

# Aplicació web interactiva de mobilitat a la comarca del Berguedà

Memòria de Projecte Final de Grau

**Grau en multimèdia**

Enginyeria Web

**Autor: Ricard Codina Codina**

Consultor: Jordi Ustrell Garrigos

Professor: Joan Soler Adillon

06/01/2020



Aquesta obra està subjecta a una llicència de [Reconeixement-NoComercial-SenseObraDerivada 3.0 Espanya de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

*A la Lluïsa, al Josep i a la Núria, que sempre han estat  
al meu costat donant-me suport, ajudant-me en el què  
ha calgut i fent la vida una mica més senzilla.*

## Abstract

L'objectiu d'aquest projecte és la creació d'un portal web que conté tota la informació referent al transport públic de la zona de la comarca del Berguedà, a Catalunya. L'usuari pot realitzar cerques de punt a punt, consultar mapes interactius i veure informació sobre les parades, línies i avisos de les circulacions de cada moment. A més, cada usuari pot enviar informes d'incidències per alertar a la resta d'usuaris de l'aplicació.

Pel desenvolupament de l'aplicació, s'ha habilitat i configurat una base de dades MySQL, s'ha programat un backend en llenguatge PHP que actua com a API REST generant arxius en format JSON a partir de les dades de la base de dades i, finalment, s'ha elaborat el frontend utilitzant el framework ReactJS, en plena expansió en l'actualitat, combinat amb CSS i Javascript (ECMAScript 6). A més, s'han integrat APIs i recursos externs de tercers, com ara, Google Maps i el resultat es presenta en forma de Progressive Web App (PWA).

Paraules clau: aplicació web, mobilitat, transport públic, generador d'itineraris, mapa interactiu, Berguedà, MySQL, PHP, CSS, Javascript, Google Maps, Progressive Web App.

## **Abstract (english version)**

The main objective of this project is the construction of a web application with all the information about mass transit in the Berguedà region, in Catalonia. Users can get directions from A to B, consult interactive maps and look for information about stops, lines and updates about disruptions in current services. Moreover, each user can send travel alerts to the rest of the app's users.

Talking about the application development, a MySQL database has been enabled and configured, a PHP language backend has been set up working as an API REST generating a response in a JSON form based on the database data, and, finally, the frontend has been created using ReactJS' framework (currently growing across the Internet), CSS and Javascript (ECMAScript 6). In addition, some third-party API and plug-ins are used, such as Google Maps, and the result is revealed in a Progressive Web App (PWA) form.

Keywords: webapp, web application, mobility, mass transit, public transportation, directions generator, interactive map, Berguedà, MySQL, PHP, CSS, Javascript, Google Maps, Progressive Web App.

## Agraïments

Agraeixo l'ajuda dels amics i familiars que m'han facilitat quan els hi he demanat pel que fa a les proves tècniques i pràctiques que s'han portat a terme en aquest projecte.

Agraeixo, també, a Bífidus Produccions la cessió dels seus servidors i recursos tècnics per poder allotjar la plataforma de desenvolupament durant la fase de producció.

Finalment, dono les gràcies a tots l'equip de professors i consultors de la Universitat Oberta de Catalunya, amb una especial menció als qui han portat aquest projecte, per haver-me convençut a ser més ambiciós en el projecte, haver-me ensenyat tot tipus de coneixements i a fer-me el seguiment del treball per obtenir un bon resultat.

# Índex

1. Introducció .....	1
2. Descripció .....	4
3. Objectius .....	7
3.1 Principals .....	7
3.2 Secundaris .....	7
4. Escenari .....	8
4.1 Projectes actuals .....	8
5. Continguts .....	11
5.1 Portada .....	11
5.2 Motor de cerca .....	12
5.3 Mapa .....	12
5.3.1 Capes .....	13
5.3.2 Parades, línies i serveis actuals .....	13
5.4 Parades .....	13
5.5 Línies .....	14
5.6 Expedicions .....	15
5.7 Sistema d'avisos .....	15
5.8 Tauler d'administració .....	16
6. Metodologia .....	17
6.1 Desenvolupament del codi .....	17
6.2 Disseny de la interfície .....	17
6.3 Recopilació de la informació .....	18
7. Arquitectura de l'aplicació .....	19
7.1 Servidor .....	19
7.2 Base de dades .....	20
7.2.1 Sistema i llenguatge .....	20
7.2.2 Estructura .....	21
7.3 <i>Backend</i> .....	22
7.3.1 Llistat d'APIS .....	22
7.4 <i>Frontend</i> .....	24

7.5 Client .....	25
7.5.1 Dominis.....	25
7.5.2 Navegador web.....	25
7.5.3 Connexió a internet.....	26
8. Plataforma de desenvolupament.....	27
8.1 Software .....	27
8.1.1 Programació web.....	27
8.1.2 Frameworks .....	27
8.1.3 Programari per a la programació del projecte.....	29
9. Diagrames UML .....	31
10. Planificació.....	33
11. Procés de treball/desenvolupament .....	37
11.1 Preparació del servidor .....	37
11.2 Creació de la base de dades.....	37
11.3 <i>Backend</i> : creació de la REST API.....	37
11.3.1 Creació de classes.....	38
11.3.2 Creació de les peticions .....	39
11.3.3 Petició GET d'itinerari.....	40
11.3.4 Petició POST dels avisos d'usuari.....	41
11.4 <i>Frontend</i> : Disseny de la interfície.....	42
11.4.1 Components.....	43
11.4.2 Estats.....	43
11.4.3 Propietats.....	44
11.4.4 Ús dels estats i propietats .....	45
11.4.4 Navegació .....	46
11.4.5 Descripció dels components .....	46
11.4.6 Estil i aparença de l'aplicació .....	48
11.4.7 Progressive Web App .....	50
12. APIs i llibreries externes utilitzades .....	51
12.1 API de Google Maps.....	51
12.2 Encoded Polyline .....	52
12.3 React-google-maps.....	52



12.4 Axios.....	53
12.5 React-bootstrap .....	53
12.6 React-Router-Dom.....	54
12.7 Fontawesome .....	55
12.8 Moment.....	55
12.9 Google Analytics.....	55
13. Prototips.....	57
13.1 Wireframes de baix nivell .....	58
13.2 Wireframes d'alt nivell.....	61
14. Perfils d'usuari .....	64
15. Usabilitat.....	65
15.1 Mapa del lloc.....	66
16. Seguretat .....	67
16.1 Peticions a l'API.....	67
16.1.1 Control d'accés HTTP (CORS).....	67
16.1.2 Autenticació JWT .....	68
16.1.3 SQL INJECTION.....	69
16.1.4 Neteja de la informació rebuda.....	69
16.1.5 Validació dels camps d'usuari .....	70
16.2 Còpies de seguretat.....	70
16.3 HTTPS.....	70
17. Tests.....	72
17.1 Tests d'usuari .....	72
17.2 Tests d'usabilitat i seguretat.....	73
18. Versions de l'aplicació.....	75
19. Requisits d'instal·lació.....	76
19.1 Requisits del servidor.....	76
19.2 Requisits del client .....	76
20. Instruccions d'instal·lació .....	77
20.1 Instal·lació de la base de dades.....	77
20.2 Instal·lació i execució del <i>backend</i> .....	77
20.3 Instal·lació i execució del <i>frontend</i> .....	78

21. Instruccions d'ús .....	79
21.1 Cerca d'un itinerari.....	79
21.2 Consulta de la informació d'una parada .....	79
21.3 Consulta dels horaris d'una línia .....	79
21.4 Indicar un avís o incidència .....	80
21.5 Accedir a l'administrador.....	80
22. Bugs.....	81
23. Projecció a futur .....	82
24. Anàlisi de mercat.....	83
24.1 RATP .....	83
24.2 Transport for London .....	84
24.3 Mapa Swiss Railways Network .....	85
24.4 TMB Maps .....	85
24.5 Assistant SNCF.....	86
25. Conclusions .....	88
Annex 1. Lliurables del projecte .....	90
Annex 2. Captures de pantalla.....	91
Annex 3. Llibre d'estil .....	94
Annex 4. Tests d'usuari.....	100
Annex 5. Informe del test d'usabilitat.....	104
Annex 6. Bibliografia .....	122
Annex 7. Vita .....	124

# Figures i taules

## Índex de figures

Figura 1 Exemple d'un recorregut a través del Mou-te .....	9
Figura 2 Exemple de cerca a l'aplicació de l'Aquí Berguedà.....	9
Figura 3 Pàgina d'informació del transport a la demanda, on la informació està en enllaços .....	10
Figura 4 Captura de la portada de l'aplicació .....	11
Figura 5 Representació esquemàtica de l'arquitectura del servidor en el moment de publicació de la memòria..	20
Figura 6 Components que formen part de la portada de l'aplicació .....	24
Figura 7 Diagrama UML de la base de dades.....	31
Figura 8 Diagrama UML del <i>backend</i> .....	32
Figura 9: Línia del temps de les fases del projecte .....	35
Figura 10 Captura del programa Postman amb una petició GET a la REST API .....	38
Figura 11 Funció <code>obteDades()</code> de la classe <code>Linia</code> .....	39
Figura 12 Capçaleres de la petició GET de la informació d'una parada .....	39
Figura 13 Mètode <code>afegir()</code> de la classe <code>AvisUsuari</code> .....	42
Figura 14 Components i llibreries externes que carrega el component que genera la pàgina d'una línia.....	43
Figura 15 Mètode <code>obteDades()</code> del component <code>parada</code> . Es veu la crida a l'API a través d' <code>axios</code> i, si aquesta ha resultat satisfactòria, s'assignen les dades rebudes a l'estat del component.....	44
Figura 16 Estat del component <code>AvisosUsuari</code> . Disposem d'un mètode per alterar-ne el seu valor, el mètode s'assigna a un botó. Quan canvia l'estat, apareix o desapareix la finestra que conté un dels subcomponents. ...	44
Figura 17 Funcionament dels estats i les propietats al component <code>Resultats</code> . Els estats que controlen els diferents subcomponents estan en el component <code>Resultats</code> . Aquest estats es transmeten a través de les propietats. En alguns casos, només transmetem la informació que volem mostrar en cada moment (el mapa només mostra l'itinerari seleccionat en cada moment). A la llista de resultats és on l'usuari pot escollir quin itinerari vol veure en cada moment, per això li passem el mètode <code>setItinerariVisible()</code> que modificarà l'estat del component <code>Resultats</code> , ja que actua sobre l'estat del component des d'on es transmet, no en el que s'aplica. Com que l'estat canvia, la informació que es transmet també, per tant, el mapa i el component de detall, mostraran una nova informació, només la del nou itinerari seleccionat. ....	45
Figura 18 Diferents rutes que conté el component <code>Main</code> per tal de generar cada pàgina concreta a partir de la URL passada .....	46
Figura 19 Funció <code>dibuixaFilaHorari</code> del component <code>TaulaHorari</code> que genera la informació de l'horari de cada parada per implementar-la a la taula horària del component <code>Linia</code> .....	47

Figura 20 Codi de l'estructura de la portada. Bootstrap adapta l'ordre de les columnes segons l'amplada disponible.....	49
Figura 21 Variables del full d'estil CSS .....	49
Figura 22 Línia de codi que carrega les tipografies des de Google Fonts .....	50
Figura 23 En funcionar com a PWA, l'aplicació mostra a l'usuari un missatge per instal·lar l'aplicació, que queda integrada amb la resta d'aplicacions del telèfon i funciona com a tal.....	50
Figura 24 Funció consultaLloc() que retorna una Array amb la informació del lloc passat com a atribut. Es pot apreciar la construcció de l'URL de l'API. ....	51
Figura 25 Expressió dins del component de react-google-maps per generar els marcadors que representen les parades.....	53
Figura 26 Creació de la biblioteca d'icones de Fontawesome, on s'importen i s'afegeixen al contenidor de la biblioteca.....	55
Figura 27 Un dispositiu consulta l'aplicació i es pot seguir en temps real el seu comportament.....	56
Figura 28 Mapa de l'aplicació web .....	66
Figura 29 Funció afegir de la classe AvisUsuari on es veu l'ús de bindParam en una ordre SQL.....	69
Figura 30 Informació del certificat HTTPS entre el servidor de l'aplicació i el domini .....	71
Figura 31 Versió preliminar de l'aplicació on les icones d'una parada i un bus en circulació són molt similars ....	73
Figura 32 Resultats dels tests d'usabilitat i seguretat, per la versió d'escriptori (esquerra) i de dispositius mòbils (dreta) .....	74
Figura 33 Portada del lloc web de la RATP on el buscador d'itineraris se situa per sobre de la capçalera de la pàgina .....	83
Figura 34 Exemple del gràfic d'afluència de passatge a l'estació de Moorgate .....	84
Figura 35 Vista del mapa a la zona de Berna .....	85
Figura 36 Menú desplegable del mapa de TMB amb només algunes línies seleccionades .....	86
Figura 37 Captures de pantalla de l'aplicació Assistant SNCF on es veuen les tres pantalles dels avisos d'usuari .....	87

## Índex de taules

Taula 1 Característiques del servidor.....	19
Taula 2 Navegadors principals compatibles amb ECMAScript 6 .....	25
Taula 3 Procediments per anomenar les diferents versions de l'aplicació .....	75

# 1. Introducció

Durant els darrers anys, el sistema de transport públic a la comarca del Berguedà ha anat creixent amb l'aparició de noves línies d'autobús entre municipis, l'ampliació dels serveis a les rutes ja existents i la introducció de noves formes de prestació del servei, com el transport a la demanda o el bus urbà de la ciutat de Berga. Aquests fets, i el descens generalitzat del preu per trajecte, ha contribuït a l'augment de la demanda dels diversos mètodes de transport públic a la comarca que s'ha constatat en els últims anys i que s'espera que es vagi mantenint en el futur.

Fixant-se en les xifres publicades per l'Autoritat del Transport Metropolità al seu estudi TransMet de 2018<sup>1</sup>, el darrer disponible, es veu que durant aquest any es van efectuar un 14% més de viatges que durant l'any anterior, el 2017. Tot i que, en números absoluts, la xifra de viatges pot semblar petita (200.000 viatges registrats), si es compara amb els valors que es registren a l'entorn més immediat a Barcelona, representen uns números considerables per a una comarca de gairebé 40.000 habitants<sup>2</sup>.

No obstant, aquesta evolució en l'increment del servei i l'augment del passatge no han anat acompanyats d'una millora de la informació que rep el passatger. Ara per ara, l'usuari ha de regirar diverses pàgines web per trobar l'hora en que passa el següent bus o quina és la parada que té més pròxima. A més, aquestes webs, a vegades, solen ser incompletes, desactualitzades o poc entenedores.

Ara bé, l'interès per trobar aquesta informació hi continua sent ben present. Si s'agafen les dades d'una web que ofereix un apartat per consultar els horaris de l'autobús, la de l'Aquí Berguedà<sup>3</sup>, es troba que s'han registrat una mitjana de 1.200 visites mensuals durant els primers 8 mesos de l'any 2019<sup>4</sup>. Ara bé, si s'analitza la web, cosa que s'ampliarà en apartats posteriors d'aquesta memòria, es veu que ha estat dissenyada amb tecnologies senzilles, obsoletes i poc interactives de cara a l'usuari.

En un context d'emergència climàtica, la societat hauria d'apostar per l'ús de mètodes per desplaçar-se que siguin respectuosos amb el medi ambient i que aconseguixin treure emissors de CO<sub>2</sub> dels carrers i carreteres del nostre país. S'ha d'intentar dotar a les persones d'eines que els permetin facilitar els seus desplaçaments, ja sigui amb la introducció de nous mètodes de transport, la millora dels existents, o la facilitat d'accés a

<sup>1</sup> (Autoritat del Transport Metropolità, 2018)

<sup>2</sup> 39.104 habitants (Institut d'Estadística de Catalunya, 2018).

<sup>3</sup> <https://www.aquiberqueda.cat/serveis/autobusos/>

<sup>4</sup> Dades Google Analytics Gener-Agost 2019, total 9.650 visites registrades.

tots els detalls i informacions per poder realitzar el seu desplaçament quotidià amb transport públic comparat amb el quin efectua actualment, que podria ser més contaminant.

I és que, quan es parla d'informació, en l'actualitat, una de les principals fonts per trobar-la és a través de la xarxa. Una xarxa que ha conegut un munt d'evolucions en els darrers temps. Lluny queden les pàgines web fetes amb llenguatge html pur, plegades d'enllaços i amb problemes de compatibilitat entre navegadors, fent que els programadors gastessin molt de temps per poder aconseguir que un usuari que utilitzava Internet Explorer veiés el mateix que un que feia servir Netscape. Per no dir, la gran varietat d'estructures i dissenys que hi havia, cadascuna a gust i coneixements del desenvolupador del portal.

El cert és que, ara, es disposa de l'oportunitat de treballar amb estàndards i noves característiques que s'han anat afegint. Molts dels navegadors antics han desaparegut<sup>5</sup>, i la pràctica totalitat dels actuals interpreten de la mateixa forma aquests estàndards. Això representa un clar guany de temps a l'hora de programar i, a més, també ha facilitat la creació de pàgines web més segures, accessibles i estables.

A part, ha aparegut un grau d'interactivitat que no es tenia en els llocs webs primitius. L'usuari passa de només consultar informació a ser-ne partícip, a ser el generador de contingut. D'això se'n troba un clar exemple a les xarxes socials, com Facebook o Twitter, on veiem que és l'usuari el qui escriu un text, penja una foto, comparteix un enllaç... que serà el que rebran la resta d'usuaris, que també tenen l'opció de filtrar de qui i de qui no volen rebre informació.

A tot això, s'hi ha de sumar les diferents eines que han aparegut per facilitar el desenvolupament de nous llocs webs. Si s'agafa la part de programació, han sorgit noves llibreries i biblioteques de components, anomenats *frameworks*, que ajuden a escriure codi web a partir de diferents mòduls ja establerts, i així poder arribar a crear nous webs de forma més senzilla i amb un codi més estructurat i fàcil de comprendre per als altres desenvolupadors que siguin coneixedors del *framework* utilitzat. El procés que s'utilitza és molt similar al que es fa servir per fer aplicacions per a un ordinador per això les anomenem aplicacions webs.

Per la part del disseny, aquestes convencions entre desenvolupadors també han comportat l'arribada de diferents pautes de disseny per tal d'unificar criteris a l'hora de preparar una interfície per a aquestes aplicacions web. Això és beneficiós tant per al programador com

<sup>5</sup> Hi ha alguns navegadors que no han incorporat els nous estàndards i s'han acabat quedant obsolets (com Internet Explorer) o, fins i tot, han desaparegut (com el cas de Netscape). Igualment, cal indicar que no tots els navegadors han incorporat les noves característiques al mateix ritme i això pot provocar problemes de compatibilitat sobretot, si es fan servir característiques recents i poc implementades (The Evolution of the Web, 2012).

per l'usuari. Per una banda, el disseny de les aplicacions es pot fer en base a unes pautes, i així, evitem gastar més recursos temporals i econòmics en la creació d'un disseny específic per a l'aplicació. Només cal centrar-se en les particularitats que es mostraran en pantalla. Per la part de l'usuari, és important tenir en compte que tant sols només hagi utilitzat una pàgina que segueixi unes pautes de disseny concretes ja estarà familiaritzat amb la interfície de la resta d'aplicacions que facin servir les mateixes pautes, d'aquesta manera es millorarà l'experiència de l'usuari en una aplicació.

Un altre aspecte que també ha anat *in crescendo* durant els darrers anys ha estat el canvi de la forma on es consumeixen aquests llocs o aplicacions web. Òbviament, es refereix al consum d'internet a través de dispositius mòbils, els anomenats *smartphones*, que han provocat una democratització de l'accés a Internet i la possibilitat de realitzar-ne la connexió des de gairebé qualsevol punt del planeta. Això és reflecteix en les dades, ja que diversos estudis<sup>6</sup> apunten a que més de la meitat de peticions de llocs webs ja es realitzen a través de dispositius mòbils.

Això comporta que, ara mateix, no només s'ha de plantejar les aplicacions web per tal que siguin vistes des d'un equip d'escriptori, normalment amb una pantalla gran i en format quadrat o horitzontal, si no que, molt probablement, la major part de les consultes ens arribin des de dispositius que disposen d'una pantalla molt més petita i en format vertical. Per tant, s'ha de tenir la capacitat d'adaptar-se a aquest escenari de múltiples dispositius que no pas discriminar-ne algun i deixar-lo de banda.

La seguretat d'aquestes aplicacions també és un punt que està a l'ordre del dia a l'hora de referir-se a aspectes relacionats amb la enginyeria web. És primordial tenir en compte la forma en que està protegit el servidor web on s'allotja la nostra aplicació contra possibles atacs o incidències, ja sigui a causa d'un atacant maliciós, o *hacker*, o algun altre factor extern més difícil de controlar, com podria ser una catàstrofe mediambiental.

Aquest projecte, la memòria del qual teniu entre mans, vol satisfer tots aquests punts exposats en forma d'aplicació que ajudi als usuaris potencials del transport públic a la comarca del Berguedà a realitzar el seu viatge de forma senzilla i fiable, alhora d'estar informats de possibles incidències o alteracions al servei. Una oportunitat per implementar els nous estàndards i recursos web en una eina que pugui donar un benefici a la comunitat, amb l'ambició de tenir més recorregut un cop finalitzada la creació de la mateixa.

<sup>6</sup> Als Estats Units, el 58% de les visites als llocs web es van realitzar a través de dispositius mòbils (Perficient Digital, 2018), una situació majoritària que es dona des del 2016, quan les consultes des del mòbil van superar a les fetes des d'equips d'escriptori (StatCounter, 2016).

## 2. Descripció

El projecte que planteja aquest treball de final de grau vol resoldre els múltiples problemes que es plantejaven en l'apartat anterior amb la creació d'una plataforma que englobi tota la informació necessària per desplaçar-se utilitzant el transport públic per la comarca del Berguedà. El treball planteja la creació d'aquesta plataforma en forma d'aplicació desenvolupada utilitzant tecnologies web per tal que sigui accessible i consultada des de qualsevol lloc i dispositiu.

L'aplicació satisfà totes les necessitats d'informació detectades però mostrar-les d'una forma interactiva, i moderna aprofitant les oportunitats i funcionalitats que ens ofereixen els llenguatges de programació web més actuals. Per aquest motiu, de cara a l'usuari, les característiques de l'aplicació són les següents:

- Motor de cerca que permet a l'usuari obtenir itineraris entre dues poblacions o dues direccions concretes que especifiqui. Aquest buscador fa ús de totes les dades d'horaris i recorreguts de les línies per oferir la millor opció de recorregut per l'itinerari, dia i hora especificats, tenint en compte que hi ha trajectes que requeriran més d'una línia per a ser completats i que els seus horaris no són els mateixos entre un dia laborable i un dia festiu.
- Visualització d'un mapa a pantalla completa que mostra totes les poblacions, carrers i carreteres de la comarca, amb possibilitat de modificar el nivell d'ampliació del mateix o desplaçar-lo en la direcció que decideixi l'usuari.
- Geolocalització de l'usuari sobre el mapa amb llistat de parades i oferta de transport públic més pròxims a ell.
- Visualització de les línies i parades de transport públic sobre el mapa amb capacitat per mostrar i ocultar només les que l'usuari vulgui. Cada parada i línia ofereix una descripció de les mateixes amb les properes sortides o l'horari d'un dia concret.
- Representació sobre el mapa de la posició aproximada dels vehicles segons el seu horari teòric.
- Alertes sobre el servei, que introdueix l'administrador, i que poden afectar a una estació, a una línia o un servei concret.
- Alertes que poden generar els propis usuaris, alertant de retards o incidències d'un servei concret. Així l'usuari també té l'opció de contribuir a la informació que es mostra a la pàgina.



- Panell d'administració, des d'on es poden modificar els continguts de l'aplicació dirigit a usuaris sense el coneixement tècnic suficient per poder-los modificar directament des de la font.

El projecte també serveix per aprofitar totes les noves opcions que ens ofereixen les noves eines que han aparegut recentment en el camp de la programació web. D'aquesta manera es vol aconseguir una aplicació amb més funcionalitats, i un millor rendiment i potència. Per tant, l'aplicació està dissenyada i programada satisfent els següents punts:

- S'executa en un servidor web, que pot oferir l'aplicació a través del navegador web i, per això, és consultable des de qualsevol dispositiu que tingui aquest programari i connexió a internet, però també és possible executar-la en un entorn *offline*.
- S'aprofundeix en l'ús de les pautes web o *frameworks* que faciliten l'escriptura de codi web. En concret, s'utilitzarà PHP per al *backend*, generant una API REST per a l'obtenció del contingut i ReactJS per al *frontend*. Tot això, combinat amb HTML, CSS i Javascript per a la creació de la interfície.
- S'avalua l'opció de convertir l'aplicació en Progressive Web App (PWA), en clara expansió actualment en detriment de les aplicacions natives que s'instal·len als dispositius mòbils.
- Les consultes entre la base de dades i la web s'efectuen de forma asíncrona, així no suposa una discontinuïtat de la pàgina de cara a l'usuari.
- Aprofita recursos que ofereixen diversos serveis externs de tercers a través de les anomenades API. Aquest és el cas de la visualització del mapa o l'obtenció d'itineraris específics entre dos punts que no pugui calcular el sistema.
- És consultable en tot moment (disponibilitat) i des de qualsevol dispositiu (ordinador, mòbil, tauleta,...) amb totes les opcions, és a dir, que des de qualsevol aparell es poden realitzar totes les funcions.
- Està preparada per a futures millores i ampliacions, permetent la reutilització del codi i facilitant el seu manteniment gràcies als comentaris al codi i les possibilitats d'endrega que ens permeten els llenguatges de programació orientats a l'objecte.
- Té present la compatibilitat entre navegadors, tenint especial cura en que l'ús de tecnologies recents pot comportar que els navegadors i programes de consulta actuals encara no les admetin.

Amb tots aquests punts esmentats, aquest projecte vol ser un treball de professionalització on es posen en pràctica tots els recursos que tenim a l'abast en el camp de l'enginyeria

web, presentant atenció a les darreres innovacions i millores que hi ha hagut en aquest àmbit. A més, i de forma transversal, també es tracten aspectes relacionats en els camps de:

- Disseny gràfic i disseny d'interfícies multimèdia, per l'elaboració de la interfícies de l'aplicació, sobretot en la creació de les pautes i maquetes per desenvolupar-la (*wireframes*, llibre d'estil...).
- Arquitectura de la informació i disseny de bases de dades, per classificar tot el contingut de forma correcta a l'aplicació, per tal que la programació sigui més senzilla i efectiva.
- Fotografia digital i tractament d'imatge, pel que fa a la composició d'imatges i ús dels aparells de fotografia per a la realització de les possibles imatges que pugui requerir l'aplicació.
- Gestió de projectes, durant tot el procés d'elaboració i finalització del treball, en aplicar les metodologies i processos de realització de projectes.
- Seguretat i qualitat en servidors web, pel que fa a la protecció del servidor web on estarà emmagatzemada la web i a la prevenció de possibles atacs.

Finalment, s'elabora la memòria que conté tota la informació relativa a objectius, metodologia i planificació de treball del projecte, els recursos emprats per la seva elaboració, una descripció dels passos més destacats en la producció del mateix, el seguiment del llançament de l'aplicació i un anàlisi dels punts més destacats en matèria de seguretat i característiques del servei resultant. A més, s'inclouen com a lliurables el codi font de l'aplicació i presentacions en format vídeo com a introducció a l'aplicació.

# 3. Objectius

## 3.1 Principals

- Aprendre a desenvolupar una aplicació web utilitzant *frameworks* i tecnologies web dinàmiques.
- Dissenyar una API per servir la informació entre la base de dades i la interfície que es mostrarà a l'usuari.
- Posar en marxa i configurar un servidor per tal d'executar aplicacions web.
- Recollir dades i emmagatzemar-les de forma adient en una base de dades.
- Saber utilitzar biblioteques i APIs externes per integrar-les a l'aplicació.
- Crear una eina col·laborativa d'informació i que aquesta pugui ser visionada per la resta d'usuaris.
- Posar a disposició de la societat d'una eina per satisfer la necessitat d'informació.
- Desenvolupar un projecte complet de principi a fi, amb totes les seves fases.
- Demostrar la plena comprensió i assumpció dels coneixements i competències adquirits durant la realització del grau universitari.

## 3.2 Secundaris

- Col·laborar amb diferents organitzacions i institucions per detectar les necessitats i demandes a cobrir.
- Aprendre a gestionar i destriar informacions i recerca feta en línia o llibres físics.
- Utilitzar sensors i dades generades pel dispositiu i aplicar-les a les funcionalitats de l'aplicació web.
- Definir els límits del projecte i futures ampliacions, millores i modificacions.

## 4. Escenari

L'aplicació desenvolupada té com a objectiu poder acompanyar al viatger que utilitzi o vulgui utilitzar el transport públic abans, durant i després del seu trajecte. Aquest és l'escenari on s'ubica aquest projecte. L'usuari ha de poder preparar el seu viatge abans de sortir de casa, de manera que pugui conèixer totes les possibilitats que té al seu abast i quina és la que s'adequa més a les seves necessitats.

Quan emprèn el trajecte, ha de poder estar informat de qualsevol incidència que es produeixi. Aquest és una de les manques que hi ha en l'actualitat. Moltes vegades, falta la informació a l'usuari quan una de les expedicions va tard o té algun problema. Per intentar pal·liar aquest problema, a part de que l'administrador de l'aplicació pugui entrar els avisos, el propi usuari també ho pot fer (de forma més limitada) per avisar a la resta dels usuaris d'algun possible problema. A més, es crea un registre que serveix per analitzar i fer una estadística de possibles reiteracions d'incidències.

Finalment, l'aplicació acompanya al viatger un cop abandona l'autobús, l'anomenada 'última milla' fins a la seva destinació, amb el recorregut de l'itinerari a peu i les indicacions fins al seu destí. Analitzant els tres casos, l'aplicació ha de ser compatible multidispositiu ja que segurament, la planificació del trajecte es farà des d'un ordinador però les consultes durant el trajecte s'hauran de realitzar des d'un aparell mòbil (*smartphone*).

