**Introducció a l'aplicació web**

**Hosting de l'aplicació**

**Introducció**

L'aplicació és accessible a través de la URL: *https://pfc-family-search.herokuapp.com/*

Per tal d'accedir a la zona d'exemples de l'aplicació cal identificar-se primer amb FamilySearch. Per fer-ho, cada professor necessitarà un compte de FamilySearch diferent. A l'apèndix [] s'inclouen diferents usuaris creats per cada un dels membres del tribunal, però cada un també pot crear el seu propi.

Hem escollit la plataforma Heroku per desplegar l'aplicació ja que oferia un ventall d'eines i documentació amplies ideals per un desenvolupador novici de la plataforma Node.js.

No obstant, no són només les facilitats pels nous desenvolupadors el que ens van portar a desplegar l'aplicació a Heroku, sinó també que el seu pla gratuït s'ajustava en gran mesura el que el projecte requeria i sempre existia la possibilitat d'augmentar la capacitat de processat necessària sota demanda.

Les característiques que més ens atreien de la plataforma Heroku són presentades en els següents apartats.

**Fàcil configuració**

Per una aplicació simple com la desenvolupada per aquest projecte, no fa falta canviar res en el codi d'una aplicació Node.js normal. Només hem d'incloure un fitxer a l'arrel del projecte amb el nom 'Procfile', que indica la comanda que s'ha d'utilitzar per iniciar l'aplicació. En el nostre cas, el fitxer 'Procfile' conte la línea de codi: '*web: node app.js'*

**Fàcil desplegament al núvol**

Heroku es troba molt ben integrat amb Github, l'eina que utilitzem per mantenir sincronitzats els desenvolupaments en les diferents estacions de treball. Gràcies a aquesta integració, Heroku disposa d'una comanda que ens permet afegir un remot de repositori a les nostres carpetes de desenvolupament.

*> heroku git:remote -a pfc-family-search*

Un cop tenim el remot de Heroku configurat, per desplegar l'aplicació al núvol només hem d'utilitzar la comanda:

*> git push heroku master*

Aquesta comanda indica que volem penjar a Heroku el contingut actual del nostre repositori.

**Entorn de proves local**

Apart del entorn de proves local que podem configurar mitjançant l'instalació de Node.js en el nostre sistema, Heroku també ens permet simular la nostre aplicació en un entorn de producció Heroku a nivell local mitjançant la comanda:

*> heroku local web*

Tot i que no aporta gaires diferencies respecte a desplegar l'aplicació en un entorn local mitjançant Node.js de la forma regular, pot esdevenir una eina útil sota certes circumstàncies.

**Decent versió gratuïta**

Durant tot el procés de desenvolupament l'aplicació es trobava sota el paquet gratuït ofert per Heroku.

Aquest ofereix les següents funcionalitats:

* Desplegament des de repositoris GIT.
* Actualitzacions automàtiques.
* Auto reparació d'aplicacions.
* Logs del sistema.
* Número de processos diferents suportats: 2
* 1000 hores mensuals de 'dyno' actius. L'aplicació s'adorm després de 30 minuts d'inactivitat.
* Dominis customitzables.
* 512MB de RAM

**Fàcil escalatge de l'aplicació**

En cas de que es desitgi millorar la capacitat de concurrència de l'aplicació per poder rebre més peticions i realitzar més processos diferents al mateix temps, aquesta és fàcilment escalable mitjançant la inclusió de nous 'dynos'.

Els 'dynos' són els contenidors que executen les comandes dels usuari. Per la nostre aplicació web bàsicament es necessiten 'dynos' que processin el tràfic HTTP i HTTPS. Gracies a que el nostre servidor és bastant simple (hem posat la lògica d'interacció de FamilySearch en la capa del controlador) és probable que no faci falta augmentar el nombre de 'dynos' inicial.

