Projeto 2

Universidade de Aveiro

João Oliveira, Bernardo Barreto,
Bernardo Falle, Diogo Vicente

VERSAO 1

Projeto 2

DEPARTAMENTO DE ELETRÓNICA
TELECOMUNICAÇÕES E INFORMÁTICA
Universidade de Aveiro

João Oliveira, Bernardo Barreto,
Bernardo Falle, Diogo Vicente
93282 joaoroliveira@ua.pt, 93271 @ua.pt,
93331 mbfale@ua.pt, 93262 dvicente@ua.pt

12 de junho de 2019

Resumo

Este projeto foi feito com o intuito de ganharmos mais conhecimentos na parte de programação em python tal como na utilização de HTML , protocolos HTTP, API's, entre uma serie de outras coisas. O objetivo final do trabalho era fazer uma pagina web que fosse possivel ser visualizada uma galeria de fotos, e fazer uma vasta pesquisa de acordo com vários argumentos que utilizador quisesse utilizar, podendo estes ser RGB, nome do objeto, nivel de confiança. Com isto realizamos codigo em pyhton utilizando a biblioteca da cherryPy criando assim um servidor para que fosse possivel estabelecer uma ligaçao entre a nossa database e o nosso site. Na parte de HTML foi utilizado todo o procedimento normal e após isto a utilização do css e do js. O primeiro para dar style ao nosso site, tornando assim mais *user friendly* e também com um aspeto melhor. O JavaScript foi usado para criar funções obtendo assim valores de diferentes sliders, etc... para depois podermos utilizar isto no nosso código e obter os resultados que o utilizador pretende. A identificação de objetos e de cores domintantes utilizadas como metodo de pesquisa no site são obtidas utilizando também o nosso código em Python juntamente com a máquina fornecidas pelos docentes da cadeira. Tornando assim um site completamente funcional e que identifica diversos objetos.

Agradecimentos

Agradecer aos colegas que sempre se disponibilizaram para ajudar. E um especial agradecimento a todos os docentes da cadeira que nos ajudaram em todas as nossas dúvidas.

Conteúdo

Capítulo 1

Introdução

Este relatório foi criado com o propósito de ir ao encontro com os objetivos mencionados no segundo projeto realizado no âmbito da unidade curricular de Laboratórios de Informática. O relatório primeiramente irá sofrer de uma breve apresentação introdutória, após isto teremos todas as fases fundamentais do projeto divididas por capitulos, por fim temos uma breve conclusão acerca do projeto.

Para este projeto foi-nos proposto que realizassemos um sistema que permita visualizar uma biblioteca de imagens com suporte para a procura e a identificação dos objetos contidos nas bibliotecas armazenadas, pode ser todo o tipo de coisas como, pessoas, automoveis, bancos, animais, etc.. Desta forma temos uma máquina disponibilizada pelos docentes da disciplina, que identifica estes objetos. Após isto foi-nos colocado o desafio de criar numa linguagem, à nossa escolha, nos optamos por python, um programa que relacionado com um site pode-se de facto dar-nos a biblioteca de imagens como foi proposto inicialmente.

O relatório será divido de acordo com as diferentes componentes do trabalho em capitulos, e subcapitulos que irão explicar devidamente todos os procedimentos e cuidados que tivemos ao realizar este trabalho. Este será o primeiro capitulo em que obtemos uma pequena introdução teórica do trabalho. Prontamente começamos com a explicaçao do Capítulo 2, após isto iremos falar da **??** explicando como funciona seguido do **??** e concluindo com a **??** destes todos.

Capítulo 2

Front-end

Na parte de Front-end, focamo-nos a usar html, css e js.
No css usamos o bootstrap que nos ajudou com todas as classes para que podessemos dar style ao projeto de maneira a que este ficassse esteticamente melhor e mais organizado. Javascript usamos algumas funções de modo a que conseguissemos retirar valores do nosso slider, por exemplo.