### 4.1 Projectes actuals

Abans de l'inici d'aquest projecte, el contingut que proveeix aquesta aplicació es podia trobar a les xarxes d'una manera poc organitzada i de difícil obtenció per a l'usuari. Es troba alguna aplicació que permet cercar els horaris, per una banda, i pàgines on es recull informació important per desplaçar-se (com per exemple, els horaris complets, com funciona el sistema de transport a la demanda, etc...), per l'altra.

Aprofundint, la Generalitat de Catalunya ha desenvolupat una aplicació web anomenada Mou-te<sup>7</sup>. Accedint a aquest lloc web, trobem un cercador que ens permet generar un itinerari a partir de dos punts concrets. Permet realitzar recorreguts amb transport públic entre dos punts qualsevols de Catalunya i combina la informació de tots els operadors (bus, tren, metro...). Com a punts negatius, a vegades ofereix recorreguts estranys i la consulta d'horaris específics és una mica enrevessada de cara a l'usuari.

<sup>7</sup> Enllaç a <https://mou-te.gencat.cat>

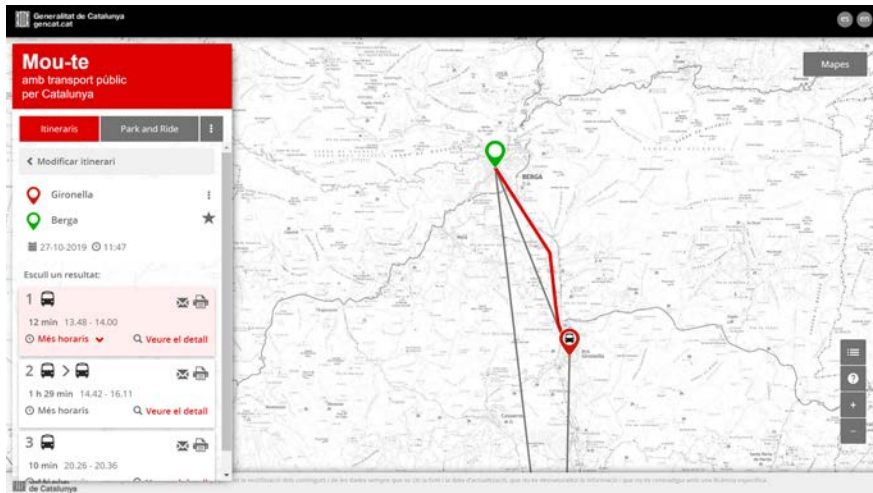


Figura 1 Exemple d'un recorregut a través del Mou-te

Una altra font d'informació per realitzar aquestes consultes són les pròpies pàgines dels operadors del servei. En el cas del Berguedà, la companyia que opera línies amb més passatge és ALSA. La pàgina d'ALSA permet realitzar les cerques d'horaris per dia i hora però només entre parades. A més, cal indicar de forma correcta i concreta el nom de la parada perquè, si no, no es localitza de forma correcta.

També hi ha una iniciativa privada de centralitzar aquesta informació de transport públic, que engloba el portal web Aquí Berguedà. En aquesta web s'hi troben taules amb horaris i enllaços als PDF de les diferents línies de la comarca. És una forma poc interactiva de buscar un horari entre dos punts però hi ha usuaris que prefereixen veure-ho així. També hi ha en marxa un senzill buscador d'horaris entre dues parades, però només d'una línia i que no inclou totes les parades.

**aquí BERGUEDA Horaris autobús Berga - Barcelona**

Proper bus  
Barcelona → Berga  
**20:10**  
Sortida 1:18:10

Berga

↕

Barcelona - Ronda Universitat

Dia de la setmana  
Dimecres

**Veure els horaris**

→ Consulta tots els horaris i línies dels autobusos del Berguedà  
🖨 Imprimir

	Berga	→	Barcelona - Ronda Universitat	Observacions
🚌	05:30	→	07:25	Entrada per Av. Diagonal. Para a Palau Reial, Pl. Francesc Macià, Avinguda Diagonal / Balmes i Balmes / Gran Via
🚌	06:30	→	08:35	Entrada per Vallvidrera. Para a Via Augusta / Tres Torres (FGC) i a Plaça Francesc Macià (Diagonal/Villarreal)

Figura 2 Exemple de cerca a l'aplicació de l'Aquí Berguedà

Finalment, a la comarca opera l'anomenat 'Transport a la demanda' que impulsa el Consell Comarcal del Berguedà. Consisteix en que l'usuari que vulgui utilitzar l'autobús, ha de fer una trucada el dia anterior sol·licitant el servei. Tota la informació està centralitzada al lloc web de la institució comarcal però només consisteix en una sèrie d'enllaços a PDF explicatius.

The screenshot shows the website's header with navigation links: INICI, INSTITUCIÓ, SERVEIS, NOTÍCIES, TRÀMITS, CONVOCATÒRIES, SEU ELECTRÒNICA, and RECOLLIDA PORTA A PORTA. The main heading reads 'El Consell Comarcal assumeix directament la gestió per telèfon del transport a demanda'. Below this, there is a yellow box with contact information: 'Els usuaris poden sol·licitar el servei que es realitza fins a les 17h del dia anterior laborable o la data del viatge contactant amb el Consell Comarcal del Berguedà' and 'COM Selectiva / WhatsApp: 655300355'. To the right, a text block explains that the phone number for requesting the service is 643 39 82 85, which has been changed from 11 since the beginning of the year due to organizational changes and the desire to improve the service. It notes that requests should be made via phone or WhatsApp at least one business day before the travel date. A list of municipalities served is provided: Ripoll, Manresa, Barcelona, Borredà, Cardener, Castellet, Castellar de l'Arany, Castellar de l'Hug, and Castellar del Riu. On the far right, there are social media sections for Twitter and Instagram, both showing a 'Carregar més...' button.

Figura 3 Pàgina d'informació del transport a la demanda, on la informació està en enllaços

L'aplicació desenvolupada ha intentat englobar els punts favorables de cadascun d'aquests llocs web intentant superar-ne les mancances que podem detectar a cada un d'ells.

## 5. Continguts

El mapa interactiu del transport públic a la comarca del Berguedà és una aplicació web que combina diferents apartats sobre aspectes relacionats amb els desplaçaments per la comarca. S'intenta assolir que el canvi entre les diferents parts de la pàgina es faci de forma transparent i gairebé imperceptible per a l'usuari, a diferència de llocs webs que han de carregar de nou cada pàgina cada cop que es prem un enllaç. Totes aquestes pàgines estan desenvolupades seguint els estàndards web de HTML, CSS, JavaScript i els frameworks PHP, React per al *frontend* i Node.JS per al *backend*.

Dit això, es poden considerar les següents parts de la pàgina.

### 5.1 Portada

La porta d'entrada a l'aplicació és la portada. Des d'aquí l'usuari pot accedir a les altres pàgines que formen part del lloc web. Al capdamunt de la pàgina s'hi situa l'encapçalament, amb el logotip de l'aplicació i l'accés al menú, ja sigui mostrant-ne les opcions (a les pantalles amples) o amb un botó que fa aparèixer tots els elements del menú (en dispositius mòbils).

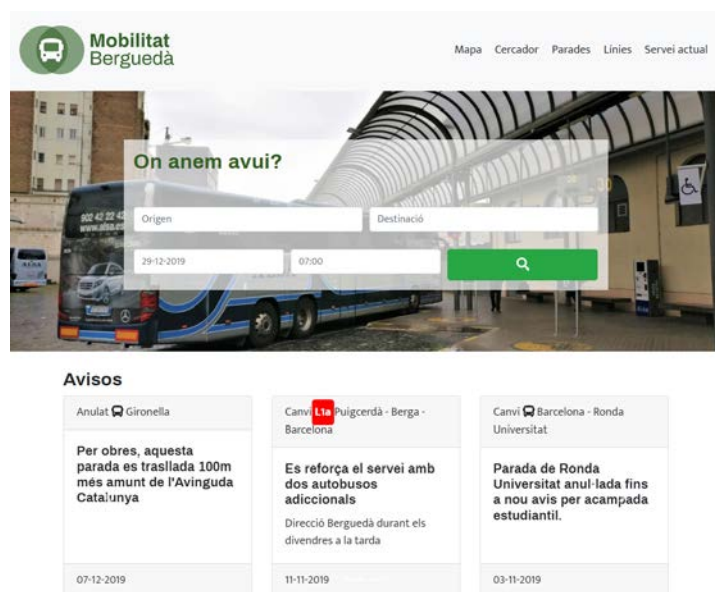


Figura 4 Captura de la portada de l'aplicació

La part central l'ocupa el formulari del motor de cerca, que és la part principal de l'aplicació. Des d'aquí, l'usuari pot especificar la consulta de l'itinerari que vol realitzar. Del motor de cerca se'n parla a l'apartat següent. Des d'aquí també es troba un enllaç que obre el mapa per buscar una parada concreta o visualitzar-ne les línies (punt 5.3).

Just a sota, en forma de carrusel, apareixen els diferents avisos actius, ja siguin de parades, línies o expedicions concretes. Aquests avisos estan ordenats de més a menys propers a la

data actual. Un cop, la data de finalització de l'avís ha passat, desapareixen d'aquest apartat de la portada.

Per tancar la portada, hi ha dos mòduls que mostren les parades més consultades, és a dir, les més importants, que enllacen a la pàgina de descripció de la parada (punt 5.4). Al seu costat, un llistat amb les línies ordenades pel seu número, de més petit a més gran. Com passava amb les parades, cada línia porta a la seva pàgina de descripció individual (punt 5.5).

A sota de totes les pàgines, s'hi inclou el peu de pàgina amb la informació de qui ha fet la pàgina, els col·laboradors i el contacte.

## 5.2 Motor de cerca

Quan l'usuari activa aquesta opció, es troba amb un formulari amb diverses caselles. Ha d'indicar una adreça d'inici i una altra de final. Mentre l'usuari escriu, el sistema li ofereix diverses opcions auto completades, corresponents a les parades que pot escollir.

Seguidament, ha d'escollir la data i hora en què vol realitzar el trajecte, ja que, els horaris no són els mateixos els dies laborables que els festius.

Un cop realitzada la cerca, l'aplicació mostra fins a quatre opcions que corresponen a la realització del trajecte. Aquestes propostes es fan en base als itineraris i horaris més òptims introduïts al sistema, però tenint la consideració de dos casos:

- Si l'usuari especifica l'itinerari entre dues parades inscrites al sistema, fa el càlcul del recorregut més curt (contant correspondències si n'hi ha) i els horaris més propers a la data i hora especificats.
- Si, en canvi, indica dos punts concrets ja sigui una adreça o unes coordenades, el programa ha de dividir el trajecte a recórrer en tres parts: l'itinerari a peu fins a la parada més propera, el recorregut per la línia (com el punt anterior) i l'itinerari a peu fins a la destinació. Si algun dels itineraris a peu supera els 4 km fins a la parada més propera, es dona com a resposta que l'itinerari no és possible.

Quan l'usuari selecciona un dels resultats, se li mostraran els detalls d'horari, distància, passos de l'itinerari a peu (si l'ha de fer), i informació del servei d'autobús que ha d'emprendre. A més, té l'opció d'imprimir l'itinerari, si li fa falta.

## 5.3 Mapa

Quan l'usuari accedeix a aquesta part de l'aplicació troba un mapa que ocupa la totalitat de la pantalla amb les línies i parades de l'autobús dibuixades. Per mostrar aquest mapa es fa servir una API d'un servei extern de proveïment de cartografia, Google Maps. A més, s'utilitza el servei de geolocalització del navegador que permet mostrar la ubicació actual de



l'usuari al mapa. Les mapes i línies estaran dibuixades seguint les característiques següents:

- Les parades estan ubicades segons la seva coordenada GPS.
- Les línies estan traçades entre les diferents parades del seu recorregut.
- Segons el nivell de zoom del mapa, el traç que representa les línies és més o menys gruixut i la icona de les parades més o menys gran. És a dir, amb la visualització d'una superfície més gran, els elements gràfics que apareguin al mapa seran més petits o se'n reduirà la quantitat per no sobresaturar la pantalla d'icones o d'informació.

L'usuari té l'opció de mostrar o ocultar el menú de capes, sense que això afecti al que pot visualitzar al mapa, utilitzant els botons corresponents.

### **5.3.1 Capes**

En aquest apartat, l'usuari té l'opció d'activar o desactivar el què vol veure al mapa. Des d'una opció de la interfície pot:

- Mostrar o amagar les línies que desitgi amb les seves parades corresponents.
- Mostrar o amagar les parades, des d'un llistat ordenat segons si es troben situades a la comarca o no. Degut al gran volum de parades, també es disposa d'una caixa de text que filtra les parades a partir del que ha escrit l'usuari.
- També pot mostrar o ocultar els autobusos que estiguin circulant en aquell moment en base al seu horari teòric. Això és una funció que es podria dotar de funcionalitats en temps real, si s'equipessin els vehicles amb un sensor GPS que també fos transmissor remot de la seva ubicació.

### **5.3.2 Parades, línies i serveis actuals**

Quan l'usuari fa clic en una parada, una línia o un autobús que està circulant actualment, apareix una finestra amb un accés a la informació que es genera a través dels apartats de parades, línies i expedicions que es detallen en el següent apartat. La finestra apareix sobreimpressionada, i l'usuari la pot tancar a través d'un botó per tornar a la vista inicial del mapa. També pot realitzar la mateixa acció fent clic o prement fora de la mateixa finestra.

## **5.4 Parades**

Cada parada té una pàgina pròpia on l'usuari pot consultar la informació següent de cadascuna d'elles:

- Direcció completa de la parada i la seva situació en un mapa que l'usuari pot ampliar o desplaçar com vulgui
- Línies que hi estacionen, amb un enllaç cap a la pàgina d'informació de la línia seleccionada.
- Horaris amb les properes sortides segons l'hora i dia en que s'està consultant l'aplicació. S'indica la línia i la seva destinació. En fer-hi clic, s'obre la pàgina de detall de l'expedició seleccionada.
- Horaris dels autobusos segons la seva destinació. En una taula es mostren els autobusos concrets que circulen per aquella parada respecte la destinació escollida. La taula mostra els serveis o expedicions dels dies feiners, dissabtes i festius. Igualment es mostren els horaris en direcció contrària, així l'usuari ho pot tenir com a referència per al seu trajecte, si és el mateix de forma habitual.
- Alertes temporals de la parada. En forma de fitxa, es mostren els avisos temporals o permanents que afecten a aquella parada.
- Si no es selecciona cap parada concreta, apareix un llistat amb enllaços a totes les parades, separades entre les que són situades a la comarca del Berguedà i les que no.

## 5.5 Línies

Similar a la de l'apartat anterior, cada línia disposa d'una pàgina pròpia que inclou la següent informació:

- Relació completa de totes les parades del seu recorregut en forma de termòmetre i la impressió sobre un mapa de l'itinerari seguit. Aquest mapa es pot mostrar o amagar, segons les preferències de l'usuari. Si la pantalla és petita, el mapa apareix ocult per defecte.
- Horaris complerts per dies. L'usuari pot seleccionar la direcció i el dia de la setmana i es genera una taula amb tot l'horari. Es disposa de l'accés a la pàgina per a cada expedició.
- Alertes temporals de la línia. En forma de fitxa, es mostren els avisos temporals o permanents que afecten a aquella línia.
- Si no es selecciona cap línia concreta, apareix un llistat amb enllaços a totes les línies introduïdes al sistema.

## 5.6 Expedicions

Com els dos apartats anteriors, cada expedició disposa d'una pàgina pròpia on s'hi pot consultar les dades següents:

- Relació completa de totes les parades del seu recorregut en forma de termòmetre, l'horari en el que teòricament efectua la parada i un mapa amb el recorregut. Si aquella expedició està en circulació, es mostra la icona sobre el termòmetre de la línia i sobre el mapa per fer-ne el seguiment en temps real.
- Informació sobre els dies de circulació i el període de l'any en que ho fa.
- Alertes temporals de l'expedició concreta. En forma de fitxa, es mostren els avisos temporals o permanents que afecten a aquella expedició.
- Avisos introduïts pels usuaris a través del sistema d'avisos. Es mostren les incidències notificades pels usuaris i en l'hora que ho han fet. A més, disposa del botó que obre la finestra per introduir un avís, com s'explica en el següent apartat.
- Si no es selecciona cap expedició concreta, es mostra una llista amb les expedicions que estan circulant en el moment actual amb l'origen i destinació de cadascuna i indicant si hi ha alguna incidència que afecta a l'expedició.

## 5.7 Sistema d'avisos

A part del sistema d'alertes que poden introduir els administradors de l'aplicació, es dota d'una funcionalitat que permet a un usuari alertar a la resta d'alguna incidència que es pugui haver produït en un servei concret. Aquesta funcionalitat es troba dins de la pàgina de cada expedició, ja que els avisos van relacionats amb un servei concret. Les dades que s'indiquen per registrar la incidència són:

- Autobús que pateix la incidència.
- Estació on es troba.
- Motiu de la incidència (retard, cancel·lació, afluència de viatgers...).

Un cop introduït, es mostra a la resta d'usuaris. La incidència registrada es mostra durant el dia natural en que s'ha produït, al dia següent és deixa de mostrar. Tot i així, queda emmagatzemat un històric de les incidències per a l'administrador per a una posterior recopilació de les dades i anàlisi de les mateixes.

## 5.8 Tauler d'administració

Hi ha una part només accessible a usuaris concrets. Des d'allà es poden modificar les línies, introduir nous serveis i parades o afegir els avisos d'alerta a línies, parades i serveis que poden ser permanents o temporals. Aquest apartat, per tant, inclou:

- Pàgina de login, per comprovar que l'usuari està registrat al sistema. En cas positiu, el *backend* retorna un token de seguretat que s'utilitza per poder realitzar les accions d'afegir, editar o eliminar elements de la base de dades.
- Edició dels diferents elements de la pàgina, amb possibilitat d'afegir-ne de nous o eliminar-los. Apartats per a les línies, parades, expedicions i avisos.
- Històric dels avisos enviats pels usuaris, amb una llista dels avisos introduïts a través del sistema d'avisos per poder-ne extreure dades de qualitat del servei.

## 6. Metodologia

Aquest projecte ha emprat diversos mètodes de recerca i treball per tal de satisfer els objectius plantejats en els apartats anteriors. Aquests mètodes s'han centrat bàsicament en la tasca de programació de l'aplicació web, el disseny de mateixa i la recerca de la informació que contindrà. Per tant, els diferents recursos es poden descriure de la manera següent:

### 6.1 Desenvolupament del codi

- S'ha analitzat els diferents llenguatges web que s'utilitzen actualment, donant una especial atenció a les noves tecnologies aparegudes recentment i escollir la quina millor es pugui adequar a la finalitat del projecte.
- S'ha observat diverses pàgines webs que utilitzen nous formats i tecnologies de programació web per veure'n les millores i qualitats respecte llenguatges més antics.
- S'ha seguit diversa formació de forma online per aprendre les claus bàsiques d'aquestes noves formes de llenguatge web que es complementarà amb el temari vist durant el grau en multimèdia de la Universitat Oberta de Catalunya.
- S'ha fet recerca sobre les bones pràctiques de programació web tenint especial cura en l'àmbit de la seguretat i la usabilitat. Això també s'ha tingut en compte, a l'hora de treballar en servidors que siguin accessibles a través d'Internet.
- S'han fet anotacions al codi font de l'aplicació, de forma que sigui més adequat de comprovar i verificar l'ús i desenvolupament de les diferents funcions i característiques.
- S'han realitzat diferents tests per comprovar les funcionalitats de l'aplicació. Entre ells, un test entre usuaris on s'han efectuat diverses proves per detectar possibles *bugs* o errors que estaven presents a l'aplicació.
- S'ha atès a les instruccions i consells que puguin aportar diversos professionals en el camp de l'enginyeria web com l'equip de professors i consultors de la Universitat Oberta de Catalunya.

### 6.2 Disseny de la interfície

- S'ha realitzat un estudi de webs de temàtica similar a la que es vol preparar. Això ha permès agafar bones pràctiques a l'hora de dissenyar una interfície i que, a la vegada, sigui familiar per a l'usuari.

- S'ha analitzat l'ús de diferents línies d'estil que s'utilitzen per desenvolupar aplicacions. D'aquesta manera, s'ha intentat mantenir la uniformitat respecte a les pautes que es marquen respecte les altres aplicacions.
- S'han realitzat diversos esbossos de la interfície per avaluar-ne la seva idoneïtat i presentar-los com a test per escollir-ne la millor proposta.
- S'ha redactat un llibre d'estil propi de la pàgina amb tots els detalls per facilitar el desenvolupament del disseny de l'aplicació i les seves futures ampliacions.

### **6.3 Recopilació de la informació**

- S'han fet cerques sobre la informació que hi ha disponible actualment a la xarxa sobre el transport públic a la comarca del Berguedà, com està distribuïda, la forma que és accessible i la que hi manca.
- S'ha treballat de forma conjunta amb institucions i operadors per obtenir informació fiable i actualitzada i que ho pugui seguir sent en un futur, un cop finalitzat el projecte.
- S'ha estudiat la forma més adient d'emmagatzemar la informació rebuda i amb la possibilitat que sigui oberta i accessible per a altres projectes d'aquest àmbit.
- S'han fet proves amb una petita quantitat d'informació per provar el correcte funcionament de l'aplicació abans de la introducció definitiva de totes les dades.

Finalment, i englobant els tres apartats s'ha elaborat aquesta memòria de forma que quedin reflectits tots els passos en l'elaboració del projecte i quedi com a text consultable per a possibles futurs projectes i derivats així com a referència per a possibles modificacions, ampliacions o esmenes a l'aplicació que es projecta.

# 7. Arquitectura de l'aplicació

## 7.1 Servidor

L'aplicació s'emmagatzema i s'executa en un mateix equip servidor que incorpora el funcionament de servidor per al *backend*, *frontend* i la base de dades. Els servidors poden ser configurats de manera per adquirir diferents funcionalitats (correu, impressió, web...). En el cas d'aquest projecte, s'utilitza un sol servidor per realitzar les tasques de base de dades, web i aplicació. El servidor s'encarregarà de gestionar i respondre les peticions del client realitzades a través del navegador web.

Durant el desenvolupament del projecte, s'ha utilitzat un servidor contractat externament, és a dir, a una empresa que ofereix els seus recursos, els hi cedeix un servidor amb una adreça IP concreta i permet als seus clients configurar-lo de forma que ells requereixin. En aquest cas, el servidor l'ha albergat l'empresa alemanya Hetzner i aquestes són les seves característiques:

<i>Sistema Operatiu</i>	Linux Debian 9
<i>Processador</i>	Intel Xeon Processor (Skylake, IBRS), 2 cores
<i>Memòria RAM</i>	8 GB
<i>Disc Dur</i>	75 GB

Taula 1 Característiques del servidor

En aquest servidor s'hi ha realitzat una instal·lació LAMP, per tal d'inicialitzar el servidor amb dues funcionalitats, la web i la de base de dades. LAMP és el acrònim de Linux-Apache-MySQL-PHP. Els arxius de l'aplicació web s'emmagatzemen a la memòria del disc dur del servidor i és el servidor Apache qui s'encarrega de redirigir al contingut corresponent totes les peticions HTTP que rep el servidor. S'ha dividit les dues peticions en dos subdominis principals, ja que l'aplicació s'executa a través d'un port concret del servidor, i Apache s'encarrega d'interpretar l'accés des del domini cap a aquest port<sup>8</sup>. L'API, té un subdomini diferent per tal de accedir només als fitxers PHP que generaran l'API.

S'ha pres aquesta decisió per tal de que, a nivell dels dominis DNS l'API i l'aplicació estiguin separats, i ja que, Apache, quan se li assigna a un domini que porti a una aplicació que s'executa a través d'un port de l'aplicació (cas que es donava durant la fase de

<sup>8</sup> Durant el desenvolupament del projecte s'han fet servir els dominis [api.mobilitat.ricardcodina.cat](http://api.mobilitat.ricardcodina.cat) per al *backend* i [mobilitat.ricardcodina.cat](http://mobilitat.ricardcodina.cat) per al *frontend*.

desenvolupament de l'aplicació), reserva tots els seus camins per l'aplicació i no es poden variar per a utilitzar-los per un altre servidor, per tant, no podríem accedir a domini/api.

MySQL és la base de dades que s'executa al servidor, tal i com es detalla al punt següent. També es fa la instal·lació del llenguatge i servidor d'execució PHP per interpretar les ordres que rebrà a través dels frameworks (punt 8.1.2) i també del generador de l'API. Finalment, s'ha instal·lat el procés de Node.js, a través d'NPM, que serveix per l'execució de l'aplicació web.

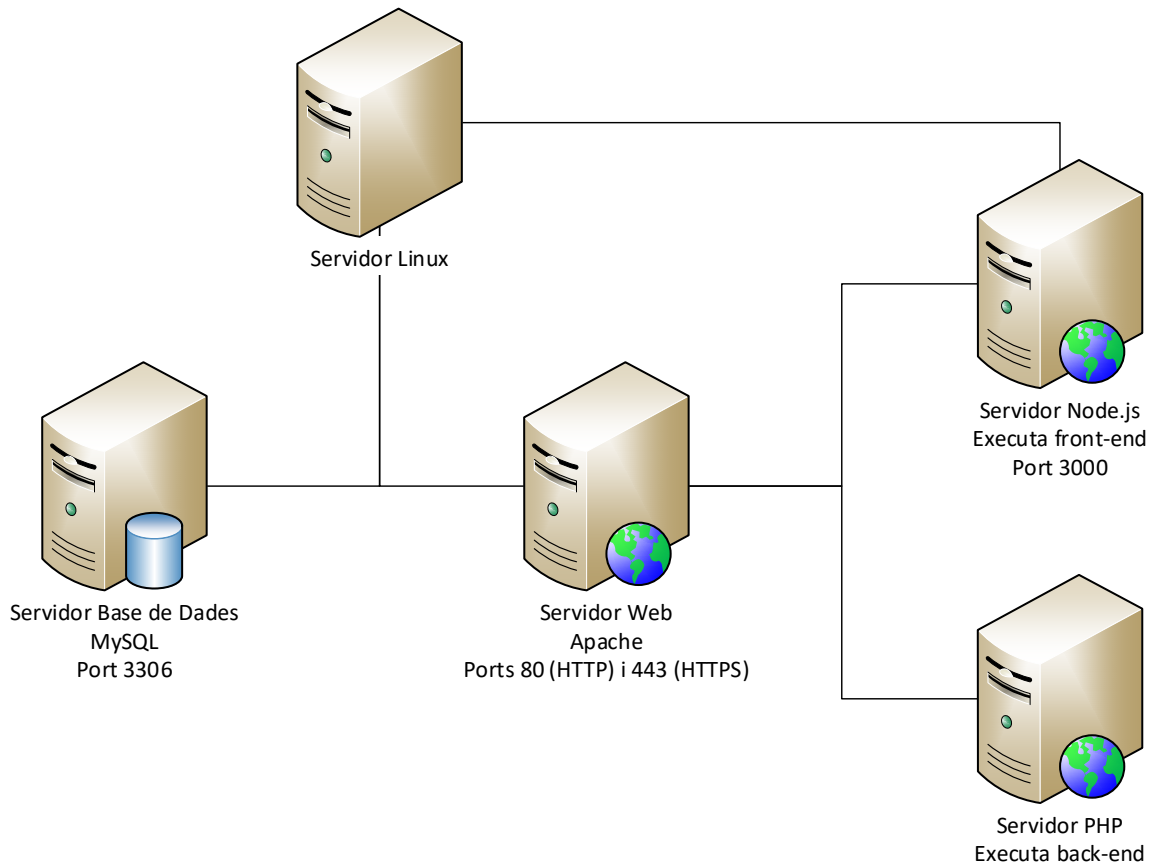


Figura 5 Representació esquemàtica de l'arquitectura del servidor en el moment de publicació de la memòria

## 7.2 Base de dades

La base de dades es on resideix tota la informació que conté l'aplicació que farà crides a la base de dades a través del *backend* per obtenir el contingut que necessita per poder-lo mostrar a l'usuari. En l'actualitat, en aplicacions i pàgines web s'utilitzen dos tipus de bases de dades principalment, les *mySQL* i les *noSQL*.

### 7.2.1 Sistema i llenguatge

Per una banda, el sistema **mySQL** utilitza el llenguatge SQL, acrònim de Structured Query Language, que permet generar les consultes i ordres a la base de dades i es distribueix de forma gratuïta i sota llicències de programari lliure. Organitza les seves dades en taules, estructurades a la vegada en columnes i files. Com avantatges, permet la integració senzilla amb altres llenguatges de programació per obtenir-ne les dades i hi ha a l'abast diferents



eines per poder gestionar les dades que formen part de la base de dades de forma més senzilla com, per exemple, PHPMyAdmin que permet modificar el contingut de la base de dades a través d'una aplicació web.

Per altra banda, en aplicacions de *big data* o que gestionin molt contingut en temps real, han guanyat pes les bases de dades dites **NoSQL**. En aquest tipus, les dades s'emmagatzemen de forma no relacional, i utilitzant diferents tipus de categoritzadors per ordenar les dades. Per exemple, d'aquest tipus, trobaríem MongoDB, que organitza les dades en col·leccions i documents, i enregistra les dades en un format similar al de JSON, és a dir, amb valors posats entre claudàtors i separats per comes.

Entre els dos tipus de bases de dades trobem avantatges i inconvenients. Si en *mySQL* hem de definir prèviament l'estructura de cada taula (dades de cada columna, índex, claus,...), en el format *NoSQL* no hi ha restriccions de disseny a l'hora d'introduir nous elements en una col·lecció, de fet, no és necessari que cada element contingui els mateixos ítems que els altres elements en una mateixa col·lecció<sup>9</sup>. Per tant, *mySQL* és molt més rígid que bases de dades *NoSQL*, i això fa que sigui més propens a errors si no s'especifica el tipus de dada que sol·licita, no s'omplen tots els camps...

