

**Curso Técnico Superior Profissional em:** Tecnologias e Programação de Sistemas de Informação

**1.º Ano/ 2.º Semestre**

**Unidade Curricular:** Desenvolvimento Web - Back-End

**Data Entrega:** 14/05/2019

**Docente:** David Jardim

**Época:** Normal \_\_\_\_\_

### API EM REST PARA UM SITE AGREGADOR DE FOTOGRAFIAS

---

1. O projeto deverá ser feito no máximo por grupos de 2 elementos e submetido em .zip no formato **P1\_nºaluno1\_nºaluno2** no Moodle e os alunos deverão efetuar a defesa oral do projeto
2. Crie um ficheiro **photos.json** (1 valor)
  - a. Adicione várias entradas que representem uma fotografia ao ficheiro respeitando a formatação JSON. Cada fotografia deverá ter os seguintes atributos:
    - i. id - uuid
    - ii. uploader - string
    - iii. title - string
    - iv. description - string
    - v. size - number
    - vi. url - string
    - vii. likes – number
    - viii. dislikes - number
    - ix. comments - Array
    - x. tags – Array
3. Crie um ficheiro server.js (1 valor)
  - a. Configure e inicie um servidor em Express.js
  - b. Importe o módulo “fs” para manipular ficheiros em Node.js
  - c. Importe o módulo “body-parser” para efetuar o *parsing* do *body* adjacente aos pedidos HTTP
4. **PARTE A** - Implemente os seguintes *endpoints* REST utilizando os métodos HTTP adequados (9 valores)
  - a. Listar todas as fotografias existentes no ficheiro JSON e devolver na resposta. (1 valor)
  - b. Adicionar uma nova imagem e atualizar o ficheiro (gravando o mesmo). O ID terá que ser um *Universally Unique Identifier* (UUID) gerado de forma aleatória com a ajuda do módulo “uuid/v1”, pode ser necessário instalar o package. Deverá ser enviada uma mensagem de sucesso indicando que a fotografia foi adicionada. (2 valores)
  - c. Selecionar todas as fotografias de um determinado uploader e devolver essa lista na resposta. (2 valor)
  - d. Incrementar o número de likes e atualizar o ficheiro. Devolver entrada atualizada na resposta. (2 valores)
  - e. Listar todas as fotografias que contenham determinados tags e devolver a lista na resposta. (2 valores)

Cofinanciado por:

5. **PARTE B** - Implemente os seguintes *endpoints* REST utilizando os métodos HTTP adequados (9 valores)
- a. Selecionar apenas uma fotografia pelo seu ID e devolver a mesma na resposta. (1 valores)
  - b. Apagar uma fotografia existente e atualizar o ficheiro, indicar mensagem de erro se o ID da fotografia a apagar não existir ou mensagem de sucesso caso seja apagada. (2 valores)
  - c. Incrementar o número de dislikes e atualizar o ficheiro. Devolver entrada atualizada na resposta. (2 valores)
  - d. Adicionar comentários a uma determinada fotografia pelo seu ID e atualizar o ficheiro. Devolver a entrada atualizada na resposta. (2 valores)
  - e. Listar todas as fotografias ordenadas por ordem de likes e devolver a lista ordenada na resposta. (2 valores)

Cofinanciado por:

