React

S5. Acudits

Descripció

En els anteriors lliuraments, les dades que hem usat en les nostres webs els agregat nosaltres en el programa (hardcoded), però no sol ser l'habitual.

Com en la majoria de webs reals, consumirem les dades d'una API en aquest exercici. Per sort, no haurem d'implementar una API per a guardar les dades en una base de dades, i poder consumir-los amb una sèrie de crides. En lloc d'això, usarem una API ja feta que ens permetrà obtenir el llistat de naus fàcilment.

Els dos temes més importants que posaràs en pràctica en aquest projecte són Typescript i obtenció de dades mitjançant anomenades API Rest a un servidor.

Una empresa de coaching està portant un experiment en empreses de Barcelona, en la qual està mesurant l'impacte de l'humor i la diversió en la productivitat.

Ens han demanat una aplicació web que mostri acudits als empleats abans de començar la jornada laboral.

Seràs l'encarregat de dur a terme la base del projecte per a fer una demo en dues setmanes amb el client i comenrzar les proves amb usuaris reals.

Informació de API a consumir

Crearem una web d'acudits, consumint dades d'una API gratuïta que no requereix clau. Veuràs que és molt divertit i interessant poder obtenir dades d'una API, imagina la quantitat de webs que pots fer!

En l'àmbit professional, quan treballis en un projecte, l'empresa normalment té un backend amb una documentació per a poder obtenir les dades. A més, en moltes ocasions, una web o app no sols té una font de dades, també és comuna utilitzar APIs de tercers. Resumint, saber consumir dades d'una API és un dels skills més importants d'un programador frontend!

(En qualsevol projecte professional, consumiràs dades d'una API, per la qual cosa has d'entendre bé a fons com realitzar crides API i el asincronismo de javascript)

A continuació es mostren links i informació que poden ser d'utilitat per a implementar les crides API en la teva web:

- La documentació de la API a consumir és la següent:

https://icanhazdadjoke.com/api

- Cida per a obtenir un acudit:

https://icanhazdadjoke.com/

- Header per a obtenir les dades en el format que ens interessa:

'Accept': 'application/json'

Posem aquest header en l'anomenada API perquè el servidor sàpiga en què format volem les dades, en el nostre cas en JSON.

Ens hem inventat aquest header? no, ho hem extret de la documentació d'aquesta API. Les APIs les crear programadors de backend, deixant sempre documentat tot perquè els programadors de frontend o altres programadors backend d'altres empreses puguin consumir les dades.

Simplificant, diguem que la URL de la API que introduïm en aquest projecte és on consumirem les dades, i el header és com volem les dades.

Notes

Tens els següents indicacions del responsable frontend:

- És **obligatori implementar tots els bucles i lògica amb ES6** (usant map, reduce, filter i sort per a manipular arrays). En cap cas es podrà usar el bucle for.

- Encara que és molt recomentadble usar Typescript en aquest lliurament, si vas una mica just de temps pots usar javascript.

- Si tens dificultats per a crear de zero un projecte Typescript, aquí tines els passos:
-> Passos per a preparar un projecte Typescript

Nivell 1

- Exercici 1

En aquest primer exercici crearem la pantalla principal que mostrarà acudits a l'usuari.

El funcionament ha de ser el següent:

- En entrar no mostrarà cap acudit. Apareixerà el títol i el botó de següent acudit"

- En prémer el botó de "Següent acudit" es farà fetch a la API d'acudits i es mostrarà per consola l'acudit en qüestió.

Nota: En aquest exercici no és necessari maquetar la web, primer farem que funcioni per a passar a aplicar-li els estils.

Tip 1: usar promises o async/await per a esperar la resposta de la API

Tip 2: abans d'usar una API en el codi, és recomanable usar Postman o eines online per a provar la API, per exemple https://apitester.com/. A més de garantir que funciona, veuràs l'objecte que retorna, per a saber utilitzar-lo.