També hi ha diferència en la manera que fem les peticions ja que, com s'ha explicat, *mySQL* utilitza el llenguatge *SQL* que permet executar múltiples tipus de consultes. Amb *NoSQL* s'utilitzen expressions que, tot i que solen ser més senzilles a l'hora d'inserir o eliminar elements, es fan més complexes si hem d'unir diverses col·leccions. En aquest cas, si s'ha de tenir relacions entre taules, el llenguatge *SQL* ens ofereix més possibilitats.

Per tant, i analitzats els requeriments per a aquest projecte, s'utilitza una base de dades *MySQL*, ja que les dades són rígides, i hi ha relació entre les diferents taules. Les dades que conté cada taula estan especificades des de bon començament i és important les diferents dependències entre els elements de les taules. A més, moltes de les peticions es podran generar utilitzant el propi llenguatge *SQL*. Aquestes peticions son les que després el *backend* interpreta i s'enviaran al *frontend* a través de l'API *REST*.

### **7.2.2 Estructura**

Pel projecte del mapa del mobilitat s'han definit tot una sèrie d'entitats i atributs en base als requisits d'informació que contindrà l'aplicació. A l'apartat 9, es pot consultar el diagrama *UML* de l'estructura i la seva definició.

<sup>9</sup> (Simform, 2019)

## 7.3 Backend

El *Backend* de l'aplicació s'encarrega de generar les peticions que necessita l'aplicació i retorna la informació sol·licitada cap al *frontend*. Aquesta funció la fa a través de crides a una REST API generada amb llenguatge PHP i consultes a la base MySQL que retorna una resposta en format JSON per tal que pugui ser interpretada pel generador de la interfície del *frontend*.

API significa *Application Programming Interface* i s'encarrega de connectar l'aplicació amb el servidor<sup>10</sup>. Per tal d'obtenir la informació que es mostrarà a la pàgina, l'aplicació (*frontend*) fa una petició a l'API (*backend*) a través de les diferents consultes que s'han establert. Aquesta API retornarà un objecte en format JSON contenint la seva resposta. Acte seguit, l'aplicació interpreta la informació rebuda i la mostra a l'usuari. Aquest seria el funcionament d'una API GET, la majoria que s'utilitzen en aquest projecte. Però també n'hi ha de tipus POST, DELETE, PUT, COPY...

En aquest projecte, l'API s'ha desenvolupat en base al llenguatge PHP i fent crides a la base de dades en llenguatge MySQL. Per una banda, PHP interpreta els valors que rep, genera la consulta a la base de dades i codifica les respostes en format JSON, per tal que sigui interpretable per al *backend*. Durant les peticions PHP també es generen les diferents funcions que permeten fer funcionar el motor de cerca i també peticions a altres APIS externes (veure punt 12) per tal de generar la resposta que s'ha de facilitar.

### 7.3.1 Llistat d'APIS

Es detallen les peticions API utilitzades per generar consultes del *backend* cap al *frontend*. Depenent del mètode utilitzat, cada petició realitza una acció o una altra.

<b>NOM</b>	Parada
<b>URL</b>	parada/
<b>MÈTODES</b>	GET: Retorna tot el llistat de parades del sistema. GET (?id=:id): Retorna la informació de la parada que correspon a la id. POST: Afegeix la parada amb la informació rebuda. PUT: Edita la parada d'una ID específica amb la informació rebuda DELETE: Elimina la parada de la ID especificada.
<b>HEADERS</b>	En el cas dels mètodes POST, PUT i DELETE necessiten l'enviament del token generat per procedir a l'execució de l'acció si aquest és correcte.

<b>NOM</b>	Línia
<b>URL</b>	linia/
<b>MÈTODES</b>	GET: Retorna tot el llistat de línies del sistema. GET (?codi=:codi): Retorna la informació de la línia que correspon al codi indicat. POST: Afegeix la línia amb la informació rebuda. PUT: Edita la línia d'un codi específic amb la informació rebuda DELETE: Elimina la línia del codi especificat.

<sup>10</sup> (freeCodeCamp, 2016)

**HEADERS** | En el cas dels mètodes POST, PUT i DELETE necessiten l'enviament del token generat per procedir a l'execució de l'acció si aquest és correcte.

<b>NOM</b>	Expedicio
<b>URL</b>	expedicio/
<b>MÈTODES</b>	GET: Retorna tot el llistat d'expedicions del sistema. Si ?circulaAra mostra només les expedicions en circulació. GET (?id=:id): Retorna la informació de l'expedició que correspon a la id. POST: Afegeix l'expedició amb la informació rebuda. PUT: Edita l'expedició d'una ID específica amb la informació rebuda (queda pendent de programar en la primera versió de l'app) DELETE: Elimina l'expedició de la ID especificada.
<b>HEADERS</b>	En el cas dels mètodes POST, PUT i DELETE necessiten l'enviament del token generat per procedir a l'execució de l'acció si aquest és correcte.

<b>NOM</b>	Avis
<b>URL</b>	avis/
<b>MÈTODES</b>	GET: Retorna tot el llistat d'avisos del sistema. GET (?id=:id): Retorna la informació de l'avis que correspon a la id. POST: Afegeix l'avis amb la informació rebuda. PUT: Edita l'avis d'una ID específica amb la informació rebuda DELETE: Elimina l'avis de la ID especificada.
<b>HEADERS</b>	En el cas dels mètodes POST, PUT i DELETE necessiten l'enviament del token generat per procedir a l'execució de l'acció si aquest és correcte.

<b>NOM</b>	AvisUsuari
<b>URL</b>	avisUsuari/
<b>MÈTODES</b>	GET: Retorna tot el llistat d'avisos introduïts pels usuaris del sistema. GET (?idExpedicio=:idExpedicio): Retorna la informació dels avisos segons l'ID d'una expedició POST: Afegeix un avis d'usuari amb la informació rebuda.
<b>HEADERS</b>	El mètode POST no necessita token ja que l'acció és pot realitzar per qualsevol usuari.

<b>NOM</b>	Estadistica
<b>URL</b>	estadistica/
<b>MÈTODES</b>	GET: Retorna una llista amb els valors totals dels objectes emmagatzemats al sistema. S'utilitza per generar la pàgina inicial de l'administrador.

<b>NOM</b>	Generaltinerari
<b>URL</b>	Itinerari/generaltinerari.php?origen=:origen&destinacio=:destinacio&data=:data&hora=:hora
<b>MÈTODE</b>	GET
<b>HEADERS</b>	-
<b>REQUEST</b>	-
<b>RESPOSTA</b>	<ul style="list-style-type: none"> <li>• <b>dadesItinerari:</b> retorna les dades de l'itinerari <ul style="list-style-type: none"> <li>○ origen</li> <li>○ destinacio</li> <li>○ data</li> <li>○ hora</li> <li>○ paradaOrigen</li> <li>○ paradaDestinacio</li> </ul> </li> <li>• <b>itinerariAParadaOrigen:</b> retorna les dades de l'itinerari a peu generat per Google Maps.</li> <li>• <b>Itineraris:</b> Retorna els itineraris que coincideixen amb la petició <ul style="list-style-type: none"> <li>○ Opcions <ul style="list-style-type: none"> <li>▪ Linia</li> <li>▪ Expedicio</li> <li>▪ Parades</li> <li>▪ horesPas</li> <li>▪ recorregut</li> </ul> </li> </ul> </li> <li>• <b>itinerariAParadaDestinacio:</b> retorna les dades de l'itinerari a peu generat per Google Maps.</li> </ul>

## 7.4 Frontend

El *frontend* s'executa gràcies a Node.js i ReactJS, que són dos frameworks utilitzats per al desenvolupament d'aplicacions web. L'aplicació en estat de desenvolupament corre per defecte al port 3000 del servidor. Un cop es genera la versió de producció, es pot executar directament en un servidor php. La part de la interfície de l'aplicació està desenvolupada utilitzant els mètodes de render que conté ReactJS i que criden a objectes encarregats a generar una part de la pàgina.

Aquests *frameworks* s'utilitzen amb diferents mòduls o llibreries que permeten afegir característiques a l'aplicació. Per exemple, per carregar les diferents parts de l'aplicació s'utilitza el mòdul extern de *Router* que permet especificar un camí concret per a cada pàgina, o la llibreria *google-maps-react* que s'encarrega de generar el mapa des de Google Maps. El funcionament de React i els seus mòduls s'amplia a l'apartat 8.

Aquests frameworks actuen de forma conjunta amb una fulla d'estil CSS on s'hi defineixen les característiques d'estil dels diferents elements de la pàgina. La fulla d'estil conté també les ordres de disseny adaptatiu que adapta l'aplicació segons la mida de la pantalla des d'on es consulti.

Com que React, es basa en l'ús del llenguatge orientat a objectes, les diferents parts de l'aplicació s'estructuren en components. Cada component es programa de forma separada i després es genera la interfície en base als components que es renderitzen i a les variables i funcions que es transmeten entre ells. Per exemple, a la figura següent veiem a partir de quins components es renderitza la portada de l'aplicació.

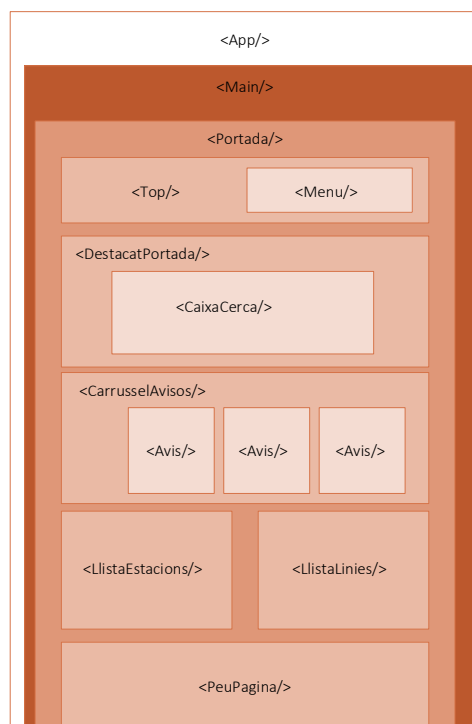


Figura 6 Components que formen part de la portada de l'aplicació

## 7.5 Client

Per tal d'accedir a l'aplicació el client només necessita alguns recursos mínims, ja que la major part de l'execució de l'aplicació es realitza al servidor. No obstant, l'equip del client hau de complir alguns requisits. L'accés a l'aplicació es fa a través d'un navegador web amb la introducció d'una direcció web que apunta cap al servidor.

### 7.5.1 Dominis

Per tal que el client pugui accedir a la nostra aplicació s'ha assignat un domini d'accés a l'aplicació. El que fa aquest domini és redirigir el nom cap a la direcció IP del servidor. Un cop arriba la sol·licitud al servidor, Apache és qui s'encarrega de servir el contingut de l'aplicació.

En aquest projecte, s'ha considerat la utilització de dos dominis diferents, un per al backend i un altre, per al frontend. S'ha pres aquesta decisió per tal de dividir més marcadament l'API de la interfície. D'aquesta manera, és té la possibilitat de tenir allotjada l'API en un servidor diferent al del frontend. Pels recursos que es tenen en aquest projecte, tant l'API com el frontend comparteixen el mateix servidor, però d'aquesta manera és té la possibilitat de separar-ho si fos necessari en un futur.

### 7.5.2 Navegador web

Com s'ha dit, s'accedeix a l'aplicació a través d'un navegador web. L'aplicació està desenvolupada amb la versió 6 d'ECMAScript (ex-JavaScript), apareguda l'any 2015 i conté noves funcionalitats, com per exemple, l'ús de classes o les funcions fletxa o lambda. Si el client utilitza un navegador web que no suporta aquesta versió d'ECMAScript i, per tant, no interpreta aquestes noves funcionalitats, l'aplicació no es mostrarà correctament i donarà errors.

Segons les especificacions d'ECMAScript 6 (W3Schools, 2015), es defineixen els navegadors que sí tenen compatibilitat amb aquesta versió i que, per aquest motiu, serien vàlids per executar el projecte:

<i>Navegador</i>	<i>Versió</i>	<i>Llançament</i>
Apple Safari	10.0	Juliol 2016
Google Chrome	58.0	Gener 2017
Microsoft Edge	14.0	Agost 2016
Mozilla Firefox	54.0	Març 2017
Opera	55.0	Agost 2018

Taula 2 Navegadors principals compatibles amb ECMAScript 6

Si el client utilitza les darreres versions d'algun d'aquests navegadors web predominants al mercat, podrà accedir i fer servir l'aplicació correctament. A més, aquests navegadors no

només es troben instal·lats als equips d'escriptori si no que també els utilitzen els dispositius mòbils, *smartphones* i tauletes. D'aquesta manera, el client pot utilitzar qualsevol aparell que tingui connexió a internet i un navegador web per consultar l'aplicació.

### **7.5.3 Connexió a internet**

Per tal de carregar l'aplicació, el client necessita una connexió a internet estable amb una velocitat superior a 1Mbps per tal de carregar de forma òptima tots els elements de l'aplicació. Els recursos que necessitaran més banda d'ampla seran els elements gràfics (imatges, símbols, logotips) i els mapes, que criden una api externa (veure punt 11).

## 8. Plataforma de desenvolupament

L'aplicació utilitza plataformes i tecnologies en plena expansió actualment, per tal de demostrar i experimentar amb els avantatges que poden oferir respecte als que s'han estat utilitzant majoritàriament fins ara. Ja que el resultat és essencialment una peça de software en forma d'aplicació web, aquest apartat se centra en aquest element.

### 8.1 Software

#### 8.1.1 Programació web

A l'hora d'endinsar-se a la programació web, s'ha de diferenciar que aquests es poden executar del cantó del client o del servidor. És a dir, qui s'encarregarà de generar el contingut que es veurà a través de la pantalla quan es consulta un lloc web.

Per una banda, en els **llenguatges del cantó client**, el codi s'executa directament en el navegador web de la persona que consulta la pàgina. El navegador rep el codi i és el mateix equip que utilitza els seus propis recursos per interpretar-los i executar les funcionalitats que pugui tenir. Algun dels llenguatges del cantó client més utilitzats podrien ser HTML, que defineix l'estructura del web, CSS, que controla el disseny i aparença de les pàgines o Javascript, que actua com a llenguatge de programació i permet interactuar amb els altres elements HTML o CSS<sup>11</sup>.

De l'altra, els **llenguatges del cantó servidor** fan que totes les operacions de càlcul i execució de codi es produeixin a l'equip servidor que retornarà al client un codi ja comprensible per ell, per tant, s'obté un codi en llenguatge del cantó client. D'aquesta manera es pot descarregar de recursos i peticions al client, ja que una bona part de l'execució del codi es passa a fer al servidor. Alguns dels llenguatges que formarien part d'aquesta categoria serien PHP, ASP.net o JSP<sup>12</sup>.

#### 8.1.2 Frameworks

Els llenguatges esmentats en l'apartat anterior permet la creació de pàgines i aplicacions web potents però, per a un programador, haver de construir tots els mètodes i estructures que requereix per fer funcionar la web li pot suposar una gran despesa de temps i recursos. A més, segurament, moltes de les funcionalitats que ha de desenvolupar ja han estat utilitzades en altres projectes. Gràcies al llenguatge orientat a objectes i al paradigma de l'encapsulació i l'ocultació han aparegut els anomenats *frameworks* web.

<sup>11</sup> (Microsoft Docs, 2019)

<sup>12</sup> (Hostinet, sense data)

Aquests *frameworks* web son grans llibreries amb recursos reutilitzables que fan més fàcil l'escriptura de llenguatge web, ja que aporten moltes funcions que permeten simplificar les tasques que s'empren usualment en el desenvolupament web. Igualment, es poden ampliar les seves funcionalitats fent crides a altres biblioteques que ens permetin afegir noves funcionalitats a les aplicacions. De la mateixa manera, l'ús d'aquestes llibreries o *frameworks* permet que el codi sigui més comprensible per a altres desenvolupadors que coneguin el tipus de *framework* que s'utilitza.

Aquest projecte utilitza principalment el *framework* ReactJS. React, com la seva pròpia pàgina web<sup>13</sup> indica, és una llibreria de Javascript per desenvolupar interfícies d'usuari. És una tecnologia desenvolupada per Facebook l'any 2013 i que segueix sent mantinguda i actualitzada per la mateixa empresa nord-americana. Actualment, moltes de la pàgines d'internet utilitzen aquesta tecnologia. A banda de les pàgines i aplicacions del mateix Facebook (Instagram, Whatsapp web...) que utilitzen React, Twitter, Airbnb o la BBC han construït el seu *frontend* a través de React (Coder Academy, 2016).

Utilitza tota la programació orientada a objectes provinent de Javascript però implementa el mètode `render()` per generar les interfícies web. A més, permet que aquestes interfícies s'actualitzin en temps real en base a la informació que té emmagatzemada a les variables d'estat. Quan un component actualitza aquest estat de l'aplicació, el propi React torna a cridar el mètode `render()` per actualitzar la interfície de forma transparent per a l'usuari. Per aquest motiu, React s'utilitza per generar interfícies d'usuari que emmagatzemen i mostren informació a l'usuari per aplicacions web.

A més, una altra de les característiques de React són els anomenats *props* o propietats. Els *props* son els atributs que es poden passar entre components. D'aquesta forma, es poden passar variables i funcions entre components per generar objectes en base als *props* passats. Aquest fet simplifica la reutilització del codi d'un component i la generació del seu contingut en base el que li passa el component superior.

Per tal que React funcioni, el servidor té instal·lat l'entorn d'execució Javascript `node.js` i la seu sistema de gestor de paquets `npm`. L'entorn `node.js` és qui ha d'executar l'aplicació al servidor i servir-la a través d'un dels seus ports. Com s'ha explicat a l'apartat anterior, després és Apache qui s'encarrega de servir l'aplicació a través del domini que consulta l'usuari. `Npm` funciona per instal·lar noves llibreries a l'aplicació i donar les ordres per executar-la, entre d'altres funcions. També s'utilitza per compilar la versió final de l'aplicació que després es pot executar només amb un servidor php sense necessitat d'obrir un port específic per a l'aplicació ni haver de tenir instal·lat `nodeJS` o `Npm`.

<sup>13</sup> (React, 2013)



A l'aplicació, també es fa ús de diverses llibreries externes que permeten ampliar les funcionalitats de React. En el cas de l'obtenció de la informació que prové del *backend*, Axios.

S'ampliarà aquesta informació al punt 12.4 però Axios s'utilitza per rebre la informació des del *backend* encarregant-se de fer les crides a l'API, rebre'n la resposta i interpretar-la. La informació rebuda se sol emmagatzemar als estats de cada component de React que necessita fer crides i després es mostra a l'usuari. Axios també és l'encarregat d'enviar la informació cap a l'API que després serà emmagatzemada a la base de dades. En rep una confirmació, de si la petició ha sigut correcta o no i, d'aquesta manera es pot mostrar a l'usuari un missatge.



### 8.1.3 Programari per a la programació del projecte

Per realitzar l'escriptura del codi s'ha utilitzat el programari gratuït de Microsoft, **Visual Studio Code**, que permet programar en diferents llenguatges, és un programa força lleuger i que carrega ràpidament i que s'ha anat popularitzant durant els últims anys. Segons el llenguatge escollit, pinta les diferents línies i elements del codi segons la seva funció i, a més, ressalta els errors en temps real. D'aquesta manera, és més senzill detectar alguns problemes derivats d'errors de picatge o de mal ús d'alguns mètodes o recursos.

No obstant, no tots els problemes es poden detectar a través de Visual Code Studio. S'ha hagut de recórrer a altres eines per realitzar les tasques de *debugging*. S'ha fet servir les **eines de desenvolupador de Google Chrome**, des d'on es pot consultar la consola (per veure els missatges que s'hi generen), les peticions dels recursos en xarxa (per exemple, per veure les peticions al backend, consultar-ne les dades enviades i veure'n les rebudes) i els diferents indicadors de i simuladors (per provar si la pàgina s'executarà correctament en dispositius mòbils).

Dins de les eines de desenvolupador de Google, s'ha afegit una extensió<sup>14</sup> que inclou tota una sèrie d'eines específiques per a React, des d'on es poden consultar tots els

<sup>14</sup> *React Developer Tools* per a Google Chrome: <https://chrome.google.com/webstore/detail/react-developer-tools/fmkadmapgofadopljbjfkapdkoienihi>

components que s'han carregat a l'aplicació, comprovar quines propietats reben de forma individual i quins son els seus estats. A més, aquests estats es poden modificar manualment per poder modificar el comportament de l'aplicació sense haver de fer els canvis al codi font.

# 9. Diagrames UML

En aquest apartat, es poden consultar els diagrames UML de la base de dades i del backend.

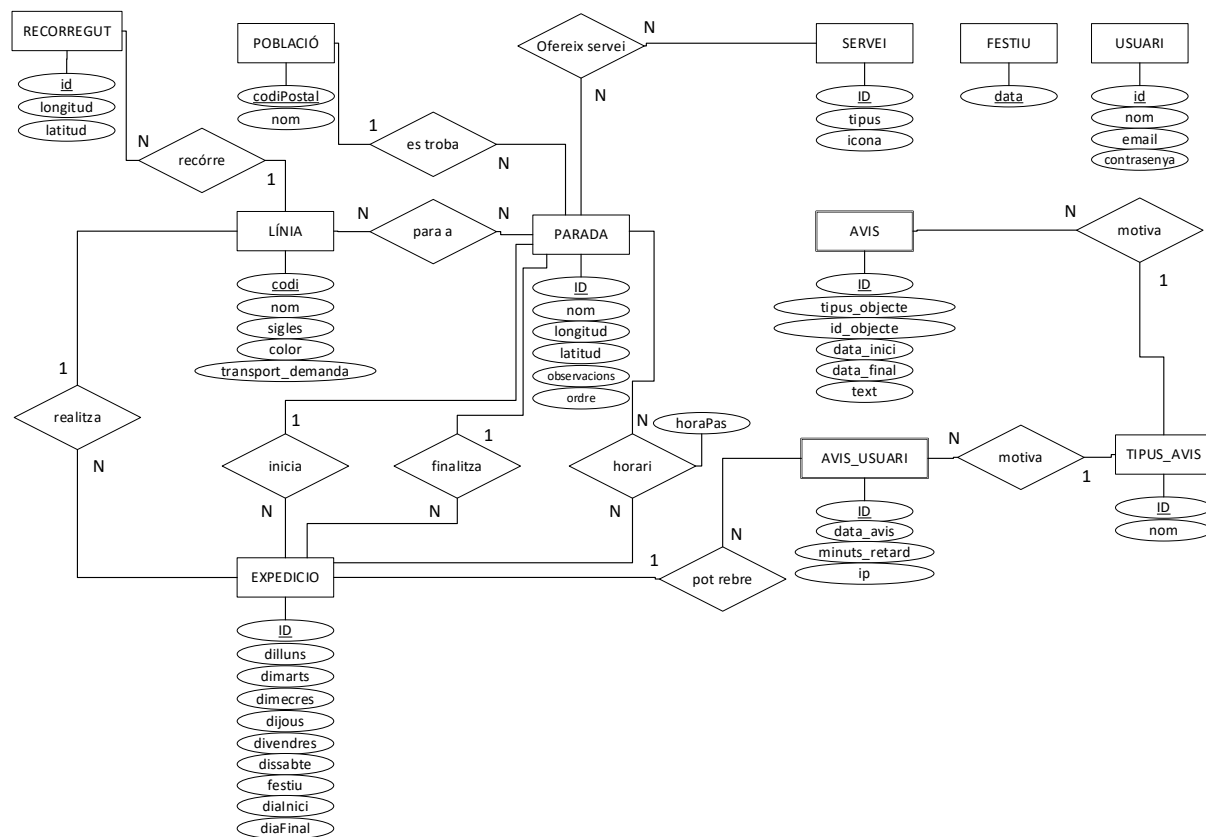


Figura 7 Diagrama UML de la base de dades

## ENTITATS

**LÍNIA** (codi, nom, sigles, color, transportDemanda): informació única de cada línia de bus. TransportDemanda indica si la línia introduïda està sota el règim de Transport a la Demanda, activable en una futura actualització de l'aplicació.

**PARADA** (id, nom, longitud, latitud, observacions, codiPostal, ordre) on {codiPostal} referencia POBLACIO(codiPostal): informació de les parades del territori, a observacions s'hi pot afegir qualsevol tipus de text, i ordre, és un ordre personalitzat de les parades per ordenar-les segons les preferències de l'administrador.

**POBLACIO** (nom, codiPostal, bergueda): registre dels municipis. Indiquem si forma part de la comarca del Berguedà o no.

**SERVEI** (id, tipus, icona): serveis que poden oferir les parades.

**RECORREGUT** (id, latitud, longitud, codiLínia) on {codiLínia} referencia LÍNIA(codi): indicació de les coordenades dels punts que uniran la línia al mapa.

**EXPEDICIO** (id, codiLínia, dilluns, dimarts, dimecres, dijous, divendres, dissabte, festiu, paradaInici, parada Final, diaInici, diaFinal) on {paradaInici, paradaFinal} referencia PARADA(id): conté tots els autobusos programats: els dies de la setmana indiquen si circula o no l'expedició el dia concret, la parada inici i final s'indiquen ja que poden no coincidir amb

les parades d'inici i final de la línia (servei parcial) i el dia d'inici i final es pot indicar en cas que l'expedició circuli només durant un període concret.

AVIS (id, tipusObjecte, idObjecte, tipusAvis, dataInici, dataFinal, títol, text) on {idObjecte} referencia o LINIA(codi) o PARADA(id) o EXPEDICIO(id) i on {tipusAvis} referencia

TIPUSAVIS(id): avisos introduïts per l'administrador. Es pot indicar la durada concreta de l'avis a través dels atributs de DataInici i DataFinal.

AVISUSUARI (id, tipusAvis, dataAvis, retard, ip) on {tipusAvis} referencia TIPUSAVIS(id): avisos introduïts per l'usuari. S'emmagatzema la data de l'avis en que s'ha generat, per mantenir-ne un històric.

TIPUSAVIS (id, nom): diferents categories d'alerta

FESTIU(data): indica els dies festius diferents del diumenge.

USUARIS (id, nom, email, contrasenya): manté una relació dels usuaris que tenen accés a l'administrador. La contrasenya s'emmagatzema de forma encriptada.

## INTERRELACIONS

EFFECTUA PARADA (id, codiLínia, idParada, ordre) on {codiLínia} referencia LINIA(codi) i on {idParada} referencia PARADA(id): assigna les parades d'una línia en l'ordre que segueixen.

HORARI (id, idExpedicio, idParada, horaPas) on {idExpedicio} referencia EXPEDICIO(id) i on {idParada} referencia PARADA(id): assigna les hores de pas per cada expedició i parada.

OFEREIX SERVEI (id, idParada, idServei) on {idParada} referencia PARADA(id) i on {idServei} referencia SERVEI(id): assigna els serveis a una parada.