De totes formes, aquest és fàcilment escalable mitjançant una simple comanda sempre i quan el nostre pla contractat amb Heroku no sigui el bàsic. Si volem augmentar a 2 els 'dynos' disponibles, executaríem la següent comanda:

*> heroku ps:scale web=2*

**Requisits de l'aplicació**

Aquesta llista pretén oferir un tast dels requisits o manaments que s'han tingut en compte durant el desenvolupament de l'aplicació web.

**Requisits funcionals**

* La web ha de permetre identificar-se amb FamilySearch mitjançant el sistema de pop-up.
* La web ha de permetre a l'usuari tancar la connexió amb FamilySearch mitjançant una funcionalitat de 'Sign Out'.
* L'aplicació ha de ser capaç de tancar automàticament la connexió amb FamilySearch si aquesta expira.
* La web ha d'incloure una secció que ofereixi un petit resum del rerefons que va originar el projecte.
* La web ha de disposar d'una secció en la que s'enumerin i exposin les diferents propostes de projecte generades pels futurs estudiants.
* L'aplicació ha d'oferir la possibilitat de cercar persones en l'arbre familiar de FamilySearch i observar-ne els detalls d'alguna en concret.
* L'aplicació ha de permetre al usuari observar la evolució geogràfica d'un cognom donat un conjunt de països i període de temps.
* L'aplicació ha de permetre la visualització del nombre de naixements, casaments i defuncions enregistrades pe un país al voltant d'un any en concret.
* La secció d'exemples ha de ser només accessible si l'usuari es troba identificat a FamilySearch i ha rebut el token d'ús pertinent.
* En cas de que el token expiri, l'usuari ha de ser redirigit a la pàgina principal en el moment d'expiració o en la seva següent interacció si aquest es troba dins de l'àrea d'exemples.
* L'aplicació ha d'emmagatzemar el token proporcionat per FamilySearch que rep l'usuari en un recurs que no sigui accessible ni modificable per tercers.
* No es permetrà a l'usuari llençar dues crides contra la API de FamilySearch simultànies per la mateixa funcionalitat des de la mateixa fulla del navegador.

**Requisits no funcionals**

* L'aplicació web ha de funcionar i ser visualitzada de forma correcta en els principals navegadors web moderns d'escriptori.
* El codi de l'aplicació serà codi obert.
* Els formularis de l'aplicació web que puguin generar errors han de proporcionar informació a l'usuari en el moment que el camp és abandonat o si s'intenta enviar el formulari amb errors.
* La web ha de ser relativament fàcil d'utilitzar, oferint les eines necessàries als usuaris i facilitant la navegació per les diferents seccions.
* Mentre l'aplicació web espera resposta de la API de FamilySearch, s'ha de mostrar a l'usuari que l'aplicació es troba esperant resultats i el progrés realitzat fins al moment.
* L'aplicació ha de donar un clar feedback a l'usuari quan la interacció amb la API de FamilySearch finalitza.
* L'aplicació ha de ser navegable de forma acceptable mitjançant dispositius mòbil. Les integracions amb la API de FamilySearch també han de ser utilitzables, però no cal que la informació resultant es trobi completament adaptada a aquests dispositius.
* Les imatges de l'aplicació s'han de trobar optimitzades en la mesura que sigui possible per intentar que aquesta carregui el més ràpid possible en un entorn d'hostalatge gratuït.
* Les imatges principals de l'aplicació s'han de carregar de formar transparent quan l'aplicació és iniciada i la primera pàgina és carregada, per facilitar-ne la visualització quan l'usuari navega entre les diferents pàgines.
* La llengua utilitzada en el web és l'anglès per tal d'ajudar i facilitar el procés de certificació.
* El projecte s'ha de trobar sota una certificació Creative Commons d'atribució no comercial.
* La informació sobre el codi font del projecte, la llicència i la facultat d'informàtica ha de trobar-se disponible en el peu de pàgina de les pàgines de la web.
* Es podrà monitorar la navegació dels usuaris pel web, així com les seves accions principals i errors generats.
* Es podrà obtenir la configuració dels sistemes amb els que s'ha navegat per la web i veure si el comportament d'algun d'ells és més propici a la generació d'errors.
* Els fitxers Javascript s'han de trobar el més al final possible dels arxius HTML per facilitar la càrrega del contingut.
* És reutilitzarà codi HTML i Javascript en la mesura que sigui possible per tal d'evitar la duplicació de contingut.

