
Projecto Computação Distribuída

Licenciatura em Engenharia Informática - Computação Distribuída

Distributed Photo Organizer

Docentes:

- Diogo Gomes (dgomes@ua.pt)
- Nuno Lau (nunolau@ua.pt)
- João Rodrigues (jmr@ua.pt)

Prazo:

22 de Junho – 24h00

Conceitos a abordar:

- Sockets
- Marshalling
- P2P
- Fault Tolerance

Introdução

Uns velhos amigos aveirenses reuniram durante anos fotos de Aveiro antiga que digitalizaram e guardaram numa pasta do seu computador pessoal, verdadeiros tesourinhos. Numa conversa entre eles resolveram partilhar fotos. Mas há um problema, juntar todas as fotos no mesmo computador não só é um esbanjamento de espaço de armazenamento, como não previne possíveis perdas de dados caso o computador falhe. O vosso projecto é pois criar um sistema de partilha P2P entre os vários amigos que lhes permita reunir em conjunto uma colecção de fotos nem iguais nem semelhantes (mesmo conteúdo mas tamanho, cor, anotações diferentes) entre si.

Para esta tarefa deverão desenvolver todo o código necessário para analisar uma pasta que contém ficheiros de imagem em vários formatos e tamanhos, partilhar essas imagens com os outros amigos num sistema P2P e garantir que todos têm acesso a uma cópia de cada imagem mesmo que um dos amigos possa perder o seu arquivo.

Para esta tarefa é fornecido este guião e um conjunto de imagens disponíveis em [3]. Devem criar o vosso projecto no GitHub Classroom usando <https://classroom.github.com/a/cY6npUiW>.

Objectivos

Este é um trabalho aberto em que terão que desenvolver um **protocolo P2P** que permita aos amigos consultar as suas fotos. No início do programa, os amigos escolhem um dos amigos como nó central. Este nó central terá como papel extra ancorar a rede P2P para lá de permitir a partilha de ficheiros.

Para o utilizador deste sistema, o programa deverá incluir funcionalidades de:

- Listar todas as imagens em sistema (através do seu identificador)
- Obter uma dada imagem, com base no seu identificador

Deverão existir 2 scripts "executáveis": `daemon.py` (que executa a rede P2P) e o `client.py` que permite executar operações sobre a rede P2P.

Funcionamento

Cada `daemon.py` deve iniciar com um argumento: nome de pasta com as imagens iniciais do peer.

O `daemon.py` deve juntar-se à rede e analisar a pasta calculando o `imagehash` [1] de cada fotografia da sua pasta, assegurando que não existem imagens repetidas (mesmo conteúdo detectado pelo `imagehash`) para lá das estritamente necessárias para garantir redundância do sistema P2P. No caso de detectar que a imagem é repetida, então a imagem deve ser apagada da pasta inicial.

O `daemon.py` deve paralelamente manter comunicação com os restantes peers e com o `client.py` a partir do qual recebe comandos.

O `client.py` recebe por argumento o endereço do seu `daemon.py` (oportunisticamente o `daemon.py` deve imprimir o seu endereço quando inicia)

Avaliação

A avaliação deste trabalho será feita através da submissão de código na plataforma GitHub classroom e de um relatório em formato PDF com não mais de 5 páginas colocado no mesmo repositório junto com o código e com o nome ***relatorio.pdf***.

Está em avaliação o protocolo definido e documentado, assim como as *features* implementadas de acordo com os objetivos incrementais:

- Identificação unívoca de uma dada imagem
- Transferência de imagem entre nós
- Pesquisa por uma dada imagem pelo seu identificador
- Eficiência de armazenamento
 - Não existência de imagens repetidas (imagens iguais ou semelhantes do dataset devem ser consolidadas com o mesmo)
 - Uma dada imagem não deve estar armazenada em mais do que 2 nós
 - As imagens devem estar distribuídas pelas rede (evitando que alguns nós centralizem as cópias de imagens)
- Resiliência a falhas
 - Caso um dos nós se desligue o sistema deve assegurar que não há perda de fotos

Referências

[1] <https://github.com/JohannesBuchner/imagehash>

[2] <https://pillow.readthedocs.io/en/stable/>

[3] <https://filesender.fccn.pt/?s=download&token=675d4815-86ba-45da-b58d-d8be6c33ecc4>