo era apresentar uma página simples e fácil de usar, usando o módulo Ratchet para compatibilidade móvel. Existiram protótipos(3.1) em que devido a problemas de compatibilidade não foram implementados. Um exemplo seria a implementação de código javascript que mostrava alterações na imagem em "direto" à medida que o utilizador alterava os efeitos ou texto.

Esta funcionalidade não foi implementada devido aos recursos utilizados no dispositivo tal como no servidor provocando grande lentidão.

3.1.1 Página Inicial

A página inicial (3.2) tem carácter simples e de uso fácil. O utilizador pode seleccionar uma imagem a qual é apresentada na própria página, e escolher os efeitos, ou o gerador de meme onde aplica texto e/ou um fundo à imagem. A letra da imagem ainda pode ser sombreada.

3.1.2 Submissão de Imagem

A submissão de imagens é feita através de um método `/api/put`, o qual usa um pedido POST que envia a imagem e os argumentos (com os efeitos a utilizar, por exemplo). Este método faz uso de um outro método na classe API para obter um ID para a imagem a carregar antes de o fazer.

3.1.3 Galeria

A página da galeria(3.3) é apresentada de forma simples, também permite o utilizador realizar votos nas imagens como consultar os mesmos. A apresentação da galeria de imagens usa o método `/api/listAll` onde é fornecida uma lista com todas imagens dos grupos, votos e autor. Os votos são realizados através de

/api/vote onde são enviados o identificador da fotografia, do utilizador e o tipo de voto.

3.2 Núcleo da Aplicação

3.2.1 Efeitos

Para a realização dos efeitos foi utilizado Python3 e o módulo Pillow. Os efeitos disponíveis no programa são: *blur*, escala de cinzentos, *lomography*, sepia e inversão de cor. Para a realização dos dois primeiros a imagem é processada com desfocagem e alterado o modo da imagem. Para os outros efeitos são aplicadas fórmulas matemáticas aos canais Vermelho, Verde e Azul para fazer modificações pixel a pixel à imagem.

3.2.2 Gerador de Memes

Este módulo permite aplicar texto à imagem tanto na parte inferior como superior, sombreado ou não. Também tem como objetivo recortar uma cara da imagem e por sobre um fundo branco ou colorido, à escolha do utilizador. No entanto, este módulo causou algum transtorno visto que não foi possível existir uma forma correta de recortar uma cara de uma imagem.