2.1 HTML

Nesta secção foi onde escrevemos todo o código HTML relativamente ao nosso site, esquematizando assim todas as páginas necessárias e o que nelas continha. Temos assim cinco ficheiros .html, em que cada um deles corresponde a uma pagina do site. O primeiro ficheiro corresponde à nossa home em que temos a nossa galeria apresentada, na segunda pagina o utilizador pode fazer uma pesquisa de imagem de acordo com o nivel de confiança, na terceira damos a hipotese de o utilizador dar upload a uma imagem para o nosso servidor que será guardada e depois apresentada na nossa galeria, na quarta será possivel pesquisar pelo objeto e pela devida cor dominante da imagem, na quinta temos apenas a informação básica de todos os membros do projeto assim como a divisão do trabalho. Tudo isto realizado na linguagem HTML com ajuda de JS e CSS que iremos explicar o porquê de seguida.

2.2 CSS

Na parte do CSS tivemos ajuda do bootstrap que nos facultou diversas classes, o que facilitou muito a parte de style do projeto. Também criamos ficheiros de css criados por nós para dar style a nossa galeria de fotos, e a outros objetos que preferimos fazer de raiz.

2.3 JS

No JavaScript utilizamos pouco, apenas realizamos pequenas funções que nos facultavam valores que depois poderiamos utilizar nos programas atrás para niveis de confiança, por exemplo.

Capítulo 3

Base de Dados

A base de dados construída na linguage SQL contém uma tabela *images* que tem 5 campos de informação

3.0.1 ID

É o campo mais simples, sendo que é o tipo *integer primary key auto increment*, isto é, cada vez que é feito o upload de uma imagem para o site o *id* incrementa automaticamente.

3.0.2 Classe

Este elemento da tabela é o nome do objeto extraído da imagem original que sofreu upload. Este nome é obtido através do JSON que é fornecido pela máquina do Docente João Paulo Barraca.

3.0.3 Original hash

Este campo é o nome que nos é dado pelo ficheiro existente que está armazenado nas pastas de storage (storage/objects) que contém as imagens originais. Este nome já vem encriptado do cherryPy.

3.0.4 Image hash

Este elemento da tabela é a designação da imagem já recortada com as coordenadas definidas pela máquina anteriormente descrita. Este Image hash contém sempre o mesmo nome (cropped) e depois, dependendo dos números de objetos extraídos, aumenta o número apresentado a seguir. Assim o nome da imagem recortada com um objeto vai ser, por exemplo: `cropped_0_OriginalHash`.

3.0.5 Confidence

Este membro da tabela vai buscar a confiança da máquina fornecida e redirecciona para o elemento *Confidence* da base de dados. Para esclarecimento, a confiança da imagem é a percentagem de certeza que o objeto extraído é, de facto, o objeto dado pela máquina.

3.0.6 Color

O último componente da tabela é o elemento Color, e é redireccionado de uma função feita pelos responsáveis por este tema. Este componente regista a cor predominante (red, green, blue) de cada objeto extraído. De seguida, armazena na base de dados.

Capítulo 4

Servidor

CherryPy é uma *web framework* do Python que através dela conseguimos utilizar facilmente uma interface para um protocolo HTTP. Também pode ser chamada de *web application library*.
O cherryPy tira proveito da linguagem dinamica do Python para passar o protocolo HTTP para uma API. O que fornece uma plataforma de confiança.

4.0.1 Statics

De modo a que o servidor consiga mostrar os conteúdos estáticos é necessário a utilização de um dicionário, *config*, que remete para as respetivas pastas destes conteúdos. O dicionário é depois utilizado ao dar *mount*.

4.0.2 Classes

List

Esta classe contém apenas um método chamado de index. Este método possui 3 argumentos que conforme os seus valores vão dar *return* a diferentes funções da *database*. Essas funções dão return a objetos JSON.

Get

Apenas contém uma função index, e como argumento o id. Esta função conforme o id passado irá devolver uma imagem completa.

Put

Esta classe também só possui uma função index que recebe um ficheiro como argumento, passando-a para *hash* em MD5, após isto a função irá guardar o ficheiro no server que postiormente manda para uma função na database que processa a imagem. Seguidamente reencaminha o utilizador para a página onde se encontrava.

Root

A root é a função que serve o conteúdo HTML ao utilizador. Possui cinco funções correspondentes às cinco páginas do site. Tem também uma função de inicialização das outras classes. Esta classe root é a classe central que serve de raiz para o servidor.