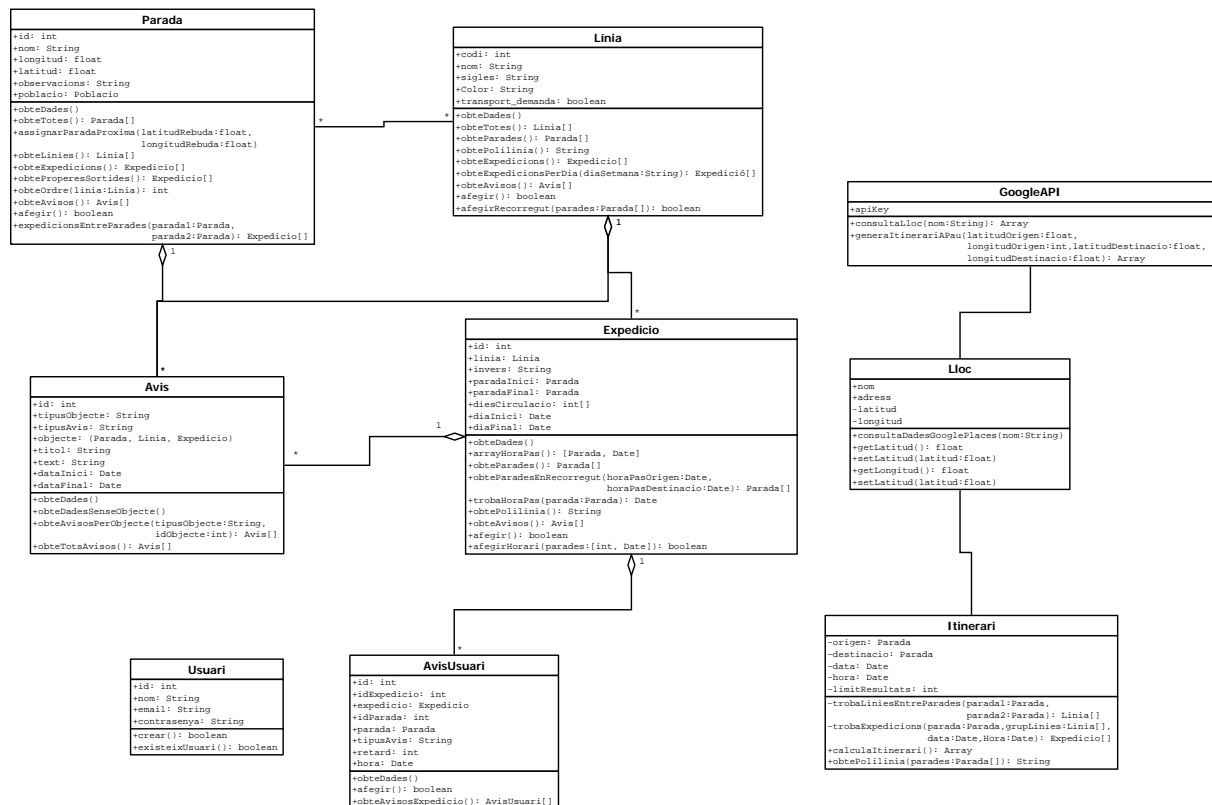


Figura 8 Diagrama UML del backend

# 10. Planificació

Aquest projecte es dividirà en diferents fases que engloben els diferents processos per planificar, dissenyar, desenvolupar i avaluar el resultat que es planteja. Les diferents tasques, fases i fites del projecte es distribueixen de la forma següent:

Tasca	Data inici	Data final	Durada
<b>Fase 1: Planificació</b>			
1. Elecció i plantejament del projecte	18/09/19	23/09/19	4 dies
2. Redacció de la part introductòria del projecte	24/09/19	01/10/19	6 dies
Fita 1: Entrega 1a PAC	01/10/19		
<b>Fase 2: Disseny</b>			
1. Preparació de la interfície			
1.1 Identificació dels elements de la interfície i pantalles de l'aplicació.	02/10/19	08/10/19	5 dies
1.2 Redacció i definició del llibre d'estil.	07/10/19	12/10/19	6 dies
1.3 Disseny i creació dels <i>wireframes</i> de baix nivell.	18/10/19	20/19/19	2 dies
1.4 Disseny i creació dels <i>wireframes</i> d'alt nivell.	21/10/19	23/10/19	3 dies
2. Anàlisi tècnic			
2.1 Elecció del llenguatge de programació.	02/10/19	06/10/19	4 dies
2.2 Aprenentatge i repàs de les funcions del llenguatge de programació.	07/10/19	20/10/19	11 dies
2.3 Definició de les fases i estructura de l'aplicació.	21/10/19	30/10/19	8 dies
3. Contingut			
3.1 Definició del contingut i informació a incorporar a la pàgina.	18/10/19	20/10/19	2 dies
3.2 Recol·lecció i ordenació de tota la informació que apareixerà a la pàgina.	21/10/19	27/10/19	6 dies
3.3 Disseny de l'estructura de la base de dades que emmagatzemarà tota la informació.	21/10/19	27/10/19	6 dies

Fita 2: Entrega 2a PAC	30/10/19		
<b>Fase 3: Producció</b>			
1. Configuració del servidor	31/10/19	03/11/19	3 dies
2. Redacció del codi font de l'aplicació			
2.1 Desenvolupament del codi del <i>back-end</i>	04/11/19	24/11/19	16 dies
2.2 Desenvolupament del codi del <i>front-end</i>	04/11/19	24/11/19	16 dies
2.3 Desenvolupament del codi de la interfície visual	18/11/19	01/12/19	11 dies
3. Base de dades			
3.1 Creació de l'estructura de la base de dades	04/11/19	10/11/19	6 dies
3.2 Introducció de una primera quantitat de dades per provar	11/11/19	13/11/19	3 dies
3.3 Introducció de la resta de dades	25/11/19	01/12/19	6 dies
4. Prototips			
4.1 Publicació del primer prototip i proves	02/12/19	08/12/19	6 dies
Fita 3: Entrega 3a PAC	08/12/19		
5. Realització de les comprovacions i verificacions de funcionament			
5.1 Proves i verificacions tècniques	09/12/19	11/12/19	3 dies
5.2 Proves i verificacions de funcionalitat	12/12/19	15/12/19	3 dies
5.3 Proves i verificacions amb usuaris	16/12/19	22/12/19	6 dies
5.4 Revisió dels resultats i aplicació de solucions	23/12/19	30/12/19	6 dies
<b>Fase 4: Difusió</b>			
1. Publicació de l'aplicació a un servidor obert al públic	30/12/19	01/01/20	3 dies
2. Difusió a través de les xarxes	02/01/20	05/01/20	1 dia

3. Seguiment del llançament i resposta dels usuaris	02/01/20	05/01/20	3 dies
4. Revisió i redacció final de la memòria	23/12/19	06/01/20	11 dies
Fita 4: Entrega final	06/01/20		

Algunes d'aquestes fases s'han produït de manera simultània, tal com es representa al diagrama de Gantt de la pàgina següent. A més, de forma paral·lela s'ha anat avançant en la redacció d'aquesta memòria tot vetllant pel compliment de les fites següents a les dates clau establertes:

- Entrega PAC 1, que inclou tota la fase de planificació: 01/10/2019
- Entrega PAC 2, que inclou la fase de disseny: 30/10/2019
- Entrega PAC 3, que inclou la primera part de producció: 08/12/2019
- Entrega final, que inclou la part final de la producció i la part de difusió: 06/01/2020

A continuació, s'hi inclou el Diagrama de Gantt elaborat amb el programa Microsoft Project, on s'han contemplat les diferents, fases, subfases, tasques i fites que apareixen al projecte. S'han tingut en compte, també, els dies en que el projecte està aturat.

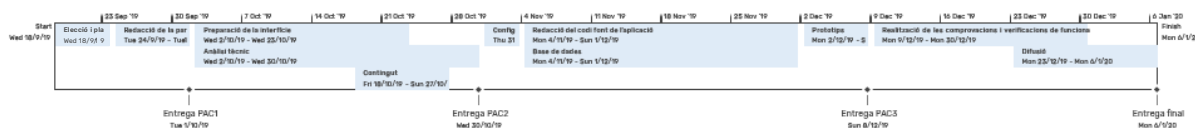
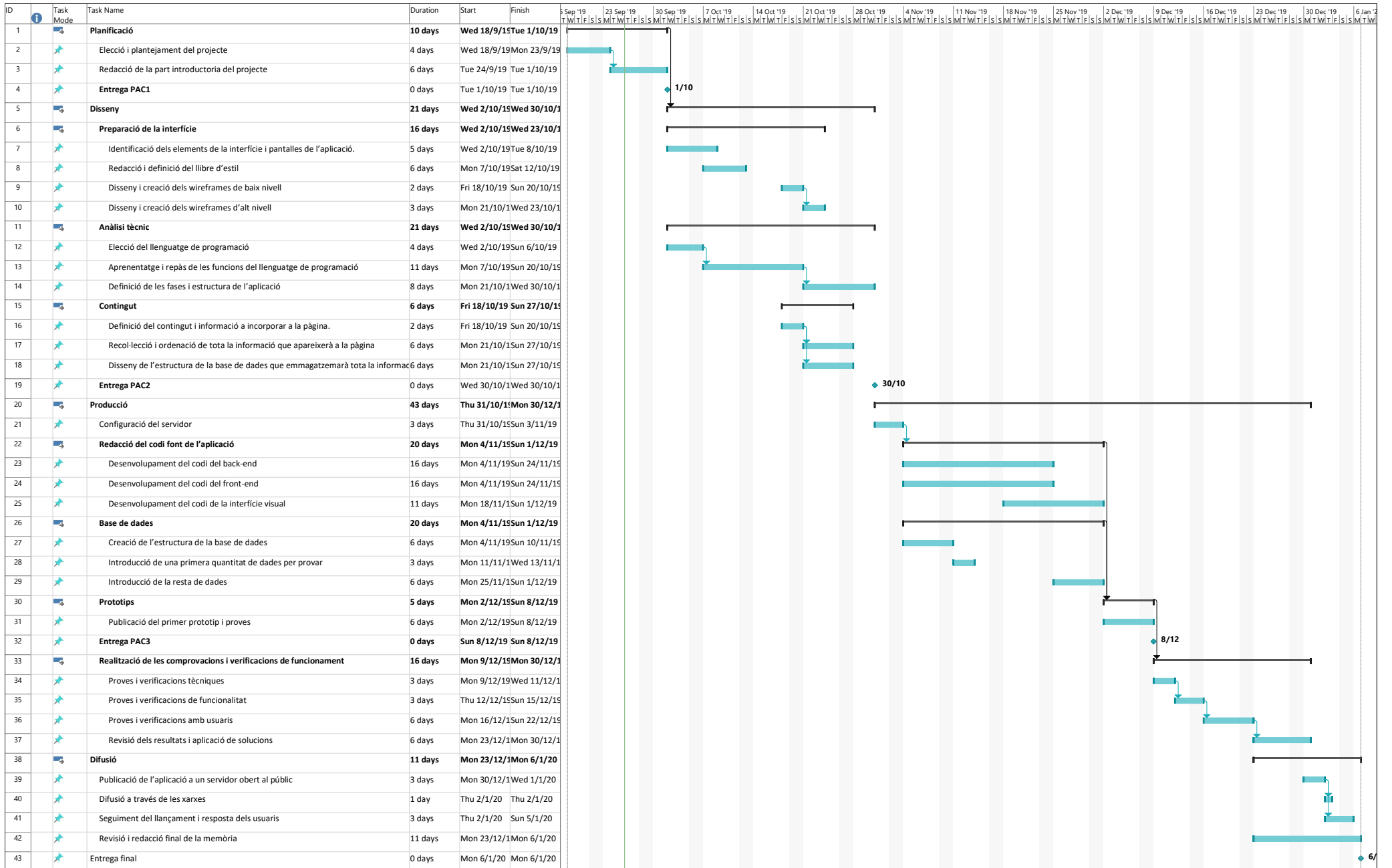


Figura 9: Línia del temps de les fases del projecte



Project: Planificació TFG  
Date: Thu 26/9/19

Task	Summary	Inactive Milestone	Duration-only	Start-only	External Milestone	Manual Progress
Split	Project Summary	Inactive Summary	Manual Summary Rollup	Finish-only	Deadline	
Milestone	Inactive Task	Manual Task	Manual Summary	External Tasks	Progress	



# 11. Procés de treball/desenvolupament

En aquest apartat, s'explica a grans trets el procés de treball en els diferents àmbits que contempla el projecte.

## 11.1 Preparació del servidor

Per executar el projecte en un servidor web, primer de tot, s'han hagut d'instal·lar tots els paquets necessaris per al seu correcte funcionament. El sistema operatiu del servidor és Linux Debian 9, i per tant, la instal·lació del diferent programari s'ha fet a través d'*apt-get*. Ha calgut realitzar la instal·lació d'Apache, MySQL i PHP, per una banda, i de NodeJS i el gestor de paquets NPM, per l'altra.

## 11.2 Creació de la base de dades

La base de dades és una part fonamental pel projecte ja que emmagatzema tota la informació que després s'emprarà per mostrar-la a l'usuari. S'ha treballat utilitzant la interfície que proveeix phpmyadmin, que permet gestionar la base de dades d'una forma més senzilla i amigable de cara a l'usuari. S'ha creat tota l'estructura seguint el diagrama UML de la base de dades.

S'han creat totes les taules (entitats) amb els seus corresponents atributs. S'ha tingut especial consideració en la creació de les claus foranies per relacionar camps entre dues taules diferents. D'aquesta manera, s'han pogut establir les relacions fortes entre taules i això provoca que, per exemple, quan eliminem una línia introduïda a la base de dades, s'esborren automàticament les entrades amb les parades de la línia que estan guardades a la taula EfectuaParada.

Un cop creades totes les taules i entitats, s'han introduït unes quantes dades de prova per poder realitzar testos i comprovar-ne que la connexió funcioni de forma conforme un cop s'avanci en la creació del backend i la seva corresponent API. Les peticions a la base de dades es criden amb llenguatge SQL, específic per l'ús d'aquest tipus de bases de dades.

## 11.3 *Backend*: creació de la REST API

La obtenció de les dades s'efectua a través del *backend* o API que s'ha creat per efectuar les consultes que es realitzin des de la interfície del *frontend* cap a la base de dades. De fet, les APIs s'han convertit en una forma comuna per compartir informació, seguint el principi de consulta i resposta. En aquest projecte s'ha elaborat una REST API a partir del llenguatge PHP pur.

REST són les sigles de Representational State Transfer i és quan l'API treballa sobre el protocol HTTP per intercanviar la informació (Guzman, 2016). Són les capçaleres les que

indiquen el tipus de resposta que es rep (GET, POST, PUT...) o si la petició ha estat satisfactòria (codi 200 HTTP) o no (codi 400). La resposta se sol generar en format JSON i aquest n'és el cas en aquest treball. No obstant, també es podria generar la sortida en format XML o buscar-ne un d'alternatiu.

Per comprovar si la REST API funciona satisfactòriament, s'ha utilitzat el programa **Postman** que permet fer consultes amb els diferents valors que es transmeten i s'obté la resposta.

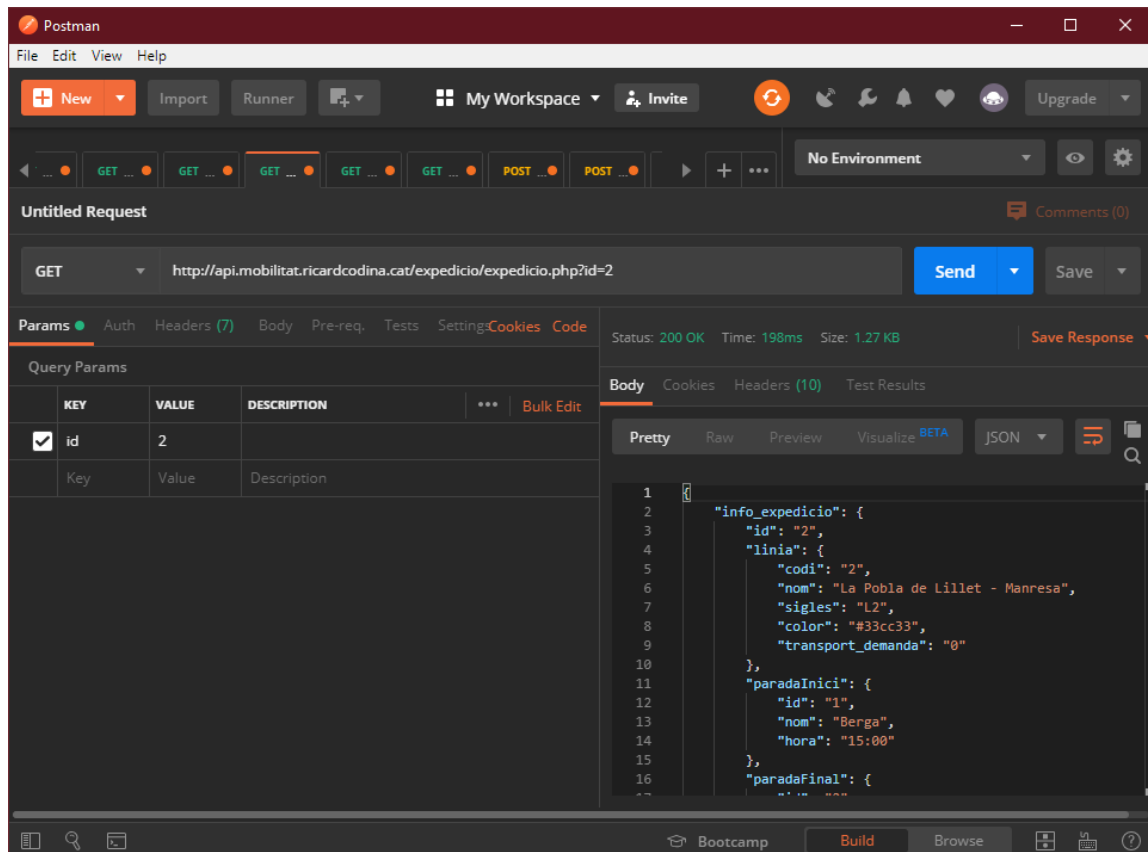


Figura 10 Captura del programa Postman amb una petició GET a la REST API

### 11.3.1 Creació de classes

Seguint el paradigma de la programació orientada a objectes, per tal de crear la REST API, s'han creat classes equivalents a les entitats de la base de dades. Aquestes classes contenen, per una banda, les variables que emmagatzemaran (corresponents als atributs de les entitats) i els diferents mètodes que generaran la informació referent a l'element. Per exemple, el mètode `d'obteDades()`, agafa de la base de dades els atributs restants a partir del id de l'objecte creat, que ha de ser assignat prèviament. En ser mètodes públics, aquestes funcions es poden cridar des de les pàgines de l'API, com s'explica més endavant.

```

// funcio readOne
// carrega informació d'una línia concreta
public function obteDades(){
    $query = "SELECT * FROM Linia A
            WHERE
                A.codi = ?
            ";
    $stmt = $this->conn->prepare( $query );

    // assignarà el valor de id que rebrà a l'interrogant que
    $stmt->bindParam(1, $this->codi);
    $stmt->execute();

    //Desmuntem les dades rebudes i les assignem a les diferents
    $row = $stmt->fetch(PDO::FETCH_ASSOC);

    $this->nom = $row['nom'];
    $this->sigles = $row['sigles'];
    $this->color = $row['color'];
    $this->transport_demanda = $row['transport_demanda'];
    $this->avisos = $this->obteAvisos();
}

```

Figura 11 Funció obteDades() de la classe Linia

Aquestes peticions es fan amb llenguatge SQL cap a la base de dades, utilitzant el PHP Data Objects, que és una interfície que incorpora el llenguatge PHP, des de la seva versió 5.1, per realitzar consultes a bases de dades (The PHP Group, 2001-2019). La configuració es troba a l'arxiu *database.php* dins de la carpeta *config*. Una altra característica és que s'amaguen els valors variables que es passaran durant la consulta a la base de dades per prevenir la injecció de SQL (veure punt 16.1.2).

Dins d'aquestes classes, es troba la classe Lloc que s'encarrega de fer les peticions a l'API de Google Maps per obtenir les coordenades del lloc que passa l'usuari. El seu funcionament s'explica més detalladament dins del punt 12 d'aquesta memòria.

### 11.3.2 Creació de les peticions

Per tal de generar les peticions que retornaran una resposta amb la informació sol·licitada s'ha creat un arxiu php per cadascuna d'elles. Aquests arxius retornaran una resposta o una altra en funció dels arguments passats a través de GET (en el cas de les peticions GET) o com a objecte (amb la resta de peticions). En tot cas, l'estructura és similar. A l'inici del fitxer, s'hi inclouen les capçaleres que necessitem de l'API (on s'hi especifica el tipus) i les classes necessàries per generar la resposta.

```

<?php
// Capçaleres per tal que funcioni l'API
header("Access-Control-Allow-Origin: *");
header("Access-Control-Allow-Headers: access");
header("Access-Control-Allow-Methods: GET");
header("Access-Control-Allow-Credentials: true");
header('Content-Type: application/json');

// Agafa l'objecte que genera l'API i les dades de la BD
include_once '../config/database.php';
include_once '../objects/parada.php';

```

Figura 12 Capçaleres de la petició GET de la informació d'una parada

Tot seguit, es mira que la petició rebuda contingui els arguments obligatoris per poder generar la resposta. Per exemple, per obtenir la informació d'una parada en concret, hem de saber el seu id per tal que pugui retornar una resposta positiva. En cas dels arguments opcionals, els assignem a la seva variable si hi són, o n'assignem per defecte, si no es transmeten.

En PHP la resposta es genera en una tupla o *array*. Per tant, aquest fitxer php s'encarregarà de generar la tupla de resposta a partir dels mètodes i variables de les classes que s'han definit en el punt anterior. Aquesta array es transforma en format JSON utilitzant el mètode genèric de PHP `json_encode()`. Igualment, passem el codi HTTP de resposta corresponent segons el cas, i un missatge que especifiqui el motiu de l'error, en cas que n'hi hagi.

En el cas de les peticions POST o PUT, la resposta correspon a si l'acció ha sigut satisfactòria o no i per obtenir les dades es fa el procés invers. Les dades s'han d'enviar en format JSON i és el mètode `json_decode`, qui converteix l'objecte rebut en una tupla d'on podem consultar les seves dades. Es passa a detallar dos comportaments concrets de l'API, un GET i un POST. A més, les peticions que es realitzen des del panell d'administració disposen d'un sistema d'autenticació en base a un token JWT (s'amplia al punt 16.1).

### **11.3.3 Petició GET d'itinerari**

Una de les funcionalitats més laborioses del projecte és la que genera l'itinerari entre dos punts. En aquest cas, l'aplicació rep quatre dades: el punt d'inici, el punt de destinació (que no tenen perquè ser una parada concreta, pot ser una adreça del municipi), una data i una hora. A més, es poden limitar els resultats que es retornaran.

A partir d'aquí, es trosseja l'itinerari en tres parts: de l'origen a la parada més pròxima al mateix, el recorregut de la parada d'origen a la parada de destinació, i de la parada de destinació al punt de destinació. Primerament, s'ha de localitzar la parada més pròxima als dos punts. Això s'aconsegueix comparant les latituds i longituds dels punts especificats (s'obtenen gràcies a l'API de Google Maps) i es demana a la base de dades si hi ha alguna parada, les coordenades de les quals, siguin pròximes a les dels punts. Si no hi ha cap parada pròxima al punt especificat, no es genera l'itinerari i retorna error.

Un mètode de la classe `Lloc`, permet generar els passos a peu des dels punts a les seves respectives parades més pròximes. Aquests passos i la línia s'incorporen a la resposta JSON que es genera.

El recorregut entre dues parades disposa d'un mètode propi que incorpora la classe `Itinerari`. El seu funcionament és el següent:

1. Mira si les dues parades tenen alguna línia en comú, per tant, estarien enllaçades amb un autobús entre les dues parades. Si no, s'hauria de fer transbord entre línies i

per tant s'executaran més ordres durant la consulta. Troba les línies en comú fent una exploració de les línies de la parada origen i comparant-les amb les de la parada destinació.

2. Consulta les expedicions que passin per la parada origen i que tinguin parada a la parada destinació, a partir de la data i hora especificats. El programa també consulta quin dia de la setmana és la data especificada i si correspon o no en festiu.
3. Un cop localitzades les expedicions que compleixen els criteris, en limita els resultats al valor sol·licitat per l'usuari i prepara un array de sortida per a cada expedició que conté la informació de la línia de l'expedició, l'expedició concreta, les dades de les parades i l'hora de pas per les mateixes.
4. En cas que hagi de fer transbord, el programa efectua uns quants passos més. A partir de les línies que paren a cada estació, el mètode les recorre buscant les parades de cada línia. Després compara si entre les parades d'una línia i de l'altra n'hi ha alguna de coincident (estació de correspondència). Si hi és, es genera l'itinerari en dues parts, primer des de l'estació origen fins a l'estació correspondència i després des de l'estació correspondència fins a la destinació. Es té en compte, que per generar la segona part, el programa té en compte l'hora en que s'arriba a l'estació de correspondència i després busca les expedicions que surten a partir d'aquesta hora d'arribada.
5. Les opcions i parts de l'itinerari es posen entre [ ] a la resposta JSON per tal que el frontend ho pugui interpretar com a Array i generar la interfície a partir del número d'opcions que rep.

#### **11.3.4 Petició POST dels avisos d'usuari**

L'única petició de tipus POST que es pot fer de forma pública és la d'afegir els avisos d'usuari. Aquesta petició permet introduir els avisos o incidències que detecten els usuaris de l'aplicació. Per evitar el mal ús d'aquests avisos, els arguments que rep són limitats i no contenen text personalitzat. Concretament, per generar la petició, es rep el número d'expedició, la parada on ha passat l'avís, el tipus d'avís (n'hi ha 3, concretament) i el retard, si és el cas.

Les dades es reben a través d'un objecte JSON que primerament PHP obté, utilitzant el mètode `file_get_contents()`, i després descodifica amb `json_decode()`. Si totes les dades s'han obtingut de forma correcta, el sistema crida al mètode `afegir()` de la classe `AvisUsuari`, que envia l'ordre SQL d'afegir amb les dades que s'han rebut.

```

public function afegir()
{
    $query = "INSERT INTO AvisUsuari
            SET idExpedicio=:idExpedicio, idParada=:idParada, tipusAvis=:tipusAvis, retard=:retard";
    $stmt = $this->conn->prepare( $query );

    //Assignem els valors amagats de la petició
    $stmt->bindParam(":idExpedicio", $this->idExpedicio);
    $stmt->bindParam(":idParada", $this->idParada);
    $stmt->bindParam(":tipusAvis", $this->tipusAvis);
    $stmt->bindParam(":retard", $this->retard);

    //Executem la consulta, si va bé retorna true, si no, false.
    if($stmt->execute()){
        return true;
    }

    return false;
}

```

Figura 13 Mètode afegir() de la classe AvisUsuari

Finalment, tant si el procés ha anat bé com no, retorna un missatge i el corresponent codi HTTP. És important destacar que les capçaleres són lleugerament diferents, ja que hem d'especificar que el mètode és POST i, a més, ja que hem allotjat l'API en un subdomini, abans d'executar el mètode POST, el navegador fa una petició OPTIONS per saber si realment és pot cridar des del domini des d'on s'està consultant la pàgina. D'aquest fet se'n parla amb més detall al punt 16.1.1.

## 11.4 Frontend: Disseny de la interfície

La interfície està desenvolupada gràcies al framework React que permet generar *frontends* potents i dinàmics amb una certa base de codi. Per iniciar el projecte, s'ha fet servir l'script *create-react-app*, que genera els arxius inicials per tal que l'aplicació pugui funcionar. Dins del paquet de l'aplicació, es troben tres carpetes: *src*, on hi ha el codi que executa react; *public*, on s'allotgen els arxius accessibles a través de l'adreça del servidor; i *node\_modules* on es troben els paquets que fan funcionar React i les diferents extensions que podem instal·lar-hi.

```

116.202.28.14 - PuTTY
cd mapamobilitat
npm start

Happy hacking!
root@debian-8gb-bifidus:/var/www/html/nodejs/mapamobilitat# cd mapamobilitat/
root@debian-8gb-bifidus:/var/www/html/nodejs/mapamobilitat/mapamobilitat# cd ../
cd: error retrieving current directory: getcwd: cannot access parent directories
: No such file or directory
root@debian-8gb-bifidus:/var/www/html/nodejs/mapamobilitat/mapamobilitat/..# cd
./
./
cd: error retrieving current directory: getcwd: cannot access parent directories
: No such file or directory
root@debian-8gb-bifidus:/var/www/html/nodejs/mapamobilitat/mapamobilitat/../../#
root@debian-8gb-bifidus:/var/www/html/nodejs/mapamobilitat/mapamobilitat/../../#
cd /var/www/html/nodejs/
root@debian-8gb-bifidus:/var/www/html/nodejs# create-react-app mapamobilitat

Creating a new React app in /var/www/html/nodejs/mapamobilitat.

Installing packages. This might take a couple of minutes.
Installing react, react-dom, and react-scripts...
[.....] | fetchMetadata: sill resolveWithNewModule @typescript-esl

```

Quan s'executa l'aplicació, el primer arxiu que es crida és *index.js* i aquí comença a renderitzar els diferents subcomponents que té per sota seu..

### 11.4.1 Components

Els components estan distribuïts i emmagatzemats en carpetes segons el seu ús en les diferents parts de la aplicació. D'aquesta manera, és possible localitzar-los de forma més eficient. No obstant, això no vol dir que no els puguem utilitzar en altres parts de l'aplicació, si no, que són accessibles en qualsevol moment. Aquest és un dels avantatges de l'ús de React en el desenvolupament del projecte, ja que podem treballar per mòduls independentment i després es poden utilitzar en diferents parts de la pàgina.

Els components de l'aplicació s'han escrit en forma de classe (a React també tenim l'opció de fer-ho en forma de mètode<sup>15</sup>) i ha de tenir un mètode render() que retorni una sèrie d'elements i expressions JSX, un llenguatge que combina elements d'HTML i JavaScript per crear comportaments dinàmics. En tractar-se d'una classe javascript, aquesta pot tenir altres mètodes que realitzin accions o generin més contingut i ser cridades des de render. Aquest render() s'executa cada cop que es produeix un canvi en l'estat o els elements del component, de forma transparent per a l'usuari de l'aplicació.

Els components importen en la seva capçalera els diferents elements que s'utilitzaran en el propi component. Es carreguen les llibreries externes (axios, react-google-maps,...) que s'utilitzin segons el cas i els altres components propis de l'aplicació que requereixi. React ens permet que un component estigui format per altres components i aquests es criden com si fos una etiqueta de JSX.

```
import React, {Component} from 'react';
import axios from 'axios';
import Carregant from './estructura/Carregant';
import {Container, Row, Col} from 'react-bootstrap';
import DetallsLinia from './linies/DetallsLinia';
import MapaLinia from './linies/MapaLinia';
import RecorregutLinia from './linies/RecorregutLinia';
import HorarisLinia from './linies/HorarisLinia';
import LlistaTotesLinies from './linies/LlistaTotesLinies';
import AvisosGrup from './avisos/AvisosGrup';
```

Figura 14 Components i llibreries externes que carrega el component que genera la pàgina d'una línia

### 11.4.2 Estats

Cada component pot tenir el seu propi estat (*state*, en anglès) que consta d'una sèrie de variables que es poden consultar i modificar durant l'execució de l'aplicació. Per consultar-les fem servir `this.state` seguit del nom de la variable que es vol cridar i per modificar-les s'utilitza la funció `this.setState()`. Com s'ha explicat a l'apartat anterior, el mètode `render()` s'actualitza cada cop que varia aquest estat del component, per tant, les dades rebudes des

<sup>15</sup> <https://reactjs.org/docs/components-and-props.html>

del backend, amb axios (veure apartat 12), es transmeten a l'estat del component i aquest actualitza els elements que genera el mètode render().

```
obteDades() {
  this.setState({generaInterficie: false }); //Abans de carregar dades, posem la careta de Carregant.
  this.noID=false;
  const id = Number(this.props.match.params.id);
  const baseUrl = `https://api.mobilitat.ricardcodina.cat/parades/parada.php?id=${id}`;

  axios.get(baseUrl)
  .then(res => {
    const dades = res.data.info_parada;
    const linies = res.data.linies;
    const avisos = res.data.avisos;
    const properesSortides = res.data.properesSortides;
    this.setState({ dades, linies, properesSortides, avisos, generaInterficie: true });
    document.title = dades.nom;
  })
}
```

Figura 15 Mètode obteDades() del component parada. Es veu la crida a l'API a través d'axios i, si aquesta ha resultat satisfactòria, s'assignen les dades rebudes a l'estat del component

Els elements de formulari, com els del panell d'administració o el que envia un avís generat per l'usuari, també enregistren els seus valors en un estat del seu component. Quan s'executa l'ordre POST de la REST API genera un objecte amb els valors de l'estat i els transmet en format JSON. Aquests estats (i els seus mètodes per modificar-los) també es poden transmetre a través de propietats, com s'explica en el següent apartat.