**Estructurà de l'aplicació web**

**Introducció**

L'aplicació web és relativament simple pel que fa referència a la navegació i seccions d'aquesta.

La figura [] mostra l'arbre de continguts accessibles. Cal indicar que aquesta figura no representa les úniques rutes de navegació possibles entre les diferents seccions, sinó un breu mapa del contingut total disponible a través del website.

[]

A continuació explicarem en més detall en que consisteix cada una de les pàgines de la nostre aplicació web, però primer volem presentar l'estructura general que presenten cada una de les pàgines del web.

La figura [] mostra l'esquema bàsic que les pàgines segueixen. Aquest pot ser descrit o desglossat en les següents seccions:

1. **Barra de navegació:** Permet desplaçar-se per les diferent seccions principals.
2. **Capçalera:** Conté el títol i subtítol de la pàgina sobre impressionat a una imatge relacionada.
3. **Contingut principal:** El contingut principal i únic d'aquesta pàgina.
4. **Footer:** Peu de pàgina. Inclou informació sobre el codi font del projecte, la llicència i la facultat d'informàtica de Barcelona.

**Home o pàgina principal**

La home és la primera pàgina que veu l'usuari quan entra a l'aplicació web. Aquesta no te cap altre propòsit que el de donar la benvinguda i enllaçar als diferents continguts.

El bloc de contingut principal de la pàgina inclou enllaços i breus descripcions a les tres seccions principals del projecte. A saber, rerefons, propostes de projecte i exemples d'implementació.

La principal diferència entre la versió per dispositius mòbils i la d'escriptori és que la primera fa desaparèixer el subtítol, simplifica el títol, transforma la barra de navegació en una d'usable per dispositius mòbil i transforma la secció de contingut principal apilant-ne el contingut de cada una de les tres columnes que la conformen.

**Rerefons**

La pàgina de rerefons conté la informació bàsica respecte l'origen, el context i les motivacions que ens van portar a realitzar aquest projecte. Consisteix en un breu resum d'alguns dels apartats de la primera secció de la memòria.

El bloc de contingut principal d'aquesta pàgina consisteix en dos grans blocs de text. El primer descriu el rerefons del projecte mentre que el segon ofereix una petita descripció del context i motivacions.

La principal diferència entre les versions d'escriptori i mòbil és que la segona presenta un títol més simple, la desaparició del subtítol i una barra de navegació adaptada a dispositius mòbils. L'estructura del contingut roman igual, això si, adaptada a la grandària del dispositiu que la conté.

**Propostes de projecte**

L'apartat de propostes de projecte recull les diferents propostes, que s'han generat per servir com a futurs projectes finals de carrera per a estudiants d'informàtica, en la mateixa secció.

El bloc de contingut principal per aquesta pàgina consisteix en dos blocs composats per petites caixes que contenen una imatge, un títol i una petita descripció de la proposta que representen. Cada una d'aquestes caixes enllaça també amb una pàgina que conté els detalls específics de la proposta.

El primer bloc de caixes representa les propostes generades pels futurs estudiants, mentre que el segon bloc està format per les propostes relacionades amb els exemples implementats.

Les principals diferencies entre les versions d'escriptori i dispositius mòbils, és que la segona presenta un títol simplificat, la desaparició del subtítol, la barra de navegació adaptada, i diferent nombre de caixes per fila segons el dispositiu utilitzat. Tres columnes per escriptoris, dues per tauletes gràfiques i una per mòbils.

**Detalls específics d'una proposta**

**Heroku**https://devcenter.heroku.com/