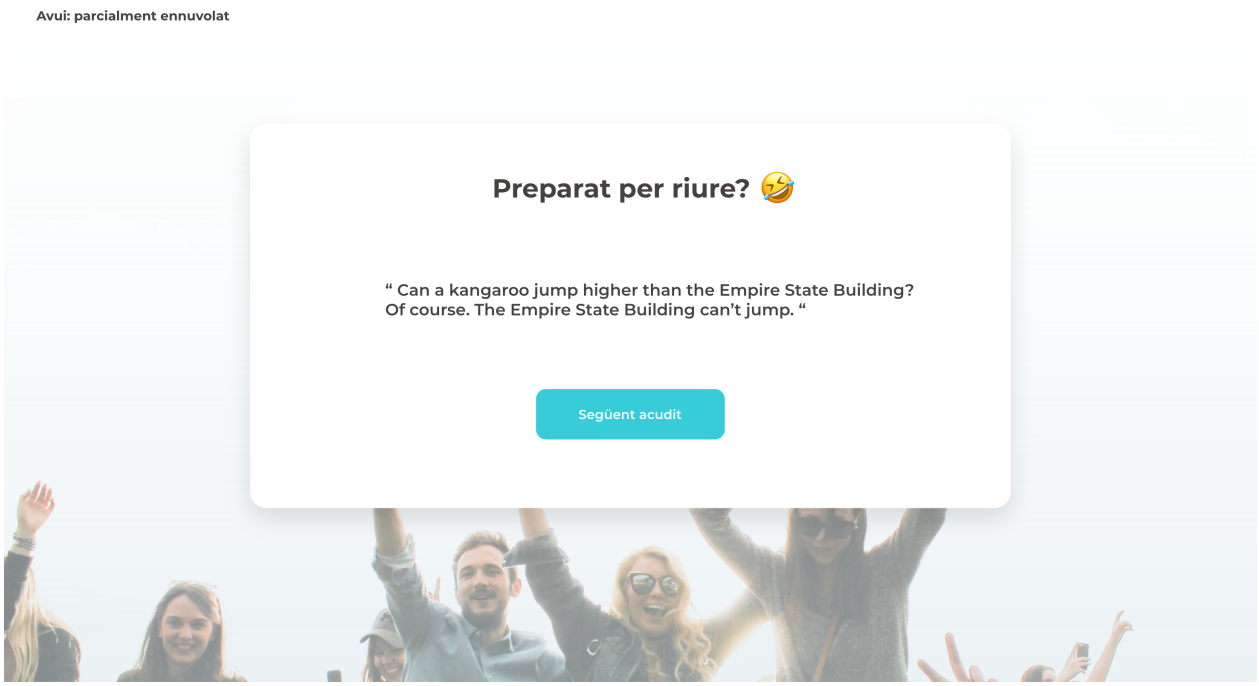
- Exercici 2

La nostra web ja obté els piuli del servidor i els vam mostrar per consola, falta mostrar-los-hi a l'usuari.

Realitzar una primera aproximació de la maquetació, col·locant cada element en el seu lloc. No et preocupis pels detalls, modificarem la maquetació més endavant.

L'objectiu d'aquest exercici és que l'usuari pugui anar visualitzant els acudits i demanar nous.

Una referència de la col·locació dels elements és la següent:



- Exercici 3

L'empresa que encarrega el projecte necessita fer un seguiment a l'ús d'aquesta web per al seu estudi.
Per a això, es necessita saber el nivell d'acceptació dels acudits, un tracking per a saber quan els empleats estan de més bon humor, i quants acudits es consumeixen de mitjana.

Com es tradueix aquesta petició en el nostre codi?

Necessitaràs generar un array anomenat reportAcudits, en el qual anirem guardant tota la informació relativa a l'acudit que ens demana el client.

Els tres camps que ha de tenir cada objecte del array són:

{

joke: "...",

score: 1,

date: ...

}

- La data de quan es va fer la valoració l'hauràs de guardar en format ISO. Més informació aquí.

- El camp score té un rang de l'1 al 3, sent un 1 la pitjor puntuació. Hauràs d'implementar 3 botons entre l'acudit i el. botó per a carregar el següent acudit, perquè l'usuari pugui puntuar-lo.

Amb la puntuació de l'acudit, juntament a l'acudit i generant una data, hauràs d'anar emplenant el array reportJokes.

Quan vagis actualitzant aquest array, amb mostrar per consola el seu contingut serà suficient

🌟 Nivell 2

- Exercici 4

Ben fet! Ja tens una web d'acudits operativa. Ja que està web està pensada per a mostrar acudits als usuaris a primera hora del matí perquè comencin bé el dia, afegirem informació meteorològica ja que els pot ser d'utilitat.

Consumir una API d'informació meteorològica i mostrar-ho en la web. Aquesta API ha de dir-se en l'obertura, no mitjançant un botó.

Nota: Encara no és necessari maquetar la web, amb mostrar una paraula que indiqui el temps és suficient.

- Exercici 5

El client ens ha comunidado en les primera proves, que els usuaris s'avorriran si sempre vam mostrar el mateix tipus d'acudits.