Un altre ús dels estats que s'ha donat en aquest projecte és pel que fa a la visualització o no dels diferents elements que genera el component. Això s'ha aconseguit combinant l'ús de estats amb valors numèrics o booleans amb expressions com if ... else. La finestra flotant que apareix per tal que l'usuari pugui informar d'una incidència es comporta segons el valor de l'estat que la controla.

```
state = {
  finestraOberta: false,
  avisos:null
}

canviEstatFinestra() {
  this.state.finestraOberta ? this.setState({finestraOberta:false}) : this.setState({finestraOberta:true});
}
```

Figura 16 Estat del component AvisosUsuari. Disposem d'un mètode per alterar-ne el seu valor, el mètode s'assigna a un botó. Quan canvia l'estat, apareix o desapareix la finestra que conté un dels subcomponents.

### 11.4.3 Propietats

Quan es fa una crida a un component de react se li poden assignar propietats (*props*, en anglès). Aquestes propietats no són específiques i poden utilitzar el nom que es prefereixi. Un cop dins del component es poden cridar aquestes propietats que s'han rebut, fent servir `this.props` seguides del nom de la propietat rebuda. Podem passar variables i fins i tot mètodes entre components.

Un exemple de l'ús d'aquestes propietats al projecte el trobem al mapa que genera l'itinerari que necessita les dades de les parades, els recorreguts a peu i la informació de les



expedicions que utilitza. Aquestes dades provenen del component Resultat i ens transmeten a través de les propietats que s'afegeixen quan es crida al component.

```
<MapaSuperior
  dades={this.state.resposta.dadesItinerari}
  itinerari={this.state.resposta.itineraris.opcions[this.state.itinerariVisible][0]}
  itinerariAParadaOrigen={this.state.resposta.itinerariAParadaOrigen.routes[0]}
  itinerariAParadaDestinacio={this.state.resposta.itinerariDesParadaDestinacio.routes[0]}/>
```

#### 11.4.4 Ús dels estats i propietats

El projecte està estructurat per un conjunt de components i els seus respectius subcomponents. Alguns d'aquests subcomponents es generen a partir de la informació que tenen els components pare i que la transmeten a través de les propietats. Si alguna d'aquestes propietats es referen a l'estat del component pare, quan s'actualitza aquest component, la informació que rep el subcomponent canvia i, per tant, es torna a renderitzar. Igualment, com que l'estat es pot transmetre a diversos subcomponents a la vegada, quan canvia l'estat es poden actualitzar diversos components al mateix moment.

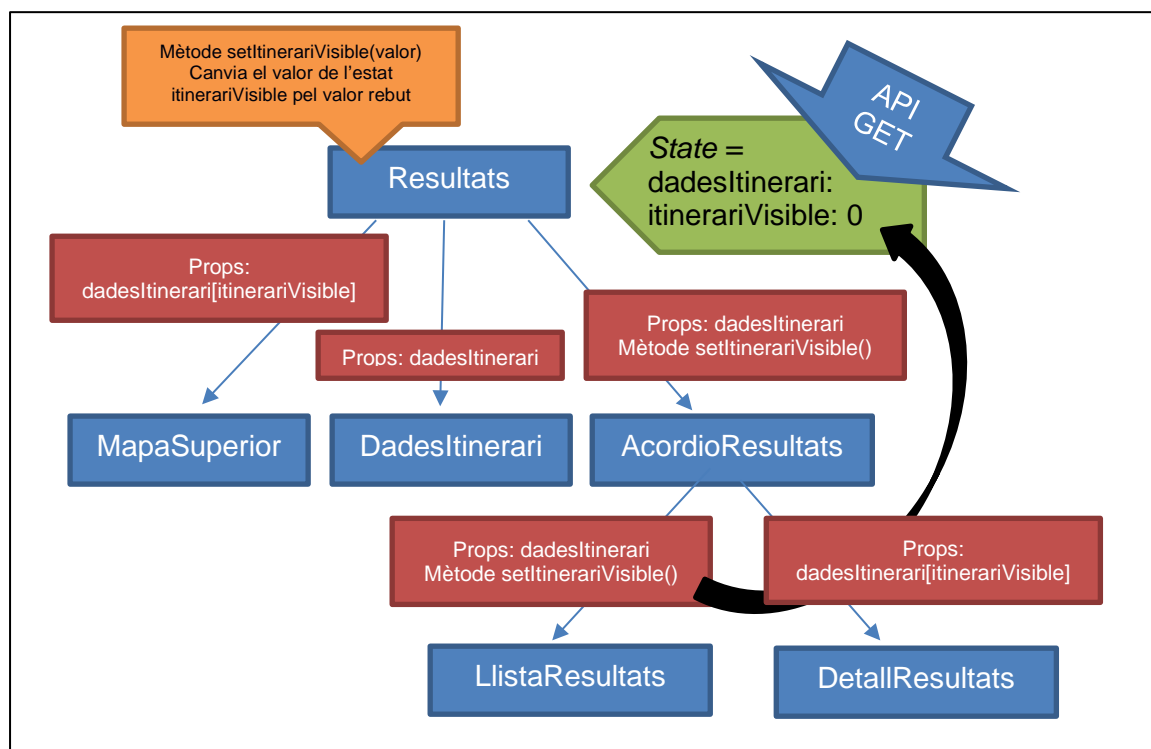


Figura 17 Funcionament dels estats i les propietats al component Resultats. Els estats que controlen els diferents subcomponents estan en el component Resultats. Aquest estats es transmeten a través de les propietats. En alguns casos, només transmetem la informació que volem mostrar en cada moment (el mapa només mostra l'itinerari seleccionat en cada moment). A la llista de resultats és on l'usuari pot escollir quin itinerari vol veure en cada moment, per això li passem el mètode setItinerariVisible() que modificarà l'estat del component Resultats, ja que actua sobre l'estat del component des d'on es transmet, no en el que s'aplica. Com que l'estat canvia, la informació que es transmet també, per tant, el mapa i el component de detall, mostraran una nova informació, només la del nou itinerari seleccionat.

Aquest funcionament per components també ens permet que un dels components es programi una vegada i agafi les dades que rep a través de les propietats. D'aquesta forma es pot utilitzar en diferents parts de l'aplicació. És el cas del mòdul d'Avisos que rep els avisos que ha de generar en forma de tupla però les dades d'aquesta s'han obtingut en el component que l'implementa. Això ho he fet per reduir el nombre de consultes a l'API i per

generar-ho tot en una sola petició. Així doncs quan es consulta una parada, l'API també retorna la informació sobre els avisos, que s'inscriu a l'estat del component principal i, aquest estat es transmet al component d'Avisos a través de les propietats.

#### 11.4.4 Navegació

Per recórrer la navegació entre les diferents pàgines de l'aplicació s'ha fet ús del component Router que inclou la llibreria externa 'react-router-dom'. Aquest component ens permet renderitzar uns components o uns altres segons la ruta que tingui establerta el navegador. A més, podem assignar paràmetres escrits a la URL que després es passen a través de les propietats als diferents components. Això s'observa a les pàgines de les parades, les línies i les expedicions, que es generen segons la id que s'escriu a la URL del navegador. No obstant, s'ha preparat cada component de forma que també mostri un missatge si no es passa cap paràmetre a través de la URL.

```
<div className="App">
  <Top/>
  <Route exact path="/" render={() => ( <Portada canviOrigen={this.canviOrigen} canviDestinacio={this.canviDestinacio} />)} />
  <Route exact path="/llistaParades" render={() => ( <LlistaParades/>)} />
  <Route exact path="/resultats" render={() => ( <Resultats itinerari={this.state.itinerari}/>)} />
  <Route exact path="/mapa" render={() => ( <Mapa/>)} />
  <Route path={['/parada/:id', '/parada']} render={(params) => ( //passem els valors de la URL a través de params
    <Parada {...params} />
  )} />
  <Route path={['/linia/:id', '/linia']} render={(params) => (
    <Linia {...params} />
  )} />
  <Route path={['/expedicio/:id', '/expedicio']} render={(params) => (
    <Expedicio {...params} />
  )} />
</div>
```

Figura 18 Diferents rutes que conté el component Main per tal de generar cada pàgina concreta a partir de la URL passada

#### 11.4.5 Descripció dels components

Es passa a realitzar una breu descripció dels components principals que formen part del *frontend* de l'aplicació.

##### PORTADA

La portada disposa del cercador d'itineraris, un mòdul que mostra els avisos de les parades i les línies i un apartat amb accés ràpid a les línies i parades més importants. El cercador d'itineraris consta d'un petit formulari amb l'opció de posar origen, destinació i data i hora del viatge. L'usuari pot escollir des de la llista de parades que li suggereix o escriure la seva pròpia adreça o lloc d'interès.

##### RESULTAT D'ITINERARI

Carrega els resultats de l'itinerari rebuts des de l'API. L'usuari pot seleccionar l'itinerari que vulgui veure entre tres propostes, per defecte, s'escull la primera, més pròxima a la data i hora escollits. Amb l'itinerari escollit es carrega el mapa de l'itinerari, amb les tres parts de l'itinerari (els dos trajectes a peu i el trajecte amb transport), i els detalls de l'itinerari

seleccionat. El *frontend* s'encarrega de mostrar només la informació de l'itinerari escollit per l'usuari.

## MAPA

El mapa mostra les línies i parades que estan introduïdes a la base de dades i la posició teòrica dels autobus que estan circulant en cada moment. L'usuari pot mostrar i amagar les línies i parades que desitgi utilitzant el menú lateral. Aquests canvis es realitzen sobre l'estat del component que es transmet al mapa i fa aparèixer o desaparèixer les diferents icones. Es pot també utilitzar el camp de text per filtrar les parades a partir d'una part o del nom sencer d'una parada. A més, hi ha la possibilitat per l'usuari de geolocalitzar-se i d'accedir a la descripció de les diferents línies, parades i expedicions.

## LINIES, PARADES I EXPEDICIONS

L'estructura d'aquests components és força similar entre ells. Es generen a partir del codi de l'element en qüestió. Implementen un mapa i la informació més destacada a través dels mòduls i de les dades que reben de l'API. S'ha fet servir una sèrie d'elements Link, que formen part de la llibreria 'react-router-dom', per relacionar uns elements amb els altres i poder saltar entre les diferents parts de l'aplicació per ampliar la informació.

Es destaca la taula horària del component línies que genera una taula segons les opcions que selecciona l'usuari: la direcció de la línia i el dia de la setmana concret. Es rep tota la informació de les expedicions i utilitzant iteracions es va generant cada fila i columna de la taula.

```
dibuixaFilaHorari(parada, index){
  if (index!=0){
    return(
      <tr>{parada.map((expedicio,index)=><td>{expedicio?expedicio:'-'}</td>)}</tr>
    )
  }

  if (index==0){ //Si rebem la primera línia, conté els codis de les expedicions
    return(
      <tr><td></td>
      {parada.map((expedicio,index)=><LinkContainer to={"/expedicio/"+expedicio}><td className="maBoto">{img
        src={iconaBusBlanca}
        width="20"
        height="20"
        className="iconaBus"
        alt="Expedicio"
      }</td></LinkContainer>)}</tr>
    )
  }
}
```

Figura 19 Funció `dibuixaFilaHorari` del component `TaulaHorari` que genera la informació de l'horari de cada parada per implementar-la a la taula horària del component `Linia`

## PANEL·L D'ADMINISTRACIÓ

El panell d'administrador consta de les pàgines necessàries per introduir, modificar o eliminar les dades que es poden consultar a l'aplicació. Per realitzar l'autenticació de

l'usuari, s'han d'omplir els camps de correu electrònic i contrasenya, que apareixen si s'intenta accedir al panell i no s'ha iniciat la sessió. Quan s'envien les dades a l'API, si l'usuari és correcte, es retorna el token corresponent per poder utilitzar les consultes POST, PUT i DELETE de l'API. El token s'emmagatzema a l'estat del component i es passa per propietats als altres subcomponents per tal que puguin realitzar les peticions a l'API. Aquest token és vàlid durant una hora.

Es disposa d'un menú lateral per accedir als diferents tipus de dades que es poden modificar: Parades, Línies, Expedicions i Avisos. El disseny dels diferents apartats és molt similar entre ells. Es visualitza una llista dels elements que s'ha seleccionat. Cada element es pot editar o eliminar. Si s'edita, apareix un formulari amb totes les dades actuals, que s'assignen a l'estat del component i l'usuari les pot modificar. Quan s'accepten els canvis, aquests s'envien a l'API a través d'una petició PUT.

Si es vol afegir un nou element, es disposa d'un botó que obre un formulari en blanc. L'usuari ha d'introduir totes les dades necessàries, i en enviar-les es realitza una petició POST. Es mostrarà un missatge de confirmació o d'error, segons com hagi transcorregut la petició.

#### **11.4.6 Estil i aparença de l'aplicació**

Per plasmar l'estil dels prototips d'alt nivell cap al resultat final s'ha fet servir dos elements principals. Per una banda, la llibreria Bootstrap i per l'altra, la fulla d'estils CSS.

La llibreria Bootstrap és una col·lecció d'elements ja predissenyats molt popular a la xarxa. Elements com botons, missatges d'alerta, camps de formulari però també altres elements més dinàmics com carroussels, o acordions d'elements, estan ja programats per poder-los utilitzar. De fet, Bootstrap és una llibreria per a Javascript amb JQuery<sup>16</sup> però se n'ha fet una versió per a React<sup>17</sup>. D'aquesta manera es pot aprofitar la potència del treball amb components del framework amb els elements preestablerts de Bootstrap.

En el projecte, l'ús d'aquesta llibreria ha sigut molt important per l'elaboració dels diferents elements adaptatius per a diferents pantalles. Es posa el cas d'una fila de columnes, en la que en la visualització des d'un ordinador amb pantalla horitzontal es poden veure una al costat de l'altra, en un mòbil (on la visualització habitual és vertical), aquestes columnes han de sortir una sota l'altra.

<sup>16</sup> Es pot consultar la documentació i descarregar a <https://getbootstrap.com/>

<sup>17</sup> S'instal·la utilitzant la consola del servidor i nodeJS, seguint els passos de la pàgina web oficial: <https://react-bootstrap.github.io/>.

```

<Container>
<AvisosGrup dades={this.state.avisos}/>
<br />
  <Row>
    <Col md><ModulLinies/></Col>
    <br />
    <Col md><ModulParades/></Col>
  </Row>
</Container>

```

Figura 20 Codi de l'estructura de la portada. Bootstrap adapta l'ordre de les columnes segons l'amplada disponible.

Ara bé, no tot el disseny de la interfície pel que fa a la seva aparença està fet amb Bootstrap sinó que s'han hagut de introduir modificacions personalitzades. La fulla d'estils CSS (Cascading Style Sheets) conté totes les ordres que afecten als diferents de la pàgina. Algunes ordres s'apliquen a elements genèrics de tota l'aplicació segons la seva etiqueta html (div, p, span, table...) i unes altres fan referència a elements concrets identificats per la seva classe. Amb React, podem assignar una classe als diferents elements utilitzant el prop de `ClassName`.

També s'ha utilitzat diferents comportaments, per alterar la reacció de diversos elements segons el comportament de l'usuari. Per exemple, trobem alguns elements que utilitzen el selector `hover`, com els div que contenen la llista amb els diferents itineraris. Això provoca que quan es passa amb el cursor per sobre, alguna de les seves propietats, com el color de fons, canvia, per fer notar a l'usuari la selecció que pren. Per contra, el `hover` només té sentit, és a dir, s'aprecia, quan s'utilitza l'aplicació amb un ratolí, ja que als dispositius tàctils la opció és prem directament i no passa per l'estat de `hover`.

Al llibre d'estil de l'aplicació (veure annex) es pot veure que s'han de seguir uns criteris respecte als colors i tipografies. No obstant, aquests es podrien modificar en un futur, en cas que així es requerís. Pel que fa als colors, aquests s'utilitzen en diferents normes del full d'estil. Es podria haver assignat el color concret cada cop que es requerís però s'ha considerat utilitzar variables CSS per realitzar aquesta funció. Les variables es declaren al principi del full i es poden utilitzar en el mateix. Així, si es modifica el valor de la variable, repercutirà en tots els elements que la utilitzen.

```

:root {
  --color-primary: #336122;
  --color-secondary: #B364A9;
  --seleccio-hover: #e5ffdb;
  --seleccionat: #8fa387;
}

```

Figura 21 Variables del full d'estil CSS

La tipografia s'obté del catàleg de **Google Fonts**, d'aquesta manera s'allibera una mica de càrrega del servidor, ja que s'obtenen els tipus de lletra de forma externa. La crida es fa

amb una línia de codi que s'incorpora a l'arxiu HTML de l'aplicació, l'index.html a la carpeta public.

```
<!-- TIPOGRAFIES -->
<link href="https://fonts.googleapis.com/css?family=Archivo:700|Mukta+Mahee:200,400,700&display=swap" rel="stylesheet">
```

Figura 22 Línia de codi que carrega les tipografies des de Google Fonts

Al codi CSS és on s'assigna cada tipografia als estils que l'utilitzen, seguint els patrons dels wireframes i prototips d'alta fidelitat (apartat 13). També s'han assignat tipografies més genèriques, disponibles a la gran majoria d'ordinadors, que es mostraran si hi ha algun problema per obtenir les tipografies des del servidor de Google Fonts. Les dues tipografies utilitzades són Archivo<sup>18</sup> i Mukta Mahee<sup>19</sup>.

### 11.4.7 Progressive Web App

React s'encarrega de preparar l'aplicació com a Progressive Web App (PWA) en el moment de la compilació. D'aquesta forma l'usuari pot instal·lar-se l'aplicació en el seu dispositiu sigui un terminal mòbil com un ordinador d'escriptori. De fet, navegadors web com Google Chrome porten l'opció de realitzar aquesta esmentada instal·lació i mostra una notificació a l'usuari avisant-lo que té l'opció de fer-ho. La gran part de fitxers queden emmagatzemats a l'aparell on s'ha realitzat la instal·lació i s'actualitza de forma remota. L'actualització es fa forçant un buidatge de la memòria cau, tal i com s'explica al punt 18 de la memòria.

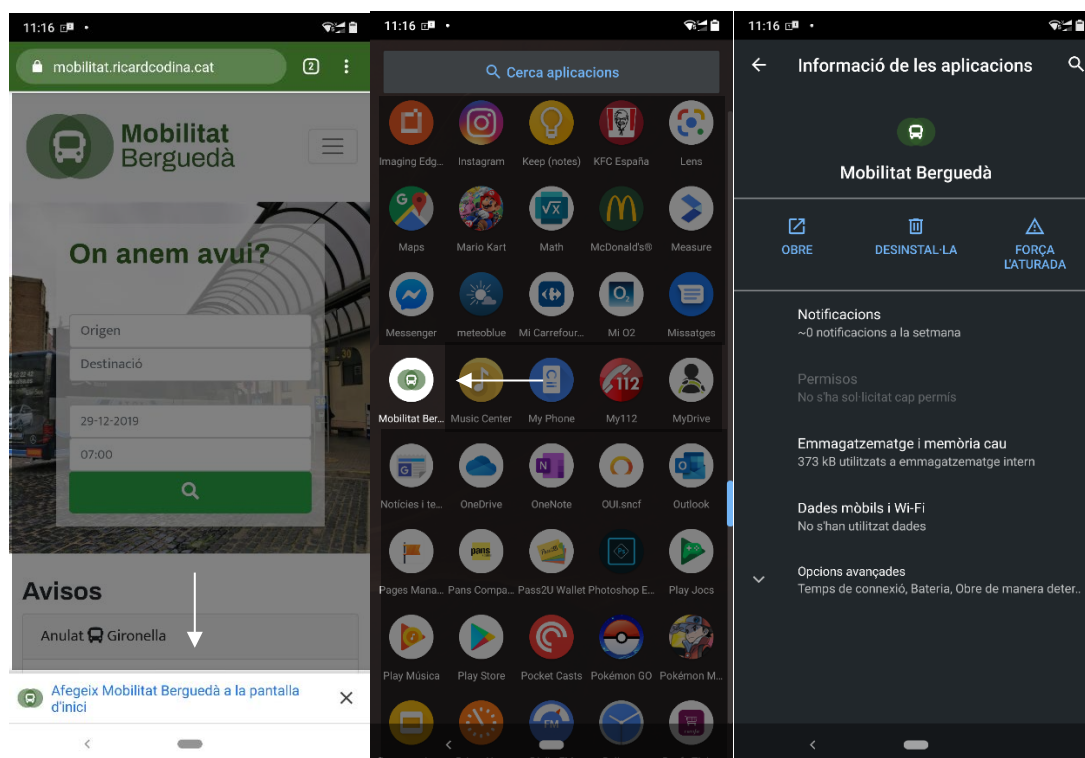


Figura 23 En funcionar com a PWA, l'aplicació mostra a l'usuari un missatge per instal·lar l'aplicació, que queda integrada amb la resta d'aplicacions del telèfon i funciona com a tal.

<sup>18</sup> <https://fonts.google.com/specimen/Archivo>

<sup>19</sup> <https://fonts.google.com/specimen/Mukta+Mahee>

## 12. APIs i llibreries externes utilitzades

Per elaborar el projecte s'han fet servir llibreries externes i crides a APIs de tercers. Aquest és un dels avantatges que té la programació orientada a objectes ja que es poden utilitzar parts de codi ja programat i reutilitzar-los per a altres projectes. Només s'ha de tenir en compte a quines funcions responen i com s'han d'executar i no cal saber de quina manera han estat programades. Amb les llibreries de React, cal saber quines propietats es poden passar i què s'obindrà com a resultat a la interfície.

### 12.1 API de Google Maps

Google Maps és una plataforma de la empresa californiana Google que permet consultar mapes d'arreu del món, obtenir rutes de punt a punt, i consultar una gran base de dades de negocis, punts d'interès i poblacions. Google Maps ofereix un servei amb diferents APIs que es distribueixen a través de Google Cloud Services. Cada petició que es fa té un cost econòmic, tot i que es disposa d'un crèdit mensual de 200€ que es pot utilitzar gratuïtament, sense cost. Es valora que, el total de les peticions que es puguin produir en aquest projecte, no superin el límit gratuït mensual. De les diferents APIs que disposa, en el projecte s'ha fet servir l'api de Google Maps Places i la de Google Map Directions.

L'API de Google Maps Places<sup>20</sup> retorna una resposta JSON amb els detalls que té Google a la seva base de dades sobre l'element sol·licitat. A l'API del projecte, es fa servir a la funció `consultaLloc()`, en la que es passa el nom del lloc a cercar a través dels atributs. PHP s'encarrega de fer la petició a l'API de Google Maps amb el paràmetre passat. A més, he afegit que el lloc que es busca se situï a Catalunya, ja que l'aplicació se situa en aquesta zona.

Cal notar, també, que hem d'afegir la clau d'usuari per poder generar la consulta i així Google sap a qui cal cobrar la petició. L'API ens retorna tot tipus d'informació (Adreça completa, tipus del punt d'interès, coordenades...) i cada classe fa una tria de la que necessita per generar la resposta de la nostra API.

```
function consultaLloc($nom){
    $nomCode=urlencode($nom);
    $url="https://maps.googleapis.com/maps/api/geocode/json?address=".$nomCode.",Catalunya&key=".$GLOBALS["apiKey"];
    $jsonData = file_get_contents($url);
    $ArrayData = json_decode($jsonData, 1);
    return $ArrayData;
}
```

Figura 24 Funció `consultaLloc()` que retorna una Array amb la informació del lloc passat com a atribut. Es pot apreciar la construcció de l'URL de l'API.

<sup>20</sup> <https://developers.google.com/places/web-service/search>

L'altra API que es fa servir és la de Google Maps Directions<sup>21</sup>, en concret en el mètode `generaltinerariAPeu`. En aquest cas, s'ha de passar més informació per generar la resposta: un origen i una destinació, que corresponen a una latitud i longitud concreta, que s'hauran obtingut a partir de la funció anterior; el mètode de transport a utilitzar, que serà a peu (*walking*); i l'idioma en que es genera la resposta, el català.

Com en el cas anterior, aquesta funció retornarà una tupla amb els valors rebuts i descodificat del JSON de Google Maps Directions. Tot el conjunt de dades, s'envia cap al backend, per exemple, en la generació de l'itinerari d'un punt a una estació.

## 12.2 Encoded Polyline

Per traçar una línia sobre el mapa, Google Maps utilitza tota una sèrie de coordenades que representen els diversos punts utilitzats per formar la línia. No obstant, si una d'aquestes línies consta de molts punts, la informació que es transmetrà serà força important. Per aquest motiu, i per reduir aquest grau d'informació, les API de Google Maps transmeten les dades de representació de la línia de forma codificada utilitzant l'anomenat **Encoded Polyline Algorithm Format**<sup>22</sup>, que converteix tots els punts de la línia en una sola línia de resposta a partir de diferents transformacions matemàtiques.

A l'API s'ha fet servir la funció externa **PHP Google Maps Polyline Encoding Tool**<sup>23</sup> que conté tot de mètodes PHP per codificar i descodificar línies en aquest format. En el cas del projecte, s'ha fet servir per generar les línies codificades a partir dels punts rebuts, per exemple, per crear la línia que uneix les diferents parades d'un recorregut.

Com que el backend transmet la línia codificada, el frontend l'ha de descodificar per poder-la representar sobre el mapa. D'aquesta funció se n'ha ocupat la llibreria externa *decode-google-map-polyline* per Javascript<sup>24</sup>, que retorna un objecte amb les seves corresponents latituds i longituds a partir d'una línia codificada. Aquest objecte es pot passar directament al mòdul de Google Maps i queda representada la línia sobre el mateix.

## 12.3 React-google-maps

L'aplicació recorre en diversos moments a l'ús de mapes per mostrar la situació de les parades, el recorregut d'una línia, el passos d'un itinerari... Aquests mapes estan facilitats per Google Maps a través de la seva API. Com en els casos anteriors, l'ús d'aquests mapes també té un cost per visualització i s'explica el seu funcionament en la seva documentació<sup>25</sup>. Per facilitar la feina amb react, també es troba a la xarxa una llibreria per l'ús de Google

<sup>21</sup> <https://developers.google.com/maps/documentation/directions/intro>

<sup>22</sup> <https://developers.google.com/maps/documentation/utilities/polylinealgorithm>

<sup>23</sup> <https://github.com/emcconville/google-map-polyline-encoding-tool>

<sup>24</sup> <https://www.npmjs.com/package/decode-google-map-polyline>

<sup>25</sup> <https://developers.google.com/maps/documentation/javascript/tutorial>



Maps que inclou molts components ja codificats per generar el mapa. Aquesta llibreria és *'react-google-maps'*<sup>26</sup>.

Aquesta llibreria s'instal·la amb el gestor de paquets npm i s'utilitza com s'hi fos un component més de React. Pot utilitzar totes les propietats de la que disposa la pròpia API de Google Maps i només cal emprar correctament els seus subcomponents (marcadors, línies, finestres, etc.).

```
{props.parades.map((parada, index)=>(  
  !parada.esBergueda && (<Marker  
    visible={parada.visible}  
    position={{ lat: Number(parada.latitud), lng: Number(parada.longitud)}}  
    title={parada.nom}  
    key={'paradaID'+parada.id}  
    icon={{url: "https://mobilitat.ricardcodina.cat/img/icona_parada_gris20.png"}}  
    onClick={() => props.obrirDetallParada(index)}></Marker>)))}
```

Figura 25 Expressió dins del component de react-google-maps per generar els marcadors que representen les parades

A més, també inclou tot una sèrie d'esdeveniments per capturar les accions de l'usuari. Per exemple, una de les funcionalitats utilitzades és la de saber quines són les coordenades del punt on ha fet clic l'usuari. Al panell d'administració, això s'utilitza per marcar el punt on s'ubicarà cada parada, quan l'afegim o quan la volem modificar.

## 12.4 Axios

Axios<sup>27</sup> és el nom d'una llibreria externa que permet fer crides HTTP a una direcció i rebre'n una resposta. Al projecte s'utilitza com a enllaç entre el backend i el frontend, tant per obtenir informació com per enviar-ne.

El seu funcionament és força simple, es crida utilitzant diferents funcions. La primera d'elles és **get** (o post, o put, segons la petició que s'hagi de fer a l'API), que té com a atribut l'adreça on fa la crida i l'executa. Després posem **then** que, si la resposta que rep és positiva, s'encarrega d'assignar en variables les dades del JSON rebut, en el cas d'una petició GET, o el missatge satisfactori, en el cas d'altres peticions com POST. Finalment, també hi ha el mètode **catch**, que s'executa si es retorna un error. També podem destriar la resposta rebuda però executar accions diferents a sí la resposta hagués estat positiva.

## 12.5 React-bootstrap

Com s'ha comentat al punt 10.4.6 d'aquesta memòria, Bootstrap és un framework que conté tot d'elements visuals preparats per fer servir. Des de botons a camps de formulari passant

<sup>26</sup> <https://www.npmjs.com/package/react-google-maps>

<sup>27</sup> <https://www.npmjs.com/package/axios>

per missatges d'alerta, acordions o desplegable. Un dels avantatges potents que té és la possibilitat de dissenyar l'estructura de la pàgina de forma que es pugui assolir un disseny responsiu i adaptatiu per a tot tipus de pantalla. La llibreria original està basada en Javascript i jQuery però es disposa de la versió de **React-bootstrap** per poder utilitzar tots aquests elements amb React.