3.2.3 Backend

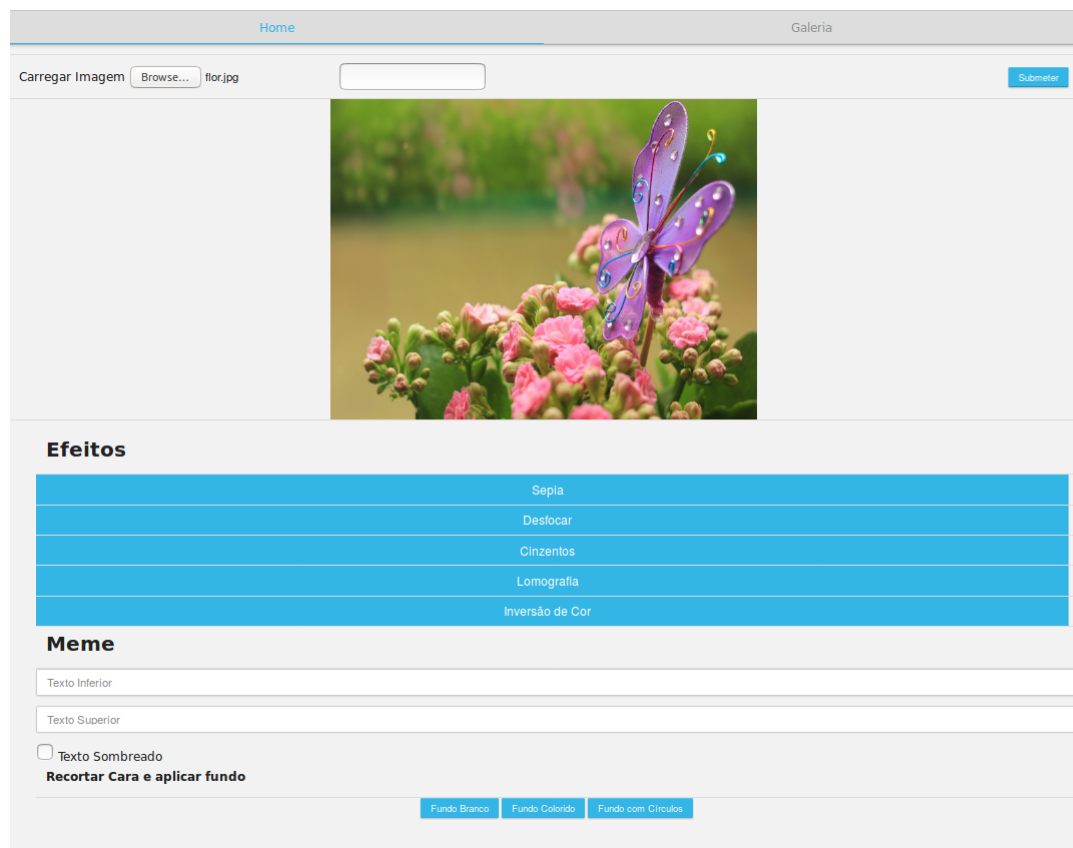


Figura 3.1: Protótipo da Página(Computador)

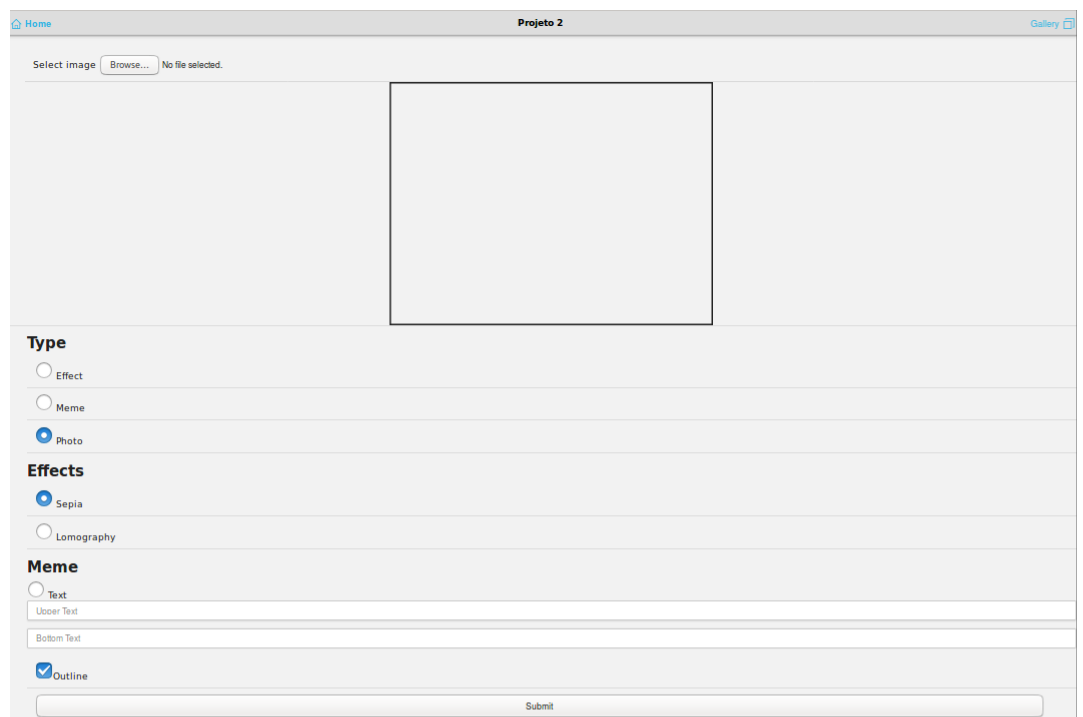


Figura 3.2: Página Inicial(Computador)

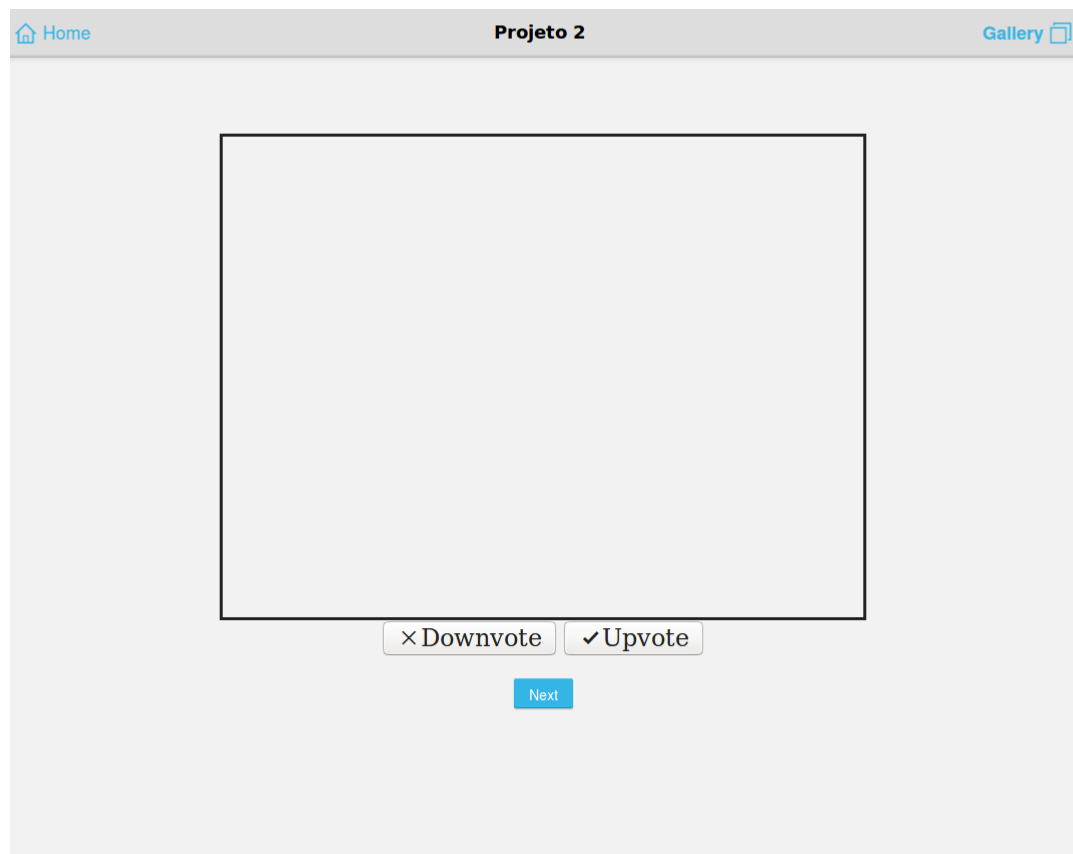


Figura 3.3: Página da Galeria(Computador)

Capítulo 4

Resultados e Análise

4.1 Página Web

As páginas não ficaram com o aspeto inicialmente pretendido devido a problemas com JavaScript. No entanto foi possível adaptar as mesma para ficarem fácil uso.

4.2 Núcleo

4.2.1 Tratamento de Imagem

O módulo dos efeito ficou feito em geral sem problemas, no entanto o mesmo não aconteceu no módulo de memes. Foram encontradas diversas dificuldades, sendo a maior delas o reconhecimento facial para recorte de uma cara, acabou por se ter de usar o módulo OpenCV para o reconhecimento dos olhos e face, sendo recortado num retângulo a cara encontrada, um problema conhecido é caso exista mais que uma cara detetada pelo algoritmo o resultado já não é o desejado. Existem dois fundos gerados pelo módulo Pillow que aplicam a cara recortada anteriormente. Foram criados dois ficheiros em Python distintos para aplicação de efeitos, um especial para mudança de cores e efeitos, outro para a criação dos memes. Todas as função destes ficheiros fizeram uso do módulo Pillow. Estes podem ser aplicados na própria consola do Python, importando os ficheiros e usar a função `meme_image("diretório.do.ficheiro",efeito,argumentos)` e `effect_image("diretório.do.ficheiro",efeito)`.

4.2.2 Backend

Capítulo 5

Conclusões

Apesar de não ter sido atingido o *design* e funcionalidades originalmente pretendidas pelo grupo, este conseguiu criar diversas aplicações ao tratamento de imagem.

Contribuições dos autores

ML trabalhou maioritariamente na backend da aplicação, realizando o servidor e a base dados tal como a adaptação dos efeitos ao servidor.

JO trabalhou maioritariamente na frontend da aplicação, dando a face ao programa nas páginas web, tal como o gerador de meme com os seus efeitos.

Acrónimos