



# Pràctica d'avaluació contínua

## UF1844: Creació d'un magatzem d'imatges

DATA  
D'ENTREGA

16 de JUNY DE 2020 23:59PM

## Introducció

Es demana crear un script en NodeJS, que permeti inserir un conjunt de camps que aportin informació sobre un recurs URL a una imatge. L'usuari ha de poder inserir diverses direccions URL: i un conjunt de camps addicionals. A més, l'aplicació ha de permetre veure totes les imatges inserides.

També es desitja que l'aplicació recopili i informi sobre certes característiques de les imatges, tal com es detalla en aquest projecte.

Els textos de l'aplicació poden estar en català, castellà o anglès.

## Funcionalitats desitjades

### Menú - 1 punt

Es demana un menú per poder navegar entre les diferents funcionalitats de l'aplicació

El menú de l'aplicació ha de tenir les següents característiques

- ~~1. Ha de tenir dos elements de menú, "Mostrar Imágenes" i "Añadir Imagen" 0.25 punts~~
- ~~2. Els dos elements han d'estar alineats horitzontalment 0.25 punts~~
3. Quan passem el ratolí per sobre, el fons l'opció de menú ha de canviar de color **0.25 punts**
4. S'ha de mostrar de manera ressaltada l'opció de menú on és trobem. Es pot fer servir negreta o en un altre color **0.25 punts**
  - a. Per exemple, si ens trobem a la pàgina "Mostrar Imágenes"; el ítem de menú "Mostrar Imágenes" ha de ressaltar per sobre de l'altre.

## Afegir nova imatge - 3 punts

Quan fem clic sobre “Añadir Imagen”; hem d'oferir a l'usuari la possibilitat d'afegir un nou URL que apunti un recurs que sigui una foto; a més d'uns quants camps addicionals. Tots els camps del formulari s'han de validar quan l'usuari faci clic al botó de *submit*. El formulari ha d'informar a l'usuari si hi ha cap error de validació. Es detalla tot a continuació:

- ~~1. L'usuari ha d'introduir un camp per posar un títol a la imatge. Aquest camp com a màxim pot tenir 10 caràcters;~~ i només pot incloure números, lletres o un guió baix<sup>1</sup> **0.75 punts**
- ~~2. L'usuari ha d'introduir una nou URL en un formulari. Aquest URL ha de ser vàlid<sup>2</sup>~~ **0.75 punts**
- ~~3. També és necessari introduir una data. Aquesta data és arbitrària. Per tant, l'usuari pot seleccionar el dia que vulgui. L'usuari ha de poder escriure la data de la imatge, o bé seleccionar la d'un calendari. Important: és suficient amb seleccionar el dia de la foto, no cal introduir l'hora ni els minuts.~~ **0.75 punts**
- ~~4. El formulari envia les dades al servidor mitjançant el mètode POST.~~ **0.25 punts**
5. Si l'URL ja existeix a la *base de dades* del servidor, hem de fer dues coses
  - NO afegir-la al magatzem d'imatges que disposem al servidor **0.25 punts**
  - Mostrar un missatge a l'usuari indicant que l'URL ja existeix al nostre magatzem, i per tant, no la podem afegir **0.25 punts**

---

<sup>1</sup> Guió baix: \_

<sup>2</sup> Només cal validar el format, no que el recurs URL sigui realment una imatge.

## Añadir imagen nueva

Url de la imagen

Envia

La URL <https://i.picsum.photos/id/866/200/300.jpg> ya existe en nuestra base de datos.

## Mostrar imatges emmagatzemades - 4 punts

Quan fem clic a “Mostrar Imágenes”; hem d’ensenyar a l’usuari totes les imatges guardades fins al moment


- ~~1. La pàgina mostra totes les imatges emmagatzemades fins al moment~~ **2 punts**
- ~~2. Per a cada imatge, es mostra el seu títol~~ **0.6 punts**
  - ~~a. El títol sempre es mostra en majúscules~~ **0.15 punts**
- ~~3. Per a cada imatge, es mostra la seva data~~ **0.75 punts**
4. Les fotos es mostren ordenades per la data que hem introduït al formulari per cada una d’elles. És a dir, a l’esquerra, o de dalt a baix en mòbils, primer hi ha d’aparèixer les imatges més recents en el temps. **0.5 punts**

No cal un disseny perfecte de la pàgina; però ha de quedar clar visualment que un títol i una data pertanyen a una imatge en concret.

## Analitzar el color predominant de cada imatge - 2 punts

Ens demanen una funcionalitat molt curiosa: volem analitzar quin és el color predominant de cada una de les fotos. El script NodeJS ha d’implementar ara aquesta funcionalitat

1. Investigar i Instal·lar un paquet amb NPM que permeti dur a terme aquesta funcionalitat **0.5 punts**

- 
2. La pàgina “Mostrar Fotos” mostra ara també una secció on apareix el color predominant de cada foto
    - a. Per a cada foto, mostrarem un quadradet, rectangle o qualsevol figura visible pintada del color predominant de cada una de les fotos **1 punt**
    - b. Dintre de cada figura de color, hi ha d’aparèixer la codificació del color textualment, sigui en notació RGB o hexadecimal. **0.5 punt**

No cal alinear cada foto amb cada color; és suficient el fet de mostrar un element visual per cada una de les fotos; i que visualment quedi clar que el color predominant pertany a una foto en concret.