Al projecte s'ha utilitzat aquesta llibreria per poder generar alguns dels elements que apareixen a la interfície i l'estructura per columnes de les diferents parts de l'aplicació. A cada component on s'ha fet servir, s'han inicialitzat només els components de Bootstrap que s'han utilitzat i es criden com si fos un altre component de React. A més, a cada component de Bootstrap se li poden passar *props* específics per personalitzar-ne les seves característiques. Aquestes propietats estan especificades a la documentació de cada element<sup>28</sup>.

## 12.6 React-Router-Dom

Per mantenir l'enllaç entre els diferents Components del projecte i habilitar el comportament dels botons d'anar endavant i enrere del navegador, s'ha utilitzat la llibreria externa React Router Dom<sup>29</sup>. Per una banda, la funció `withRouter` aplicada al component `Main`, fa que el navegador sàpiga en tot moment el camí que segueix l'usuari quan es mou entre diferents parts de l'aplicació. Per l'altra, s'han fet servir els components següents:

- **Router**, per renderitzar el component corresponent segons l'adreça URL introduïda a la barra de direccions.
- **Link**, per crear enllaços interns dins de l'aplicació que només afectin al DOM Virtual i no suposin recarregar l'aplicació completament cada cop que es canviï de pàgina.
- **LinkContainer**, com el component anterior però es pot fer servir per englobar diferents elements a la vegada, com ara altres components o elements que estan dins d'una etiqueta `<div>`.

Els enllaços es generen i funcionen però, a diferència de l'etiqueta `<a>` d'HTML, pels usuaris d'ordinador, el cursor no canvia de forma quan passem per sobre d'un d'aquests elements amb etiqueta `Link` o `LinkContainer`. Per fer aparèixer el cursor de la mà en aquests enllaços, s'ha fet servir una classe específica de CSS que inclou la norma *cursor:pointer*.

<sup>28</sup> <https://react-bootstrap.github.io/components/buttons/>

<sup>29</sup> <https://www.npmjs.com/package/react-router-dom>

## 12.7 Fontawesome

És un conjunt d'icones de tot tipus per utilitzar en pàgines i aplicacions web. Hi ha més de 7.000 icones i pictogrames<sup>30</sup> i una llibreria per React per tal de poder-los incorporar al codi de l'aplicació. En el projecte, s'han fet servir icones amb un significat prou clar per substituir elements que podrien haver estat escrits en forma de text.

La manera en que s'importen aquestes les icones al projecte es realitza de forma diferent a la resta de fotografies. El que s'ha fet ha estat importar les icones necessàries sota el paraigües d'una biblioteca per tal que estiguessin disponibles en qualsevol part de l'aplicació. Aquesta biblioteca s'inicialitza al component App.js i carrega les icones que volem tibar de la llibreria de FontAwesome.

```
import { faCheckSquare, faCoffee, faSearch, faChevronCircleRight, faWalking, faBus, faSync, faCheckCircle, faCircle, faTimesSquare, faHourglassHalf, faTimes, faUsers } from '@fortawesome/react-fontawesome';  
library.add(faBus, faCheckSquare, faCoffee, faSearch, faChevronCircleRight, faWalking, faSync, faBus, faCheckCircle, faCircle, faTimesSquare, faHourglassHalf, faTimes, faUsers)
```

Figura 26 Creació de la biblioteca d'icones de Fontawesome, on s'importen i s'afegeixen al contenidor de la biblioteca

Un cop, s'ha generat la biblioteca, a cada Component on es fa servir una icona, s'ha de cridar al component de FontAwesome i indicar una de les icones que s'han carregat a la biblioteca.

## 12.8 Moment

Moment JS<sup>31</sup> és una llibreria específica per tractar objectes del tipus *Date* amb JavaScript. Permet mostrar les dates i hores en el format desitjat i realitzar operacions entre elles. A més, porta un molt bon nombre de traduccions i localitzacions i és el component extern que permet que el calendari es mostri en català.

## 12.9 Google Analytics

Per fer un control de les visites i l'ús de l'aplicació, s'ha instal·lat el codi de seguiment de Google Analytics. Aquest codi s'afegeix a l'arxiu index.html que conté les etiquetes HTML de la capçalera (*<head>*). S'ha escollit Google Analytics, perquè disposa d'un nou panell de control optimitzat per aplicacions web. Des d'aquest panell es permet controlar les dades habituals com si fos una pàgina web, és a dir, número de visites, usuaris únics, d'on provenen les consultes, les pàgines de l'aplicació més visitades... i incorpora altres funcionalitats orientades a les aplicacions com, per exemple, la captura d'esdeveniments.

També permet monitoritzar en temps real els dispositius que estan fent servir l'aplicació. Es poden visualitzar les dades dels diferents aparells, el temps que porten connectats, on estan situats geogràficament, i com estan utilitzant l'aplicació. L'informe marca en temps real cada

<sup>30</sup> <https://fontawesome.com/icons?d=gallery>

<sup>31</sup> <https://momentjs.com/>

cop que l'usuari canvia de pantalla, prem algun element, o fa *scroll* a través de l'aplicació. Aquesta eina ha sigut útil per analitzar el comportament dels diferents usuaris que han fet ús de l'aplicació.

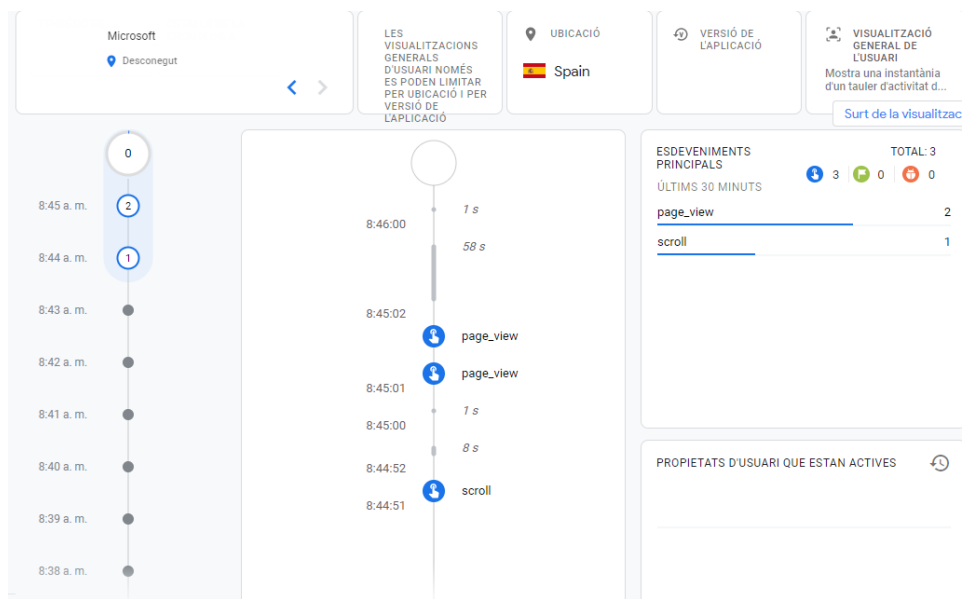


Figura 27 Un dispositiu consulta l'aplicació i es pot seguir en temps real el seu comportament

## 13. Prototips

Pel que fa el disseny de l'aplicació, s'han elaborat unes plantilles en forma de wireframes que s'han tingut en compte a l'hora d'elaborar el codi CSS. Un *wireframe* conceptualitza l'estructura de la pàgina web i actua com a guia per la distribució dels diferents elements de la pàgina.

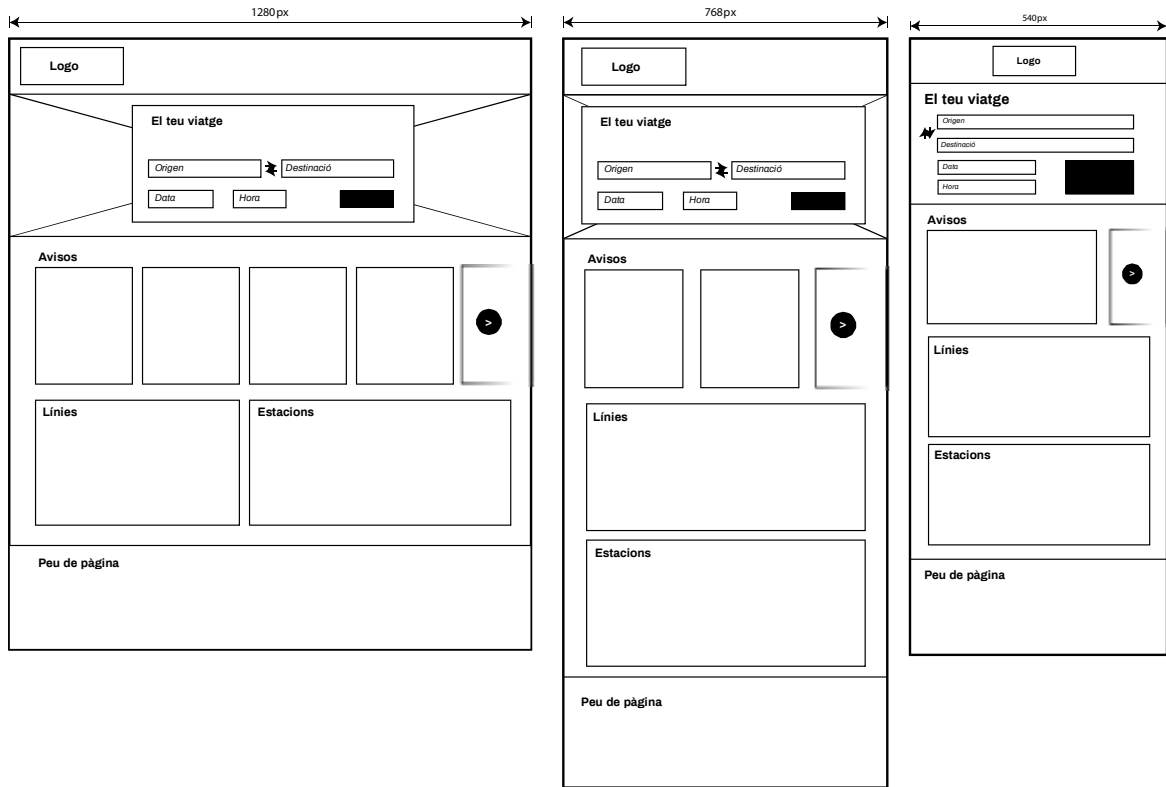
Per una banda, els wireframes de baix nivell defineixen l'estructura de les diferents pàgines que es mostren a l'aplicació en forma d'esbós. S'han desenvolupat les mostres per a pantalles d'ordinador, de tauleta i de mòbil. A continuació, es mostren els wireframes d'alt nivell on es pot apreciar una versió més semblant a la final de l'aplicatiu. És possible que alguns dels elements de la versió final no apareixin als wireframes, ja que es poden haver afegit a posteriori, a mesura que s'ha avançat en el desenvolupament de la interfície.

S'ha buscat mantenir una estructura tradicional de l'aplicació, donant prioritat a la part del motor de cerca i del mapa. A cada pàgina hi ha la capçalera, amb el menú, i el peu de pàgina. Seguidament, una franja que ocupa de banda a banda de la pantalla amb una imatge o el mapa, depenent de la pàgina on es trobi l'usuari, i a la part central la resta d'informació. S'intenta utilitzar tipografies llegibles i visibles, seguint el llibre d'estil (Veure annex), i colors per tal que la informació es pugui interpretar de forma molt visual.

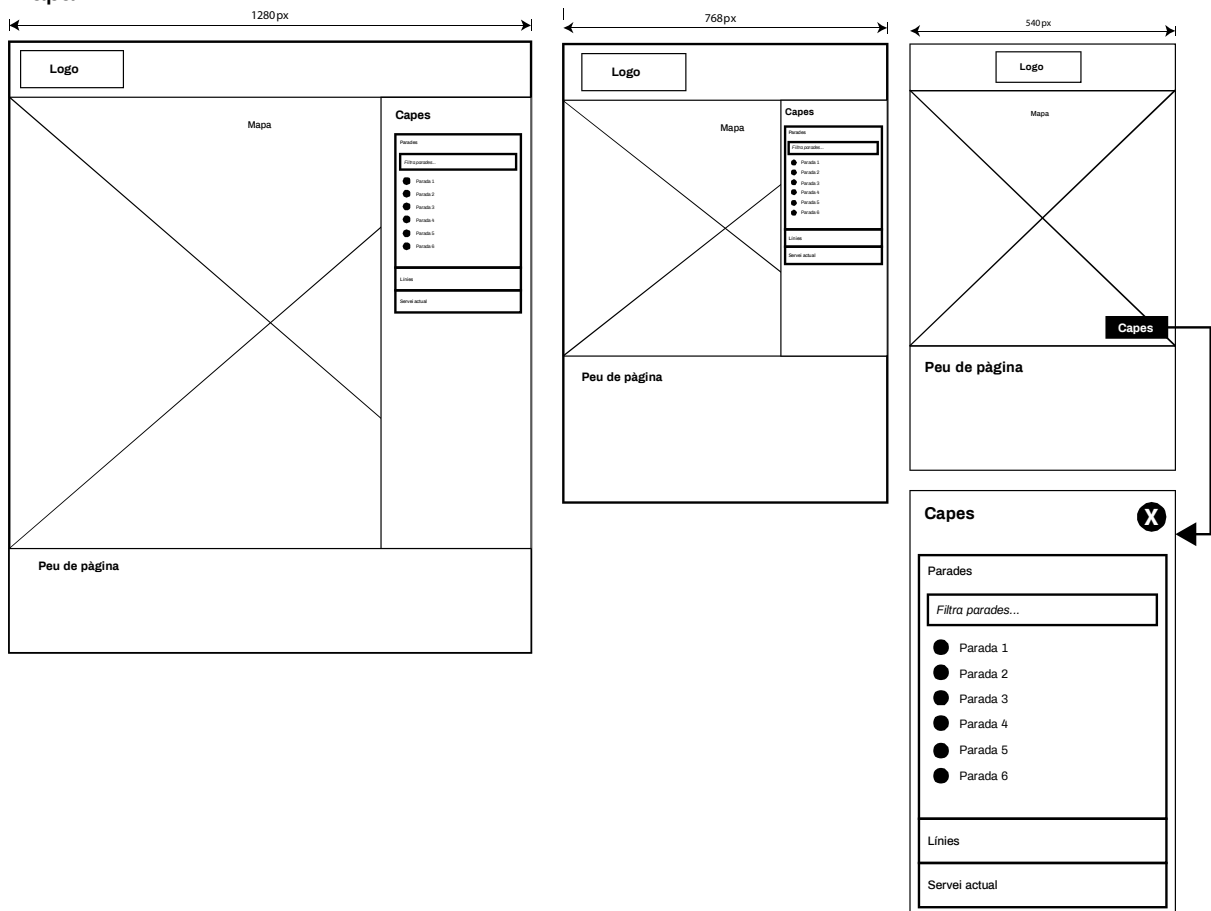
La seva retícula consta de quatre columnes, en el cas de la versió que es visualitzarà als ordinadors d'escriptori, de dues columnes per a tauletes i d'una, en cas dels dispositius mòbils.

# 13.1 Wireframes de baix nivell

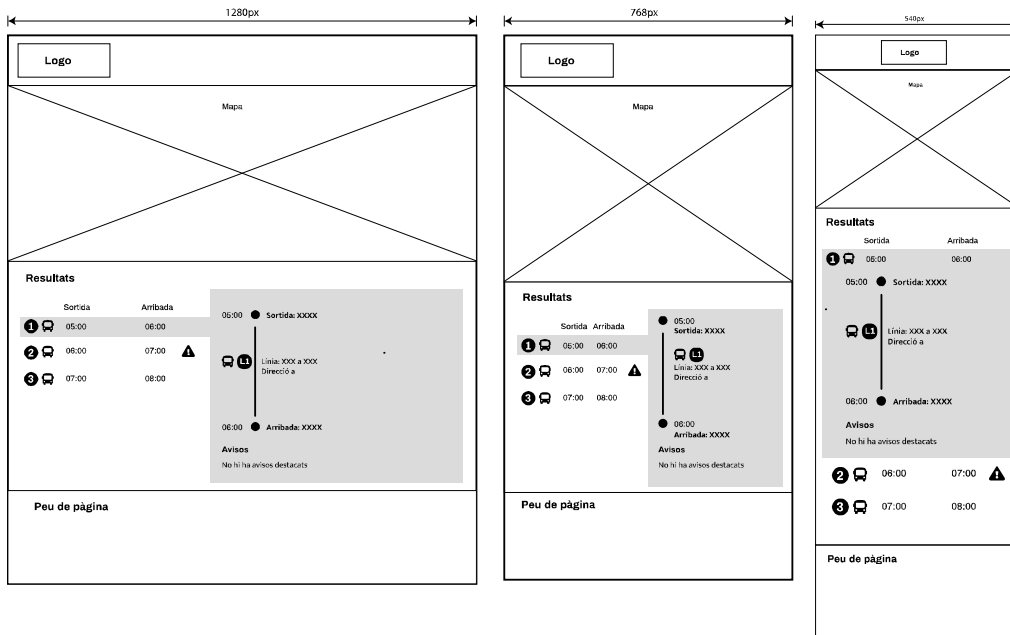
## Portada



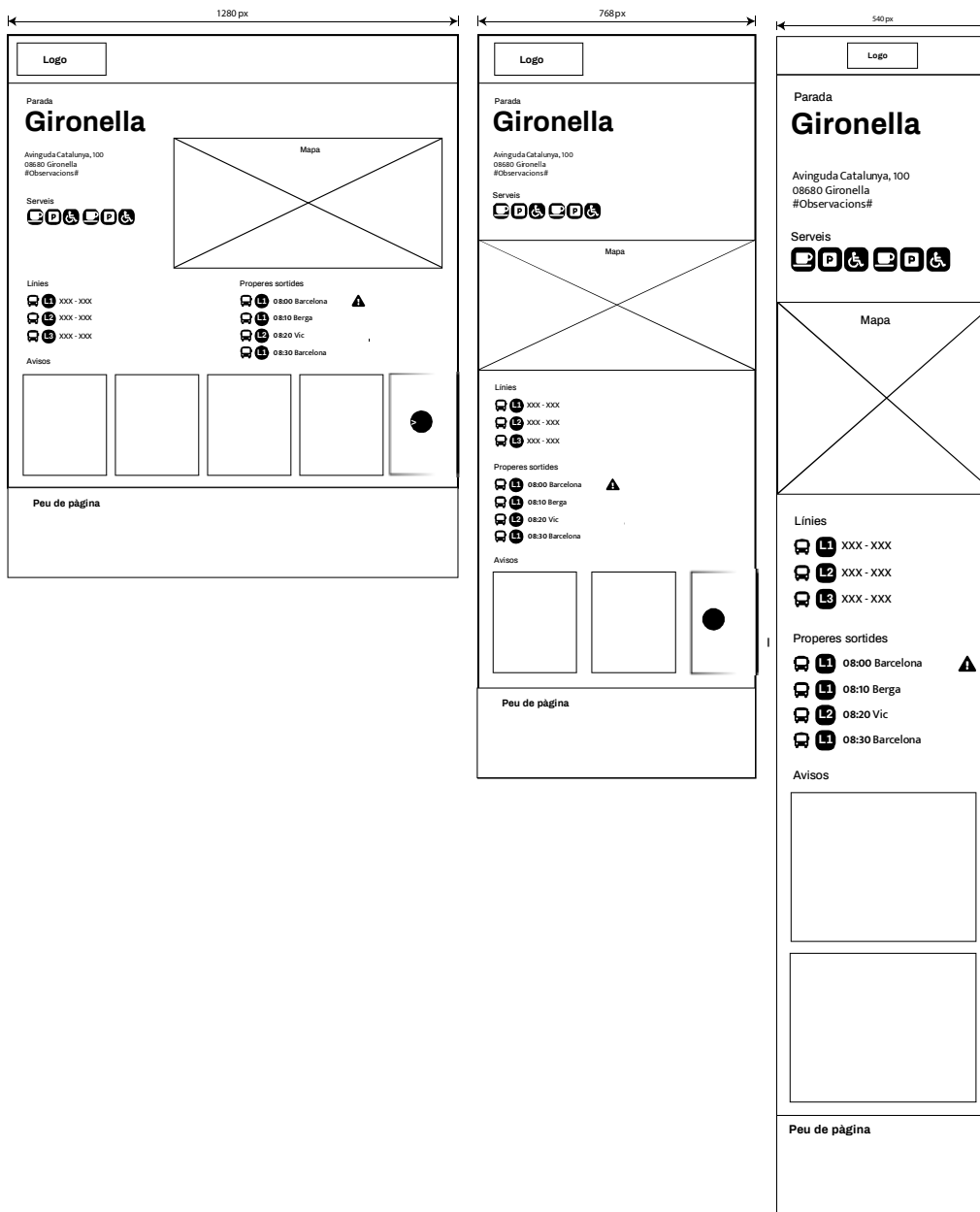
## Mapa



**Motor de cerca**



**Parada**



Linia

1280px

Logo

Barcelona - Berga

Mapa

Parada

Parada

Parada

Parada

Parada

Parada

Parada

Parada

Horari PDF

Dilluns

Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00

Peu de pàgina

768px

Logo

Barcelona - Berga

Mapa

Parada

Parada

Parada

Parada

Parada

Parada

Horari PDF

Avisos

Dilluns

Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00

Peu de pàgina

540px

Logo

Barcelona - Berga

Mapa

Parada

Parada

Parada

Parada

Parada

Parada

Parada

Parada

Horari PDF

Dilluns

Parada	08:00	08:00	08:00
Parada	08:00	08:00	08:00
Parada	08:00	08:00	08:00
Parada	08:00	08:00	08:00
Parada	08:00	08:00	08:00
Parada	08:00	08:00	08:00
Parada	08:00	08:00	08:00
Parada	08:00	08:00	08:00
Parada	08:00	08:00	08:00
Parada	08:00	08:00	08:00

Peu de pàgina

Administrador

1280px

Logo

Parades

Línies

Expedicions

Avisos

Parades

Afegir

Nom

Observacions

Mapa

Publicar

Parada 1

Parada 2

Parada 3

Parada 4

Peu de pàgina

768px

Logo

Parades

Línies

Expedicions

Avisos

Parades

Afegir

Nom

Observacions

Mapa

Parada 1

Parada 2

Parada 3

Parada 4

Peu de pàgina

540px

Logo

Parades

Línies

Expedicions

Avisos

Parades

Afegir

Nom

Observacions

Mapa

Parada 1

Parada 2

Parada 3

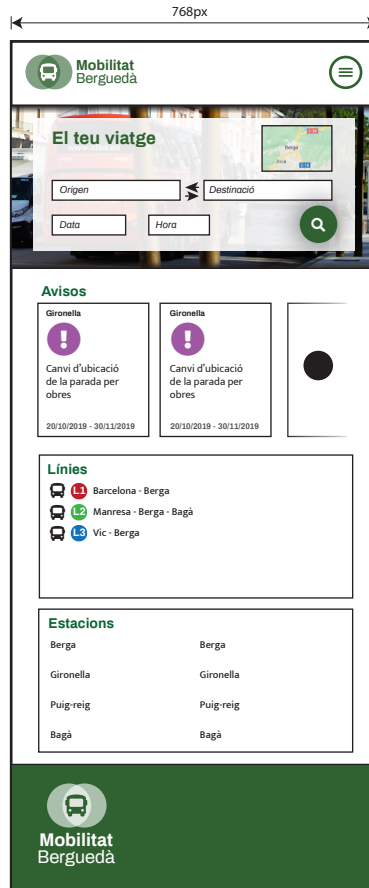
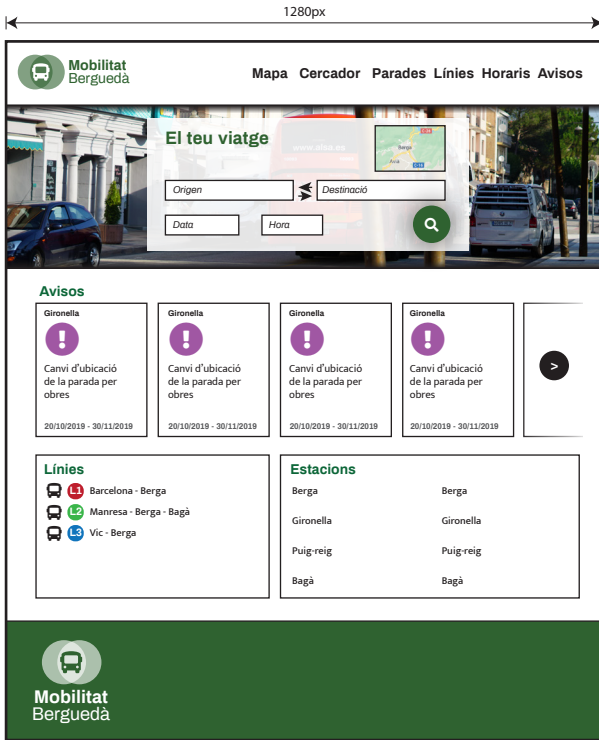
Parada 4

Peu de pàgina

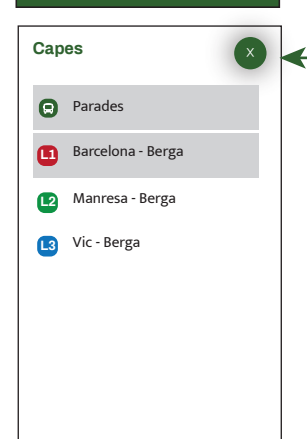
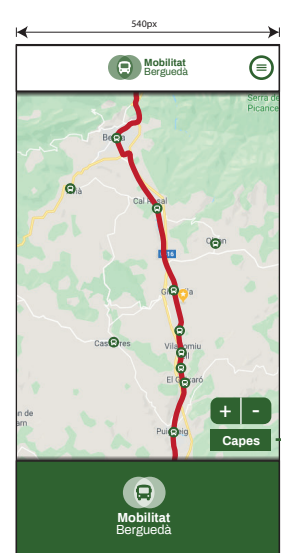
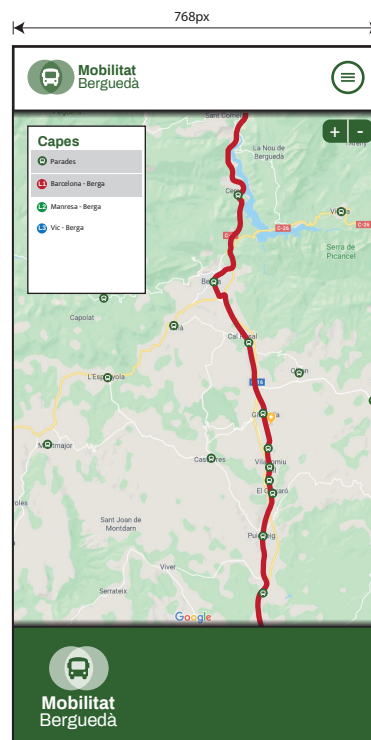
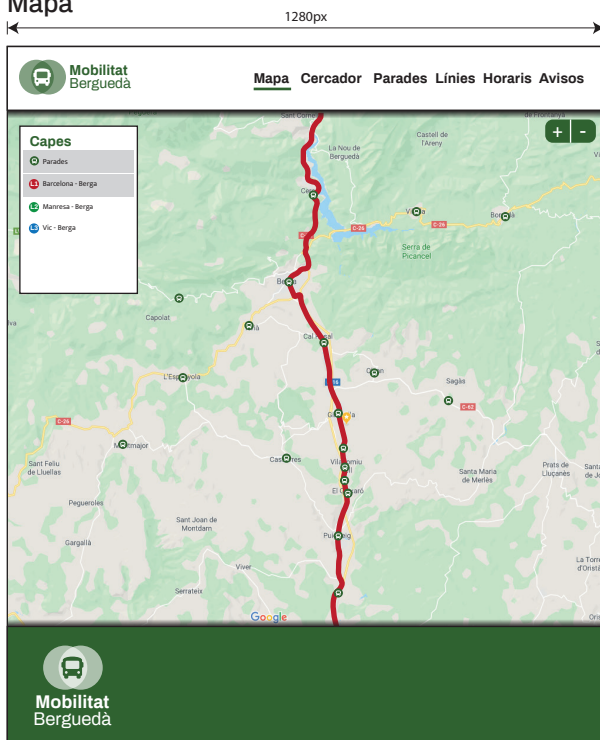


# 13.2 Wireframes d'alt nivell

## Portada



## Mapa



# Motor de cerca

1280px

**Mobilitat Berguedà** Mapa Cercador Parades Línies Horaris Avisos

**Itinerari**  
De Berga a Gironella  
el 25/10/2019 a les 05:00

**Resultats**

	Sortida	Arribada
1	05:00	06:00
2	06:00	07:00
3	07:00	08:00

05:00 Sortida: XXXX  
06:00 Arribada: XXXX  
L1 Línia: XXX a XXX Direcció a

**Avisos**  
No hi ha avisos destacats  
Afegir-ne un

**Mobilitat Berguedà**

768px

**Mobilitat Berguedà**

**Itinerari**  
De Berga a Gironella  
el 25/10/2019 a les 05:00

**Resultats**

	Sortida	Arribada
1	05:00	06:00
2	06:00	07:00
3	07:00	08:00

05:00 Sortida: XXXX  
06:00 Arribada: XXXX  
L1 Línia: XXX a XXX Direcció a

**Avisos**  
No hi ha avisos destacats  
Afegir-ne un

**Mobilitat Berguedà**

540px

**Mobilitat Berguedà**

**Itinerari**  
De Berga a Gironella  
el 25/10/2019 a les 05:00

**Resultats**

	Sortida	Arribada
1	05:00	06:00
2	06:00	07:00
3	07:00	08:00

05:00 Sortida: XXXX  
06:00 Arribada: XXXX  
L1 Línia: XXX a XXX Direcció a

**Avisos**  
No hi ha avisos destacats

**Mobilitat Berguedà**

# Parada

1280px

**Mobilitat Berguedà** Mapa Cercador Parades Línies Horaris Avisos

**Gironella**  
Avinguda Catalunya, 100  
08680 Gironella  
#Observacions#

**Serveis**  
P, D, U, W, A, B, C, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z

**Línies**  
L1 XXX - XXX  
L2 XXX - XXX  
L3 XXX - XXX

**Properes sortides**  
L1 08:00 Barcelona  
L1 08:10 Berga  
L2 08:20 Vic  
L1 08:30 Barcelona

**Avisos**  
Gironella: Canvi d'ubicació de la parada per obres (20/10/2019 - 30/11/2019)  
Gironella: Canvi d'ubicació de la parada per obres (20/10/2019 - 30/11/2019)  
Gironella: Canvi d'ubicació de la parada per obres (20/10/2019 - 30/11/2019)  
Gironella: Canvi d'ubicació de la parada per obres (20/10/2019 - 30/11/2019)

**Mobilitat Berguedà**

768px

**Mobilitat Berguedà**

**Gironella**  
Avinguda Catalunya, 100  
08680 Gironella  
#Observacions#

**Serveis**  
P, D, U, W, A, B, C, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z

**Línies**  
L1 XXX - XXX  
L2 XXX - XXX  
L3 XXX - XXX

**Properes sortides**  
L1 08:00 Barcelona  
L1 08:10 Berga  
L2 08:20 Vic  
L1 08:30 Barcelona

**Avisos**  
Gironella: Canvi d'ubicació de la parada per obres (20/10/2019 - 30/11/2019)  
Gironella: Canvi d'ubicació de la parada per obres (20/10/2019 - 30/11/2019)

**Mobilitat Berguedà**

540px

**Mobilitat Berguedà**

**Gironella**  
Avinguda Catalunya, 100  
08680 Gironella  
#Observacions#

**Serveis**  
P, D, U, W, A, B, C, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z

**Línies**  
L1 XXX - XXX  
L2 XXX - XXX  
L3 XXX - XXX

**Properes sortides**  
L1 08:00 Barcelona  
L1 08:10 Berga  
L2 08:20 Vic  
L1 08:30 Barcelona

**Avisos**  
Gironella: Canvi d'ubicació de la parada per obres (20/10/2019 - 30/11/2019)  
Gironella: Canvi d'ubicació de la parada per obres (20/10/2019 - 30/11/2019)

**Mobilitat Berguedà**

1280px

Mobilitat Berguedà

Mapa Cercador **Parades** Línies Horaris Avisos

# Barcelona - Berga

Parada

Parada

Parada

Parada

Parada

Parada

Parada

Parada

Horari PDF

Taula horària

Dilluns

Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00	08:00

Mobilitat Berguedà

768px

Mobilitat Berguedà

# Barcelona - Berga

Parada

Parada

Parada

Parada

Parada

Parada

Horari PDF

Avisos

Gironella

Canvi d'ubicació de la parada per obres

20/10/2019 - 30/11/2019

Gironella

Canvi d'ubicació de la parada per obres

20/10/2019 - 30/11/2019

Taula horària

Dilluns

Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00
Parada	08:00	08:00	08:00	08:00	08:00	08:00	08:00

Mobilitat Berguedà

768px

Mobilitat Berguedà

# Barcelona - Berga

Parada

Parada

Parada

Parada

Parada

Parada

Horari PDF

Avisos

Gironella

Canvi d'ubicació de la parada per obres

20/10/2019 - 30/11/2019

Gironella

Canvi d'ubicació de la parada per obres

20/10/2019 - 30/11/2019

Horari PDF

Taula horària

Dilluns

Parada	08:00	08:00	08:00
Parada	08:00	08:00	08:00
Parada	08:00	08:00	08:00
Parada	08:00	08:00	08:00
Parada	08:00	08:00	08:00
Parada	08:00	08:00	08:00
Parada	08:00	08:00	08:00
Parada	08:00	08:00	08:00
Parada	08:00	08:00	08:00

Mobilitat Berguedà

1280px

Mobilitat Berguedà

Parades  
Línies  
Expedicions  
Avisos

**Parades**

Afegir

Nom

Observacions

Publicar

Parada 1

Parada 2

Parada 3

Parada 4

Mobilitat Berguedà

768px

Mobilitat Berguedà

Parades  
Línies  
Expedicions  
Avisos

**Parades**

Afegir

Nom

Observacions

Publicar

Parada 1

Parada 2

Parada 3

Parada 4

Mobilitat Berguedà

540px

Mobilitat Berguedà

Parades  
Línies  
Expedicions  
Avisos

**Parades**

Afegir

Nom

Observacions

Publicar

Parada 1

Parada 2

Parada 3

Parada 4

Mobilitat Berguedà

## 14. Perfils d'usuari

Aquesta aplicació està disponible a través de la xarxa a qualsevol persona que la vulgui consultar. Preferentment va dirigida a qualsevol persona que es vulgui informar sobre les modalitats de transport que té per desplaçar-se per la comarca del Berguedà i les seves connexions amb la resta de zones de Catalunya. No obstant, podem identificar els perfils següents:

- Persona d'edat jove (entre 18 i 25 anys) que es desplaça entre el Berguedà i Barcelona, majoritàriament per qüestions universitàries, i vol conèixer els horaris que uneixen les dues zones. Té forces coneixements d'ús de noves tecnologies ja que està habituada a conviure amb dispositius electrònics, ja sigui el mòbil (smartphone) o els ordinadors. És un usuari recurrent, és a dir, que coneix la ruta que ha de fer però que vol consultar l'horari i els possibles avisos que hi hagi.
- Persona adulta habituada a fer els seus desplaçaments utilitzant vehicle privat i ara vol conèixer quines alternatives té amb transport públic. Se situaria entre els 20 i 40 anys. Es trobarien diferents nivells d'experiència pel que fa amb l'ús d'aplicacions web. Vol que l'aplicació la guiï de principi a final, és a dir abans de sortir, durant el trajecte i per acabar d'arribar a la seva destinació. A més, aquest usuari efectuarà una consulta puntual amb la nostra aplicació però que podria repetir si el resultat de la seva consulta és satisfactori.
- Persona adulta (entre 40 i 80 anys) que utilitza el transport públic habitualment però sense un patró fix, és a dir, que el seu itinerari i l'hora en que l'utilitza varia segons les seves necessitats. En aquest cas, aquest usuari se sent una mica més experimentat a l'hora d'interactuar a través d'aplicacions. Hauria de poder consultar un itinerari de parada a parada, i tenir la possibilitat de poder-lo enregistrar per així consultar-lo més endavant en una altra data que especifiqui.
- Persona adulta (entre 30 i 50 anys) que no farà ús del transport públic però que està realitzant la consulta per a una altra persona que sí que efectuarà el trajecte. La persona que utilitza l'aplicació tindrà bons coneixements d'informàtica, ans al contrari que la persona a qui li voldrem passar la consulta. Com a objectiu final, aquest usuari voldrà que el resultat de la seva cerca sigui fàcilment compartible cap a l'altra persona i que aquesta el pugui obrir sense complicacions.

## 15. Usabilitat

L'aplicació web està pensada tant per moure's tradicionalment amb l'ús d'un ratolí com amb gestos tàctils, els que es produeixen a través dels dispositius mòbils o tauletes que disposen d'una pantalla tàctil. La navegació entre les diferents parts de la pàgina es fa bàsicament a través d'enllaços.

Els enllaços que estan distribuïts per l'aplicació porten cap a les altres parts de la pàgina. Alguns d'aquests enllaços estan representats per textos i altres per botons. També hi ha el formulari del motor de cerca, on l'usuari ha d'escriure l'origen i destinació del seu recorregut i escollir la data i hora del seu viatge. En aquest darrer cas, quan l'usuari se situa sobre la casella, apareix un calendari i un selector de la hora, per tal que pugui escollir la data a través d'un calendari i no haver-la d'escriure de forma manual.

També hi ha elements d'interacció al mapa que proveeix l'API de Google Maps. En aquest cas, l'usuari el pot moure de posició clicant i arrossegant-lo, en el cas del ratolí, o amb dos dits sobre una pantalla tàctil. També pot modificar el nivell de zoom amb la rodeta mòbil del ratolí o fent el gest de 'pinça' en el cas tàctil.

La navegació per moure's entre les diferents parts principals de l'aplicació es realitza des d'un menú situat a la part superior de la pàgina que, a més, indica en quina part es troba l'usuari en tot moment. En dispositius en que la pantalla té una amplada més reduïda (terminal mòbil, per exemple), el menú queda amagat i s'hi pot accedir quan l'usuari prem el botó corresponent.

Pel que fa a l'estructura de l'aplicació web, principalment penja de la portada, d'on és possible accedir als diferents apartats de la plana. Depenent del contingut de cada pàgina, també és possible accedir als altres subapartats. Per exemple, un cop tenim el resultat d'una cerca, entre dues parades, podem accedir a la pàgina que ens mostra la informació concreta d'una parada, on també podríem accedir des de la plana que mostra tot el llistat de parades.

## 15.1 Mapa del lloc

A la figura següent es mostra la relació entre les diferents parts de l'aplicació i la seva jerarquia. També s'aprecia els diferents apartats que estan vinculats amb apartats d'altres branques.

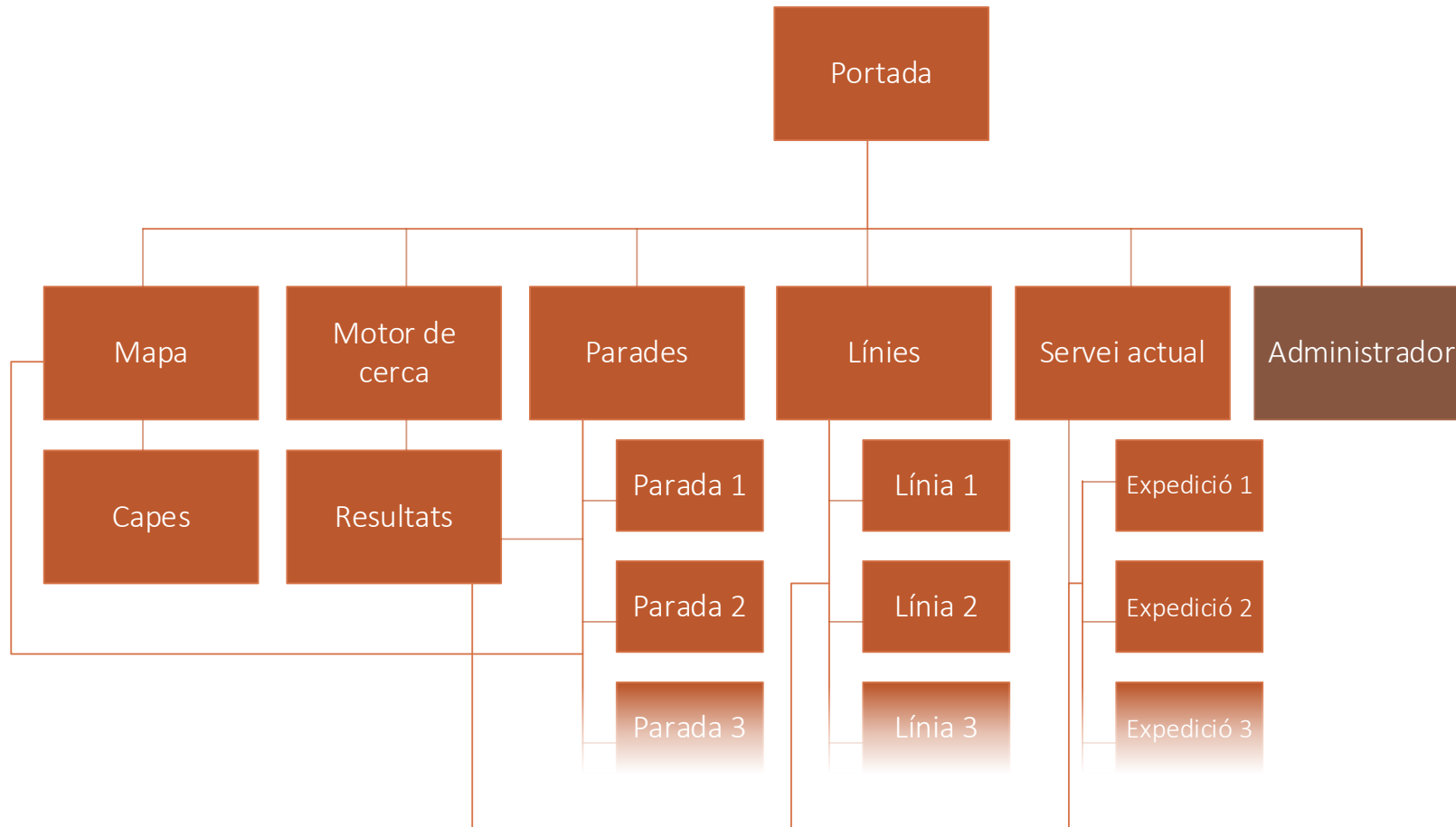


Figura 28 Mapa de l'aplicació web

# 16. Seguretat

En el desenvolupament d'aquest projecte s'han tingut en compte algunes qüestions referents a aspectes de seguretat ja que el seu resultat és un producte disponible per a qualsevol usuari amb accés a Internet i això fa que estigui exposat a possibles atacs informàtics o vulnerabilitats. Entre totes les mesures que s'han adoptat, se'n poden destacar les que es desgranen a continuació.

## 16.1 Peticions a l'API

Per fer les consultes entre el *backend* i el *frontend* es realitzen crides a l'API i, aquesta s'encarrega de fer les peticions a la base de dades MYSQL i retornar-ne la seva resposta en format JSON. L'enllaç entre les dues parts s'efectua a través de sol·licituds en una mateixa direcció. En aquest cas, si es fessin moltes peticions simultànies en aquestes adreces es podria arribar a sobrecarregar el servidor i, d'aquest fet, se'n podrien aprofitar pirates informàtics fent atacs DDoS<sup>32</sup>. Alguns mètodes ajuden a pal·liar la probabilitat que això pugui passar.

### 16.1.1 Control d'accés HTTP (CORS)

En una API es poden realitzar tota una sèrie de peticions (GET, POST, DELETE,...) que el navegador executarà si en rep la crida. En aquest projecte, el *frontend* realitza les consultes de l'API a través del mòdul *axios* que executa l'ordre `XMLHttpRequest` del navegador per tal d'obtenir les dades que mostrarà. Ara bé, els navegadors incorporen una sèrie de restriccions referents a la seguretat, i una d'elles és la **restricció de peticions entre orígens creuats**.

D'aquesta manera es restringeix l'origen des d'on es pot obtenir les dades. Si tant el *backend* com el *frontend* estan allotjats al mateix domini, `XMLHttpRequest` no tindrà cap problema en obtenir o enviar dades entre un i altre. No obstant, si es troben en dominis diferents la petició no es generarà ja que el propi navegador ho bloquejarà degut a aquesta pròpia mesura de seguretat. Així doncs, es protegeix que un script allotjat en un domini diferent al de l'API pugui fer-hi peticions, prevenint, així, atacs o peticions no desitjades des de llocs externs.

Tot i això, aquesta mesura de seguretat pot suposar un inconvenient en el cas que realment sí que volem utilitzar un *backend* i un *frontend* que es troben en dominis diferents, situació que es dona en aquest projecte. En aquest cas, s'ha de posar en marxa algun dels

<sup>32</sup> Sigles de Distributed Denial of Service, representa un atac informàtic on molts ordinadors realitzen peticions simultànies a un servidor amb l'objectiu de deixar-lo sense servei (Julián, 2012).

mecanismes que permeten posar en marxa l'**intercanvi de recursos d'origen creuat** (*Cross-origin resource sharing*, CORS<sup>33</sup>, en anglès).

En el cas de l'API, aquesta es bloqueja si intentem fer peticions com POST des d'un domini diferent. Això passa perquè XMLHttpRequest no realitza l'ordre POST directament si no que primer n'efectua una del tipus OPTIONS per verificar si està autoritzat o no per fer la crida POST que s'executarà posteriorment. Si XMLHttpRequest rep una resposta positiva (codi HTTP 204), llavors el navegador no aplica la restricció entre dominis creuats per realitzar l'ordre POST.

Ara bé, per tal que la resposta d'OPTIONS sigui positiva s'ha d'aplicar una sèrie de restriccions per poder discriminar des de quins dominis es poden realitzar peticions i des de quins no. Aquesta mesura es detalla en els headers de l'API, on hi especifiquem els dominis des d'on es poden efectuar consultes. Si la petició OPTIONS es fa des d'un domini no inclòs a la llista, retorna error i, per tant, es bloquejaria l'ordre POST posterior. Així és com funciona el mecanisme de CORS en aquest projecte.

### **16.1.2 Autenticació JWT**

Per protegir les consultes de l'API que realitzen a la base de dades accions de crear, modificar o eliminar, s'ha establert un sistema de verificació de l'usuari a través de tokens JWT (JSON Web Tokens<sup>34</sup>). Aquest token es genera si les dades passades des del component de login, l'usuari i contrasenya, corresponen a alguns dels emmagatzemats a la taula Usuaris de la base de dades.

La generació i verificació del token es fa gràcies a l'extensió de JWT per a PHP. Conté tot un seguit de mètodes que permeten la generació del token i la seva posterior verificació. A més, es permeten variar algunes de les característiques del token generat. Es destaca la possibilitat de fer-lo expirar en un moment concret. En el cas de l'aplicació del projecte, el token té una validesa d'una hora a partir del moment que es genera. Un cop ha transcorregut aquest temps, el token queda invalidat i es refusen totes les peticions que s'intenti fer amb el mateix.

Utilitzant aquest mètode també s'evita que s'hagi de transmetre l'usuari i la contrasenya cada cop que es vulgui realitzar la petició o es faci servir una clau poc segura. A més, en cas d'interceptar-se uns dels token generats, es té la certesa que aquest deixarà de funcionar al cap de poc temps, protegint l'api i la base de dades.

<sup>33</sup> [https://developer.mozilla.org/es/docs/Web/HTTP/Access\\_control\\_CORS](https://developer.mozilla.org/es/docs/Web/HTTP/Access_control_CORS)

<sup>34</sup> <https://jwt.io/>



### 16.1.3 SQL INJECTION

Un dels problemes que es podria donar afecta a les bases de dades MySQL i a la forma de fer-ne les peticions. Els pirates informàtics es poden aprofitar de la forma en que es genera la consulta per la base de dades per introduir-hi ordres que permetin obtenir dades que no han de ser accessibles o produir accions que afectin a les dades que conté.

Aquesta vulnerabilitat se sol produir quan s'utilitzen inputs (ja sigui a través de les dades que es passen a través d'un formulari (POST) o de consultes que es passen a través de la URL (GET) i aquests inputs s'insereixen directament a la consulta que s'envia a la base de dades. Agafant l'exemple que es troba al web de W3Schools<sup>35</sup>, l'atacant podria enviar una ordre que sempre fos correcta per poder obtenir les dades d'una ordre que depèn d'una condició específica.

Per evitar aquesta pràctica, es poden emprar alguns mètodes de protecció (SQL Injection Prevention, sense data). En aquest projecte, les consultes a l'API es fa utilitzant paràmetres SQL, separats de la pròpia ordre, que s'interpreten individualment, i això permet que no s'hi puguin passar altres ordres. Amb el backend PHP del projecte, aquest pas s'aconsegueix amb `bindParam` que assigna el valor que li passem al lloc que hem indicat a l'ordre. Aquest seria un exemple del codi usat:

```
public function afegir(){
    $query = "INSERT INTO AvisUsuari
            SET idExpedicio=:idExpedicio, idParada=:idParada, tipusAvis=:tipusAvis, retard=:retard";
    $stmt = $this->conn->prepare( $query );

    //Assignem els valors amagats de la petició
    $stmt->bindParam(":idExpedicio", $this->idExpedicio);
    $stmt->bindParam(":idParada", $this->idParada);
    $stmt->bindParam(":tipusAvis", $this->tipusAvis);
    $stmt->bindParam(":retard", $this->retard);

    //Executem la consulta, si va bé retorna true, si no, false.
    if($stmt->execute()){
        return true;
    }

    return false;
}
```

Figura 29 Funció afegir de la classe AvisUsuari on es veu l'ús de `bindParam` en una ordre SQL.

### 16.1.4 Neteja de la informació rebuda

Un altre element que cal tenir en compte per evitar la introducció de codi aliè és tenir controlats tots els elements input dels formularis on es pugui introduir text. Si enviem el text introduït sense tractar, podria contenir codi html que php podria llegir i executar funcions no desitjades, com per exemple cridar a altres pàgines web per executar codi maliciós (per exemple, a través d'un *iframe*).

<sup>35</sup> [https://www.w3schools.com/sql/sql\\_injection.asp](https://www.w3schools.com/sql/sql_injection.asp)

Per privar que aquestes ordres amb etiquetes HTML es puguin inserir al servidor, al projecte s'ha fet servir el mètode *htmlspecialchars* que s'encarrega de transformar tots els símbols especials (>, <, accents, etc...) en entitats HTML, així que quan es genera la pàgina no es poden interpretar.

### **16.1.5 Validació dels camps d'usuari**

Per evitar fer consultes i peticions innecessàries i per evitar que s'intentin introduir tipus de dades que no corresponen, es fan diverses validacions tant en el frontend com al backend.

En un primer moment, els camps de formulari ja estan assignats amb el tipus de dades que poden acceptar. Així, el propi navegador ja realitza una primera validació no deixant escriure valors que no corresponen als tipus especificats en l'etiqueta HTML.

Seguidament, s'analitzen els camps incorporats en els estats dels camps de React. S'ha procurat que si no estan tots els camps obligatoris omplerts no es permeti realitzar la petició cap al backend.

Finalment, el backend també disposa d'un doble control. Primerament, analitzant les variables rebudes per la pròpia lògica de l'API, i després pel sistema implementat per evitar la *SQL Injection*, que s'ha comentat anteriorment, incorpora una comprovació que si el valor passat no correspon amb l'especificat a cada valor de les entitats, tampoc realitza la consulta.

## **16.2 Còpies de seguretat**

A vegades, tot hi prendre el màxim de precaucions, es pot arribar a corrompre el sistema. Per aquest motiu, el projecte se sosté amb un sistema de còpies de seguretat. Es realitza una còpia diària dels arxius del servidor, a més d'una còpia, també diària, de tota la base SQL. La còpia dels arxius es realitza en un servidor extern que proveeix la mateixa empresa de hosting. Així doncs, es poden recuperar o clonar en un nou servidor en cas que fes falta.

## **16.3 HTTPS**

La connexió entre el servidor i el client es realitza a través del port 443, el port de connexió segura en què es basa el HTTPS (Hypertext Transfer Protocol Secure). D'aquesta forma, l'enviament de dades entre els dos es realitza de forma segura i encriptada i assegura que la pàgina que s'està visualitzant és realment la quina s'ha de mostrar i no hi ha altres ingerències externes.

Per establir la connexió a través del protocol HTTPS, s'ha de generar un certificat que enllaça el servidor i el domini. Si aquest certificat és correcte, es pot realitzar una connexió segura. Hi ha empreses que ofereixen certificats HTTPS però que tenen un cost econòmic.

Sortosament, també tenim l'opció d'obtenir certificats a través del domini públic, com els que ofereixen des de Let's Encrypt<sup>36</sup>, un projecte de l'Internet Security Research Group.

Per obtenir un d'aquests certificats, es fa servir un script que s'instal·la al servidor i que s'encarrega de comprovar la vinculació entre servidor i domini. Un cop s'ha verificat satisfactòriament, s'emmagatzema aquest certificat al mateix servidor i modifica les regles d'Apache per tal que la connexió ja es pugui realitzar amb HTTPS. Aquest certificat té una caducitat (6 mesos) i el propi script de Let's Encrypt s'encarrega de renovar-lo dies abans del seu venciment.

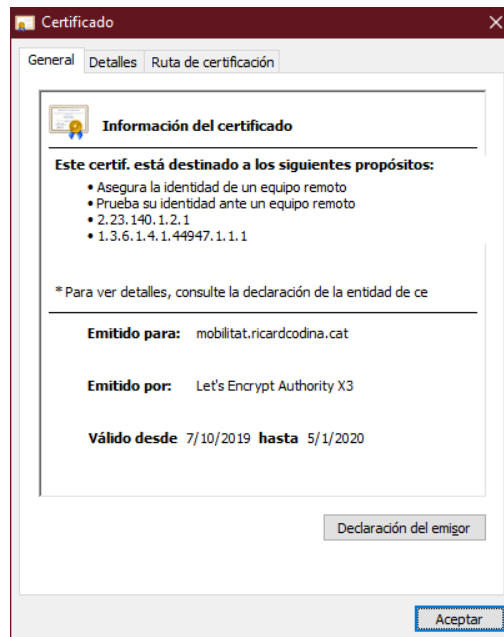


Figura 30 Informació del certificat HTTPS entre el servidor de l'aplicació i el domini

<sup>36</sup> <https://letsencrypt.org/>

# 17. Tests

Amb una versió de l'aplicació ja força avançada, es va procedir a realitzar diferents tests per comprovar-ne el seu funcionament, per veure'n la reacció de potencials usuaris i per detectar possibles errors o *bugs* no detectats anteriorment. El resultat d'aquests tests ha permès realitzar algunes millores i correccions a la versió final que es presenta en aquest treball.

## 17.1 Tests d'usuari

Per una banda, es van realitzar una sèrie de tests a diferents usuaris. Es van escollir tres perfils diferents: persona jove (25 anys) amb coneixements avançats d'informàtica i relació esporàdica amb el transport públic, persona adulta (35 anys) amb coneixements bàsics d'informàtica i usuària habitual del transport públic i persona adulta (50 anys) amb coneixements molt bàsics d'informàtica i poca relació amb el transport públic.

Als tres se'ls va demanar que intentessin assolir un conjunt d'objectius i escenaris (Jimdo 8Days, 2019) en el seu primer cop que utilitzaven l'aplicació tenint en compte les prioritats que vol resoldre aquesta aplicació. De cadascuna d'aquestes tasques es va cronometrar el temps que tardaven en realitzar-les i s'anotaven algunes observacions dels problemes o hàndicaps que s'havien trobat per aconseguir-les. Alguns usuaris van fer servir un ordinador i un navegador web per consultar l'aplicació i altres, un dispositiu mòbil amb accés a internet.

Els fulls de resposta d'aquests tests es poden trobar en els annexos d'aquesta memòria però, d'aquests, se n'extreuen les indicacions següents:

- Cal modificar el nom d'alguna de les accions perquè no queda clarament especificada. Per exemple, quan l'usuari introdueix una incidència per dir que el bus no ha arribat, l'etiqueta 'no presentat' costa d'identificar.
- Cal millorar la diferenciació de diferents elements del mapa. Per exemple, les icones entre les parades i els autobusos en circulació són molt similars i no s'acaba d'entendre el canvi de color de quan en un autobús en servei s'ha generat una incidència.

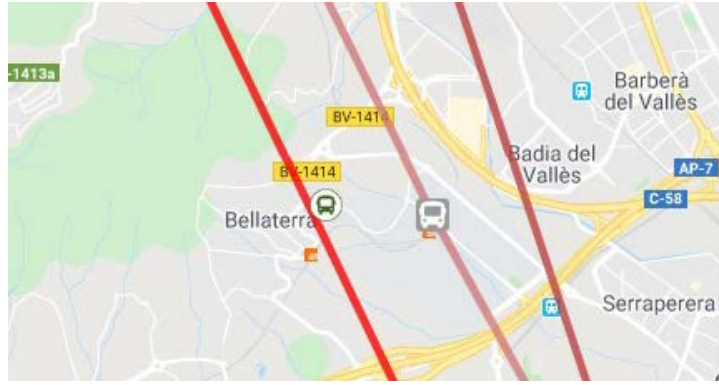


Figura 31 Versió preliminar de l'aplicació on les icones d'una parada i un bus en circulació són molt similars

- Situació d'alguns elements de la pàgina no ordenats de forma prioritària per l'usuari o poc accessibles. Per exemple, la situació dels horaris d'una parada concreta.
- Les icones per activar la geolocalització o obrir el menú de capes costen de veure. Els participants preferirien un canvi de posició o que tinguessin una etiqueta de text.
- Detecció d'algunes mancances pel que fa a l'aspecte de disseny (per exemple, no queda clara quina expedició és quina en el llistat de servei actual) i identificació d'alguns *bugs* (el filtre de parades en el menú de capes del mapa identifica els accents com a element diferent a la mateixa lletra sense accent i, per tant, si l'usuari se l'oblida, la parada ja no apareix).
- Pel que fa als temps que han tardat a realitzar les tasques, no n'hi ha hagut cap especialment alarmant pel que fa a la seva resolució. La mitjana de temps entre els tres participants és acceptable per ser el primer cop que fan servir l'aplicació.

S'ha intentat aplicar i corregir algun d'aquests fets detectats durant els tests d'usuari i s'inclouen a la versió final de l'aplicació. No obstant, i seguint la filosofia de la metodologia en cascada - *waterfall*, en anglès - (Powell-Morse, 2016), l'aplicació pot seguir en constant evolució, retroalimentant-se dels errors que es puguin detectar en les fases de proves o manteniment.

## 17.2 Tests d'usabilitat i seguretat

Per tal de realitzar els tests d'usabilitat i seguretat, s'han utilitzat les eines de **Chrome Dev Console**, que estan integrades amb el navegador de Google Chrome. Es genera un informe pel que fa als àmbits de rendiment, accessibilitat, bones pràctiques, posicionament web i de *Progressive Web App*. A més, es generen simulant un ordinador d'escriptori i un dispositiu mòbil.