Capítulo 5

Análise

Analisando todos os objetivos, e fazendo uma recolha de testes podemos comprovar que no nosso trabalho apesar da maioria das funções funcionar, não conseguimos por a funcionar corretamente a listagem tanto de fotos como de nomes de objetos.
Segue também fotos de testes utilizados no site:

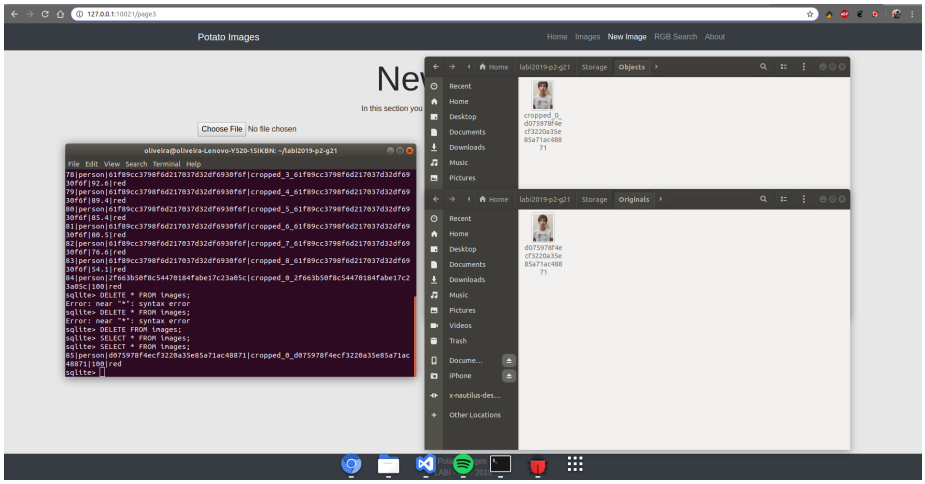

Figura 5.1: teste de adição de foto.

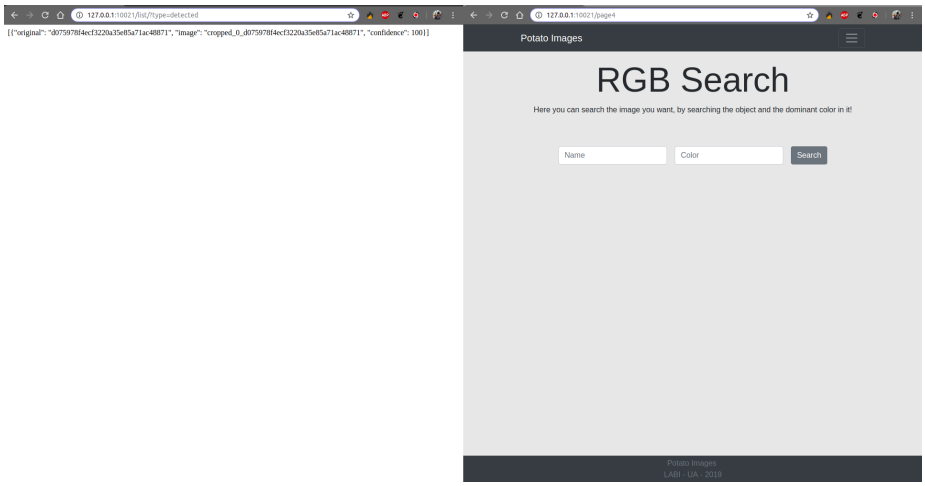

Figura 5.2: teste de verificação da cor e do objeto.

Capítulo 6

Conclusões

Concluindo este projeto, verificamos que apesar de conseguir fazer grande parte do projeto tivemos bastantes dificuldades em certas funções e em certas funcionalidades deste. Sendo algumas destas analisadas no capítulo acima, tendo assim também testes que comprovam o que está a funcionar corretamente e o que não está.

Contribuições dos autores

O João Oliveira (JO) tratou da maior parte da front-end.
O Bernardo Barreto (BB) tratou da maior parte dos requests.
O Diogo Vicente (DV) tratou da maior parte do servidor.
O Bernardo Fale (BF) tratou da maior parte da base de dados.
Apesar desta divisão todos os membros tiveram sempre em conjunto a fazer o trabalho ajudando-se mutuamente em todo o código realizado neste projeto, sendo assim a divisao da percentagem de contribuição de 25% para todos os membros.

Acrónimos

JO João Oliveira

BB Bernardo Barreto

DV Diogo Vicente

BF Bernardo Fale