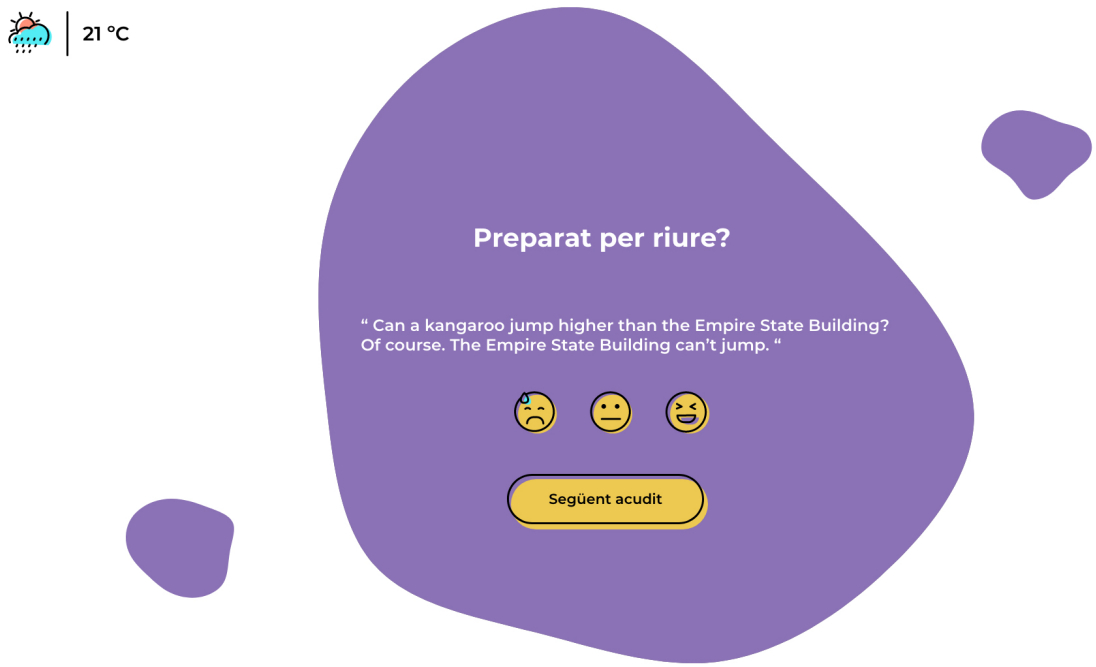
Has de buscar una altra API (o APIs) d'acudits i utilitzar-la per a alternar acudits de diferents fonts (bé alternant un de cada o de manera aleatòria).

Ajuda: Els acudits de Chuck Norris mai fallen.

Nivell 3

- Exercici 6

Maquetar la web d'acudits i temps meteorològic conforme a la següent pantalla:



Tens molts generadors en línia de gradients, formes, backgrounds... en aquest exemple es recomana usar aquesta eina onlione per a generar la base del contenidor dels acudits:

https://www.blobmaker.app/

Com hauràs pogut observar, en lloc de mostrar a l'usuari el text del temps, el traduïm en una icona i la temperatura actual.

Ajuda: la forma basi del contenidor d'acudits implemtarla amb svg.

Bonus: per a millorar l'experiència d'usuari, pots tenir guardades diverses formes del contenidor (svgs en una carpeta del projecte), i quan l'usuari demani un nou acudit, canviar la classe css del contenidor i amb això canviarà la forma (carregant un altre svg), sent superdinámica i cridanera la web!

Recordatoris

- Els sprints duren dues setmanes i comencen en dilluns.

- **És obligatori pujar tots els lliuraments almenys amb el nivell 1** al final del sprint per a poder passar al següent.

- Com a més tard el lliurament es farà el dilluns següent, dia que comença el nou sprint.

- **És recomanable intentar aconseguir les màximes estrelles possibles** en els exercicis. Si no et dóna temps, no et preocupis. És una manera de millorar el teu perfil de cara al procés d'ocupabilitat, però també es té en compte el feedback dels mentors).

Recursos

Objectius

- Afianzar l'après en els sprints de ES6, Typescript i Maquetació
- Consumir datos de un servidor mediante el uso de una API
- Mostrar al usuario los datos procedentes de la API

Durada: 4-6 dies

Lliurament: per pull request

Estat de la tramesa

Estat de la tramesa	Cap intent
Estat de la qualificació	Sense qualificació

Criteris de qualificació

Rubrica exercici M2. Classes & Arrow Functions

Exercici 1 (N.1) Es mostra un acudit en prémer el botó de següent.	No funciona *0 punts*	Funciona amb la implementació sol·licitada *0.75 punts*	Funciona sense ser implementada com l'enunciat indica *1.5 punts*
Exercici 2 (N.1) Maquetació bàsica.	No funciona *0 punts*	Funciona sense ser implementada com l'enunciat indica *0.75 punts*	Funciona amb la implementació sol·licitada *1.5 punts*
Exercici 3 (N.1) Marcador de punts per consola.	No funciona *0 punts*	Funciona sense ser implementada com l'enunciat indica *1 punts*	Funciona amb la implementació sol·licitada *2 punts*
Exercici 4 (N.2) Consultar API del temps.	No funciona *0 punts*	Funciona sense ser implementada com l'enunciat indica *1 punts*	Funciona amb la implementació sol·licitada *1.5 punts*
Exercici 5 (N.2) Segona API d'acudits.	No funciona *0 punts*	Funciona sense ser implementada com l'enunciat indica *1 punts*	Funciona amb la implementació sol·licitada *1.5 punts*
Exercici 6 (N.3) Maquetació amb blob.	No funciona *0 punts*	Funciona sense ser implementada com l'enunciat indica *1 punts*	Funciona amb la implementació sol·licitada *2 punts*

Darrera modificació -

Comentaris de la tramesa Comentaris (0)

Entregar

Encara no heu fet cap tramesa.

Next Activity

Unió Europea
Fons Europeu
de Desenvolupament Regional

Segueix-nos

Heu iniciat sessió com a Sara Planas Colmenero (Surt)