A cada informe generat es rep una nota valorada sobre 100 i una sèrie de consells per tal d'optimitzar la pàgina. Aquestes serien les puntuacions rebudes sobre la versió 0.6.0 de l'aplicació web:

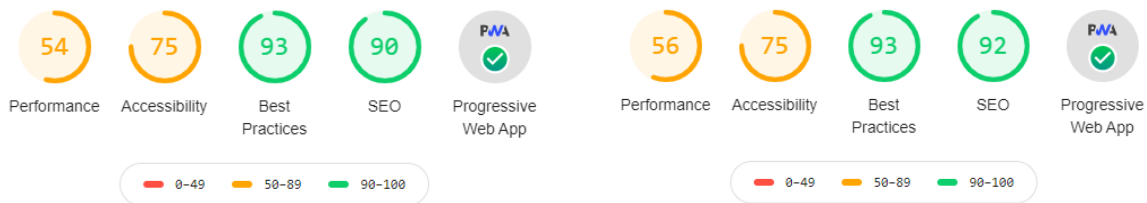


Figura 32 Resultats dels tests d'usabilitat i seguretat, per la versió d'escriptori (esquerra) i de dispositius mòbils (dreta)

Amb el resultat a la mà<sup>37</sup>, es detecta que els punts amb la puntuació més baixa són els de rendiment i accessibilitat. En canvi, els punts de SEO i bones pràctiques obtenen una nota molt alta. La validació com a *Progressive Web App* també se supera de forma satisfactòria.

Per tal de millorar els elements que han tingut una nota més pobre, l'informe ens recomana comprimir les imatges i altres recursos gràfics o utilitzar formats de nova generació que permeten estalviar temps de càrrega. També se'ns informa que es pot guanyar rendiment eliminant parts del codi que no s'utilitzen, per exemple, a la fulla d'estils CSS.

No obstant, la nota de rendiment se situa per sobre del 50 que, comparat amb altres pàgines web, és un valor habitual o sensiblement superior. Per exemple, fent la prova amb Twitter, que també fa servir el *framework* ReactJS, obté una nota de 50 en l'apartat de rendiment, i Facebook obté tan sols un 1. A la portada de la UOC, la nota de rendiment és un 20.

<sup>37</sup> L'informe complet s'incorpora als annexos d'aquesta memòria

## 18. Versions de l'aplicació

El valor de la versió de l'aplicació ens indica el seu procés de desenvolupament i millora. Per compilar una aplicació amb *npm* és necessari que aquesta contingui fins a 3 valors, seguint el format X.X.X.

Durant el desenvolupament d'aquesta aplicació, s'ha treballat amb una numeració específica referent a aquestes versions preliminars, que han seguit l'estructura numèrica de 0.X.X, on X és un número correlatiu començant pel 0. La primera versió final, la quina s'incorpora com a lliurable en aquest projecte, correspon a la versió 1.0.0. Les possibles modificacions i actualitzacions han de seguir la següent pauta, seguint les indicacions d'*npm*<sup>38</sup>:

Etapa	Pauta	Exemple de versió
Llançament	Començar amb 1.0.0	1.0.0
Pedaç	Increment del tercer dígit	1.0.1
Actualització menor	Increment del segon dígit i el tercer, torna a zero	1.1.0
Actualització major	Increment del primer dígit i el segon i tercer, tornen a zero	2.0.0

Taula 3 Procediments per anomenar les diferents versions de l'aplicació

Aquests canvis s'efectuen sobre l'arxiu *package.json* del frontend i es generen cap a l'aplicació final quan aquesta és compila (veure punt 20.3).

Ara bé, com que aquesta aplicació web compleix els requisits de Progressive Web App i es pot instal·lar als dispositius de l'usuari s'ha hagut d'utilitzar un recurs que permeti l'actualització remota de la mateixa i impedeixi l'emmagatzematge d'alguns elements a la memòria cau quan s'ha realitzat una actualització.

Finalment, s'ha optat per l'ús de **Cache Busting** per a React<sup>39</sup> que consta d'un component específic que compara la versió que té emmagatzemada l'usuari amb la versió que està disponible en línia. Si aquesta comparació determina que són diferents, força la recàrrega sencera de l'aplicació i així ens assegurem que el client visualitza sempre l'última versió de l'aplicació. Si no, ens podem trobar que, sense una actualització manual de l'aplicació per part de l'usuari, aquest continuï visualitzant versions antigues de l'aplicació perquè s'han emmagatzemat a la seva memòria cau i no s'han actualitzat.

<sup>38</sup> <https://docs.npmjs.com/about-semantic-versioning>

<sup>39</sup> <https://github.com/flexdinesh/cache-busting-example>

# 19. Requisits d'instal·lació

En el cas de voler instal·lar l'aplicació en un nou servidor per tal que sigui accessible als dispositius que la vulguin utilitzar, s'han de tenir en compte una sèrie de requisits.

## 19.1 Requisits del servidor

Tots els fitxers han d'estar emmagatzemats en un servidor o ordinador amb capacitats de servidor. Aquest servidor pot estar basat en sistemes operatius Windows o UNIX (Linux). A continuació, es detalla la configuració del servidor Linux, ja que és la que ha estat utilitzant en aquest projecte. En el cas de Windows només caldria replicar la mateixa configuració.

Per executar tota l'aplicació, caldrà realitzar una configuració LAMP que, a més del sistema operatiu Linux, inclogui a Apache, per dirigir les peticions HTTP, una base de dades MySQL, on s'emmagatzemarà la informació i PHP, que s'encarrega d'executar el codi del cantó servidor. Les versions mínimes han de ser, Apache v 2.4.25, MySQL v 5.0.12 i PHP v 7.0.33.

Per executar el backend, la configuració anterior ja és suficient i s'executarà sense problemes. Pel frontend, cal tenir instal·lat els paquets de *nodejs* i *npm*, si es vol utilitzar l'aplicació en desenvolupament. La versió que s'ha fet servir és la 10.16.3. També es recomana instal·lar *phpmyadmin* per poder gestionar la base de dades, MySQL, des d'una interfície d'usuari.

## 19.2 Requisits del client

Un cop en funcionament, per accedir a l'aplicació, caldrà tenir un navegador web compatible. Per defecte, l'aplicació es compilarà per tenir compatibilitat amb tots els navegadors dels que disposin més d'un 0,2% de quota de mercat (Create React App Docs, 2019). Dins d'aquesta llista<sup>40</sup>, hi trobem les darreres versions de Google Chrome, Mozilla Firefox, Microsoft Edge i Apple Safari. No obstant, hi pot haver altres navegadors que també siguin compatibles amb l'aplicació ja que reuneixen els requisits i estàndards necessaris per poder-la executar.

També s'ha de tenir en compte que l'execució de codi Javascript no ha d'estar desactivada (en cas contrari, la pàgina mostrarà un missatge d'error) i cal tenir també activar l'emmagatzematge de *cookies*.

<sup>40</sup> [https://browserl.ist/?q=%3E+0.2%25%2C+not+dead%2C+not+op\\_mini+all](https://browserl.ist/?q=%3E+0.2%25%2C+not+dead%2C+not+op_mini+all)



## 20. Instruccions d'instal·lació

En el moment de fer el projecte, tot el programari es troba instal·lat i en funcionament en un servidor extern i accessible des d'internet a través d'un domini. No obstant, es poden instal·lar i executar des d'un altre aparell, utilitzant els fitxers que s'entreguen com a adjunts a la memòria i configurant correctament el servidor. Primerament, s'han de complir els requisits que s'han esmentat en el punt anterior.

### 20.1 Instal·lació de la base de dades

Els arxius lliurats inclouen, dins de la carpeta BD, un arxiu .sql que conté tota l'estructura de la base de dades amb la informació actualitzada fins al dia d'entrega d'aquesta memòria. Considerant que es disposa d'una base de dades MySQL i amb accés a través de phpmyadmin, s'ha de procedir amb passos següents:

1. Accedir a l'administració de la base de dades i fer clic al botó Importa.
2. Seleccionar el fitxer mobilitatbergueda.sql que es localitza a l'ordinador o en una carpeta del servidor.
3. Fer clic a Executa i esperar que es realitzi la configuració d'importació.
4. El nom de la base de dades creada és mobilitatbergueda. Accedir dins d'ella i fer clic a la pestanya Permisos. Allà, clicar sobre Afegir compte d'usuari.
5. Omplir els valors següents:
  - a. Nom d'usuari: mobilitatAPI
  - b. Nom de servidor: localhost
  - c. Contrasenya: bergueda2019
6. Deixar la selecció de permisos com està per defecte: s'assignen tots els permisos a la base de dades 'mobilitatbergueda' però ja no se'n tenen per les altres.

Amb aquests passos, la configuració de la base de dades quedaria completada.

### 20.2 Instal·lació i execució del *backend*

Per realitzar la instal·lació de l'API REST caldrà copiar els arxius de la carpeta API en una carpeta pública del servidor, per tal de poder realitzar la connexió entre *backend* i *frontend*. L'api hauria de ser accessible des del port 80 del servidor (HTTP), a la ruta localhost/api, si s'ha copiat la carpeta directament a l'arrel dels documents HTML del servidor.

Per comprovar, que l'API funciona en perfectes condicions, es pot emprar el programari **Postman** fent peticions GET a algunes de les API REST esmentades a l'apartat corresponent i esperant que aquestes es resolguin de forma satisfactòria.

### 20.3 Instal·lació i execució del *frontend*

De la mateixa manera que en el cas del *backend*, cal copiar la carpeta d'arxius del frontend a l'arrel dels arxius html públics. Per executar la **versió de desenvolupador**, caldrà invocar a nodejs que executarà l'aplicació en un port del nostre servidor. Per fer això, s'hauran de seguir els passos següents:

1. Obrir la consola del servidor i accedir amb un compte d'usuari amb permisos d'execució.
2. Navegar fins a la ruta on tenim emmagatzemada l'aplicació.
3. Utilitzar la comanda *npm start*, per inicialitzar l'execució de l'aplicació.
4. Comprovar que l'aplicació s'executa correctament a través de la direcció i port que ens mostra la consola.

S'ha de tenir present que, quan es tanca la finestra de la consola es para l'execució de l'aplicació. Si volem mantenir-la encesa en segon pla, podem utilitzar paquets com *pm2*. Pm2 manté en execució una aplicació a través d'un port concret fins que no s'envia l'ordre d'*stop*. També caldrà configurar-lo de forma que, si es reinicia el servidor, també ho faci l'execució de l'aplicació un cop el servidor es torni a obrir.

Un altre factor a tenir en compte per executar la versió de desenvolupador és que cal substituir totes les referències al domini utilitzat per les crides a l'API (veure punt 7, sobre l'arquitectura del projecte) durant la producció del projecte i substituir-lo pel domini que l'usuari vol utilitzar i que hagi especificat a Apache. Això és pot fer utilitzant l'acció de buscar i substituir que inclouen molts editors de codi, com Visual Studio Code (veure punt 8.1.3)

Si es vol executar la **versió de producció**, només cal guardar la carpeta *build* a l'arrel dels arxius html públics i configurar Apache perquè apunti cap a aquesta carpeta. Cal tenir en compte, però, que l'adreça de l'API està implementada en el propi codi font del *frontend* i que, potser, caldria canviar-la, si fos necessari, per tal que apunti a la nova destinació. Un cop modificada de tots els fitxers, es pot executar *npm run build* per crear una nova versió final de producció i poder-la executar directament des de la carpeta *build*.

## 21. Instruccions d'ús

Es pot utilitzar per portar a terme diverses tasques relacionades amb la mobilitat, se'n detallen les instruccions de les més importants.

### 21.1 Cerca d'un itinerari

El resultat de seguir aquests passos és trobar un itinerari amb transport públic, de punt a punt.

1. A la pàgina inicial es troba un mòdul amb el cercador o es pot entrar a la pàgina dedicada al mateix.
2. S'omplen les caselles d'origen i destinació. Es pot escollir una de les parades que apareix al desplegable o escriure una adreça o nom d'un punt d'interès.
3. Indicar la data i hora de sortida en que es vol realitzar el trajecte.
4. Un cop es fa clic al botó de Buscar, l'aplicació comença a calcular el millor itinerari. Oferirà les tres millors opcions per realitzar el trajecte mostrant, en primer lloc, la que més s'adequa a les opcions que ha escollit l'usuari.
5. Es pot prémer una opció diferent i la pàgina generarà els detalls de l'opció seleccionada.

### 21.2 Consulta de la informació d'una parada

Si es vol conèixer els detalls d'una parada concreta, hi ha diferents maneres d'aconseguir-ho:

- a) A través de la pàgina Parades, on apareix un llistat amb totes les parades i en seleccionar la que es vulgui consultar, s'accedeix a la pàgina amb el detall de la parada.
- b) Des de l'apartat del mapa, es pot seleccionar la parada o bé fent-li clic al mapa o escollint-la del llistat, i apareixerà més informació sobre la parada.
- c) Des de qualsevol apartat on aparegui el nom de la parada (des del recorregut d'una línia, la consulta d'un itinerari...), fent-hi clic s'enllaça amb la pàgina de detall sobre la parada.

### 21.3 Consulta dels horaris d'una línia

Per conèixer la taula horària de les expedicions que circulen per una línia concreta, es poden seguir els passos següents:

1. Accedir a l'apartat Línies i prémer la línia que es vulgui consultar.
2. Fer *scroll* fins a l'apartat corresponent de Taules Horàries.
3. Seleccionar la direcció de la línia en que es vulgui realitzar la consulta i el dia de la setmana.
4. La taula mostrarà les expedicions segons les dades indicades. Si es fa clic al símbol de l'autobús que encapçala cadascuna d'elles es veurà la pàgina de detall de l'expedició concreta.

## 21.4 Indicar un avís o incidència

Si s'està davant d'una incidència i es vol informar a la resta d'usuaris de la mateixa, es pot indicar utilitzant l'aplicació.

1. S'accedeix a la pàgina de la línia i es busca l'expedició concreta a la taula horària.
2. S'accedeix a la pàgina de la línia i es busca l'expedició concreta a la taula horària i es clica a la icona de l'autobús per accedir a la seva pàgina, o a través de la pàgina d'una parada concreta, es busca l'expedició concreta a la que es vulgui accedir.
3. S'entra a la pàgina de l'expedició fent clic al símbol de l'autobús que hi ha just a sobre de l'expedició que volem seleccionar.
4. A l'apartat d'avisos, es prem el botó Informar d'una incidència.
5. A la finestra emergent que apareix, s'escull un tipus d'incidència entre les opcions que s'ofereixen. Cal omplir l'estació o parada on s'ha observat aquesta incidència. A més, si és del tipus 'retard' es pot especificar el temps de demora que arrossega l'autobús.
6. Si l'enviament s'efectua correctament, mostra un missatge de confirmació i la incidència apareix al llistat d'avisos, precedida de l'hora en que s'ha enviat. Les incidències a l'usuari es mostren només durant el dia natural que s'han enviat.

## 21.5 Accedir a l'administrador

Si es vol accedir al panell d'administrador s'ha de fer utilitzant la ruta /admin, del servidor on s'estigui executant l'aplicació. Hem d'introduir un email i contrasenya d'un usuari que estigui registrat al sistema. En cas contrari, hem de crear l'usuari utilitzant una petició POST a l'API `usuaris/afegirUsuari.php`, amb el nom, usuari i contrasenya, passats en format JSON.

## 22. Bugs

S'han identificat una sèrie de *bugs* i errors pendents de resoldre en posteriors versions de l'aplicació. De major a menor gravetat, es poden enumerar els següents:

- Alguns cops, la parada més pròxima que s'agafa a partir d'una adreça no és, realment, la més propera. S'haurà de modificar el codi del *backend* per tal que es compari el temps de recorregut fins a les parades més properes i esculli la correcta.
- A vegades s'obren finestres dins d'una altra finestra. Això ocorre, per exemple, si obrim la informació d'una parada des del mapa i després fem clic a una línia (que també s'obre dins d'una altra finestra). S'hauria de planejar gestionar l'estat de les finestres des d'un estat de React superior i saber quan ha d'obrir una finestra i quan no.
- En certes ocasions, ReactJS perd l'estat d'alguns dels seus components i genera una pàgina en blanc. Això pot ser degut a un mal enllaç entre components o en arribar a un límit de memòria que fa que els diversos elements deixin de transmetre el seu estat. Caldria revisar totes les connexions entre components i mirar d'optimitzar els recursos que s'engeguen quan es carrega un cert component.
- Intentar reduir la dependència de Google Maps, ja que cada petició consumeix crèdit i si s'excedeix de la quota gratuïta, l'ús d'aquest servei s'ha de pagar. Per evitar això, es poden realitzar accions com la de restringir l'ús del mapa a petició de l'usuari o reduir les peticions d'itinerari a peu, per exemple, quan l'usuari escull directament la posició d'una parada concreta.

## 23. Projectió a futur

Com tota aplicació, el producte obtingut no és final i pot afegir noves millores i característiques en un futur. Algunes d'elles no s'han afegit per manca de temps o per complicacions durant el seu desenvolupament. D'altres, perquè no s'han marcat com a prioritàries en aquesta primera versió de l'aplicació. Aquest n'és el llistat de propostes:

- **Creació d'una àrea personal per a usuaris de l'aplicació.** Possibilitat que un usuari es registri amb les seves pròpies dades i disposi d'opcions per crear itineraris recurrents i parades preferides per poder-les consultar més ràpidament quan ho necessiti. També es pot incloure l'opció d'emmagatzemar la direcció de la casa i del lloc de feina de l'usuari per poder realitzar cerques més ràpides a partir d'aquests punts ja establerts.
- **Subscripció d'alertes.** Possibilitat per a l'usuari de rebre notificacions al seu navegador o dispositiu mòbil cada cop que es produeixi una alerta en alguna línia, parada o expedició que triï. Pot anar combinat amb l'àrea personal, des d'on podria activar i desactivar aquestes alertes.
- **Petició en línia dels busos del transport a la demanda.** En aquests moments, la petició es realitza a través d'un número de telèfon del Consell Comarcal del Berguedà o per Whatsapp. L'aplicació podria integrar un gestor amb les peticions d'aquest servei realitzades, donar la possibilitat a l'usuari de fer les peticions a través de la pròpia aplicació, així que enviar alertes i missatges als usuaris que hagin fet la sol·licitud d'un servei de transport a la demanda, per avisar-los de canvis al servei o de recordatori de la seva petició.
- **Integració amb altres plataformes de transport públic.** La primera versió de l'aplicació només permet realitzar trajectes de línies que transcorren per la comarca. Es podria buscar la forma de combinar altres mètodes de transport fora de la comarca per acabar d'arribar al punt escollit com a destinació. Es podrien fer servir les dades de transport públic de Google Maps o el servei 'Mou-te' de la Generalitat.
- **Ampliació de característiques.** Afegir noves funcionalitats com l'enriquiment del format dels avisos (permetre modificar l'estil del text, afegir arxius adjunts...), augmentar la informació cap a l'usuari (introducció de els preus del trajecte, informació addicional...) i afegir altres funcions que han quedat pendents d'aplicació (serveis a les parades, o recorregut exacte sobre el mapa).

## 24. Anàlisi de mercat

Per preparar aquesta aplicació web, he analitzat diversos llocs webs de diferents operadors de transport d'arreu del món per veure'n els recursos que ofereixen i com es presenten de cara a l'usuari. A mode de resum, he agafat els següents:

### 24.1 RATP

La RATP és l'empresa que s'encarrega d'una gran part de les línies de metro, tramvia i autobús de la ciutat de París, França. Fa poc temps van renovar la seva pàgina web<sup>41</sup> per adaptar-la a les tecnologies web actuals i fer-la multidispositiu. De fet, està desenvolupada seguint la filosofia de 'Mobile first' (RATP, 2017), ja que moltes de les visites ja les reben des de dispositius mòbils. El que sorprèn d'aquesta pàgina és que donen molta importància al cercador d'itineraris i ho veiem ja que a la portada, aquest mòdul apareix fins i tot per sobre de la pròpia capçalera on hi ha el menú general del web.

El buscador de la web reconeix parades, adreces i permet obtenir la ubicació de l'usuari a través de la geolocalització. Un cop generem la cerca, el portal ens ofereix diferents opcions per fer la ruta i la mostra al costat d'un mapa. El menú de la pàgina mostra quatre opcions a les que dona més importància (itineraris, horaris, mapes i visita) i la resta d'apartats estan amagats en un submenú.

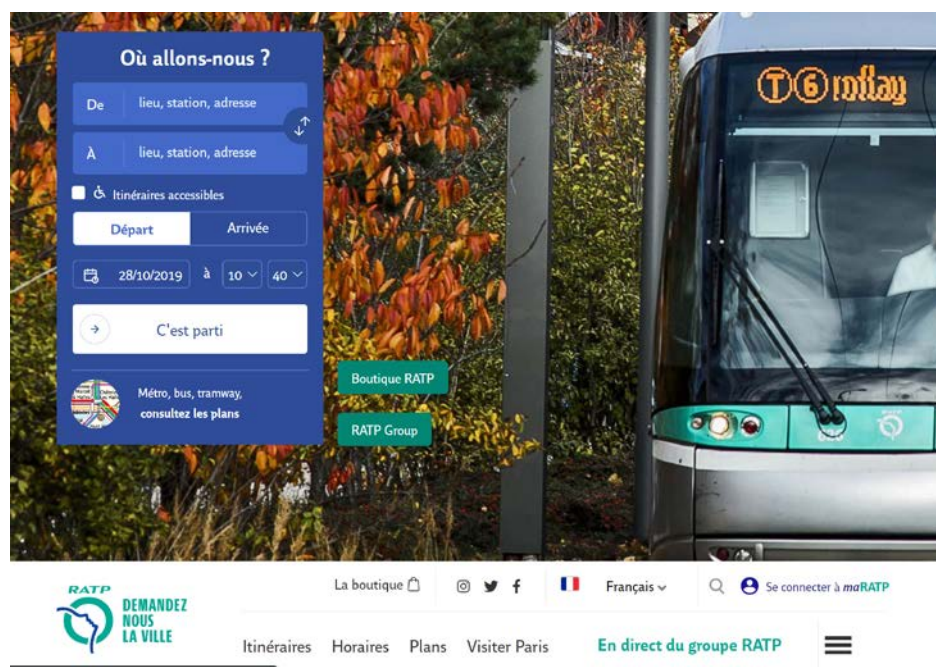


Figura 33 Portada del lloc web de la RATP on el buscador d'itineraris se situa per sobre de la capçalera de la pàgina

<sup>41</sup> <https://www.ratp.fr>

## 24.2 Transport for London

Transport for London (TfL) és d'institució governamental que s'encarrega de gestionar una bona part de la xarxa de transport públic de la ciutat de Londres, al Regne Unit. Aquesta ciutat sol estar a l'avantguarda en l'aplicació de noves tecnologies en diferents àmbits i per això, s'ha decidit analitzar la pàgina web d'aquesta entitat.

Seguint l'exemple de la RATP, el web de TfL<sup>42</sup> també situa el cercador en el primer nivell de consulta un cop s'entra a la seva portada. Les opcions són pràcticament les mateixes, però deixen a l'usuari registrar-se al lloc web i emmagatzemar els seus itineraris preferits per poder-los consultar quan ho necessiti.

Ja que la xarxa del metro de Londres sol patir forces incidències i avaries<sup>43</sup>, s'inclou un apartat on s'informa de l'estat de totes les línies. S'informa de la situació per línies però també mostra un missatge si el servei és correcte i no hi ha incidències destacades.

Una part de la pàgina que sobresurt respecte a les altres que s'han analitzat és la que permet les consultes individuals a les estacions. S'hi informa de la sortida dels trens en temps real, dels serveis que podem trobar a l'estació, la seva situació, mapes específics de l'estació per a persones amb mobilitat reduïda i, fins i tot, un gràfic amb les hores amb més afluència de passatge, similar al quin es pot veure en una fitxa d'un establiment a Google Maps<sup>44</sup>.

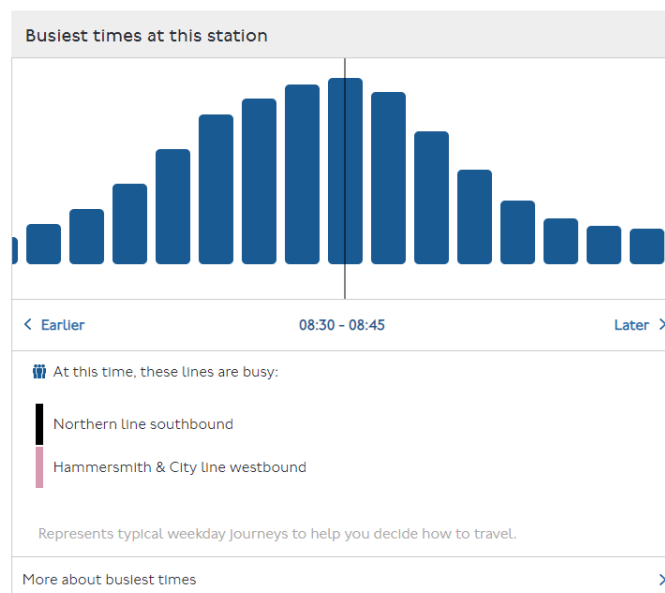


Figura 34 Exemple del gràfic d'afluència de passatge a l'estació de Moorgate

<sup>42</sup> <https://tfl.gov.uk/>

<sup>43</sup> De mitjana, es produeixen fins a 5 fallades de senyalització cada dia a la xarxa del metro de Londres (Grafton-Green, 2018).

<sup>44</sup> <https://support.google.com/business/answer/6263531?hl=en>



## 24.3 Mapa Swiss Railways Network

Aquesta plana esta feta per un particular i no representa cap organisme ni institució però s'agafa com a mostra de mapa interactiu de línies de transport públic de la Confederació Helvètica. És un projecte desenvolupat des de l'any 2007 per Vasile Coțovanu i es pot consultar de forma pública a través d'Internet<sup>45</sup>.

Veiem sobre un plànol de Google Maps les diferents línies de tren de Suïssa i, sobre ell, els diferents trens situats en base a l'horari teòric. Es produeix el moviment dels trens a mesura que passen els minuts, l'usuari pot especificar una hora concreta per veure'n la situació en aquell instant, seleccionar un tren i consultar-ne el seu recorregut i filtrar els diferents tipus de tren.

No obstant, apareixen alguns problemes en forma de *bug*, ja que si reduïm força el nivell de zoom (per exemple, fins a enquadrar el país alpí sencer), el mapa ha de renderitzar moltes icones a la vegada i aquestes comencen a pampalluguejar produint un efecte molt molest. S'hauria d'analitzar si és un error de programació o una possible limitació de l'API de Google Maps.

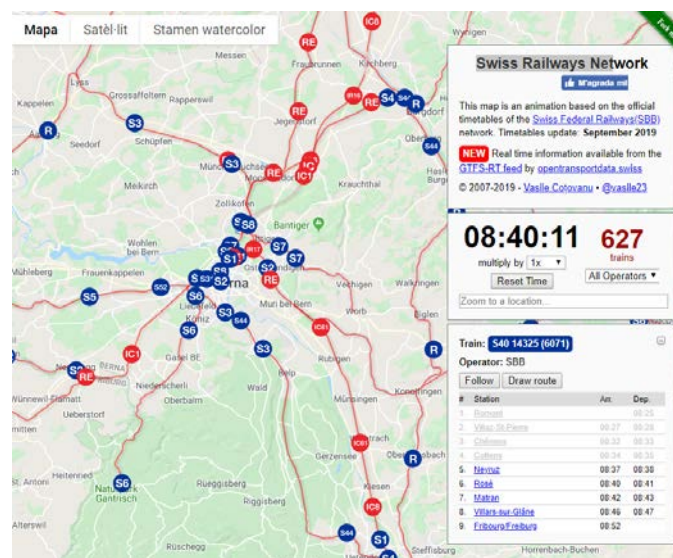


Figura 35 Vista del mapa a la zona de Berna

## 24.4 TMB Maps

Transports Metropolitans de Barcelona (TMB) a través de la seva pàgina web ofereix el servei 'Vull Anar?'<sup>46</sup> que permet planificar itineraris per la xarxa de transport públic de Barcelona. El seu funcionament és similar al quins s'han analitzat als apartats anteriors i també disposa d'un mapa per mostrar les diferents línies de metro i autobús.

<sup>45</sup> <http://maps.vasile.ch/transit-sbb/>

<sup>46</sup> <https://www.tmb.cat/ca/barcelona/vull-anar>

A través d'un menú desplegable es mostren totes les línies i es poden activar o desactivar sobre el plànol a demanda de l'usuari, així pot crear el mapa amb les línies que li interessin. A més, la informació s'adapta segons el nivell de zoom que tingui el mapa. Per exemple, si ajustem al mapa a una illa de carrers veiem impreses sobre el mapa les parades de bus, sortides de metro, etc. que no apareixen quan tenim una visió més general de la ciutat.



Figura 36 Menú desplegable del mapa de TMB amb només algunes línies seleccionades

## 24.5 Assistant SNCF

La companyia de ferrocarrils francesa, la SNCF, ofereix al client una aplicació per al seu dispositiu mòbil que permet consultar els horaris en temps real, itineraris i la situació de les estacions. Una de les característiques que també destaca d'aquesta aplicació mòbil<sup>47</sup>, és que permet als usuaris enviar alertes per avisar a la resta d'usuaris que volen utilitzar el mateix servei.

Els avisos que es poden enviar estan repartits en tres categories: aflluència de viatgers, retard d'un tren o absència del tren en una estació concreta. Quan l'usuari envia l'alerta, apareix a la fitxa de consulta del tren sota la notació que aquelles dades han estat enviades per un altre passatger, diferenciant-les de les que envia el propi operador. D'aquesta forma, es tracta d'una eina col·laborativa d'informació sense perdre de vista els avisos oficials.

Com a inconvenient d'aquest sistema és que, a primera vista, no hi ha cap filtre de la informació que es rep. Tot i que l'usuari no pot introduir text, només pot escollir entre les categories esmentades anteriorment, podria passar que qui introdueix l'avís s'equivoca de

<sup>47</sup> Es pot descarregar gratuïtament des de les botigues d'aplicacions Apple Store i Google Play.

tren i apareix referenciat de forma incorrecta, o algú envia informació falsa de manera malintencionada i això pot confondre a la resta d'usuaris de l'aplicació.

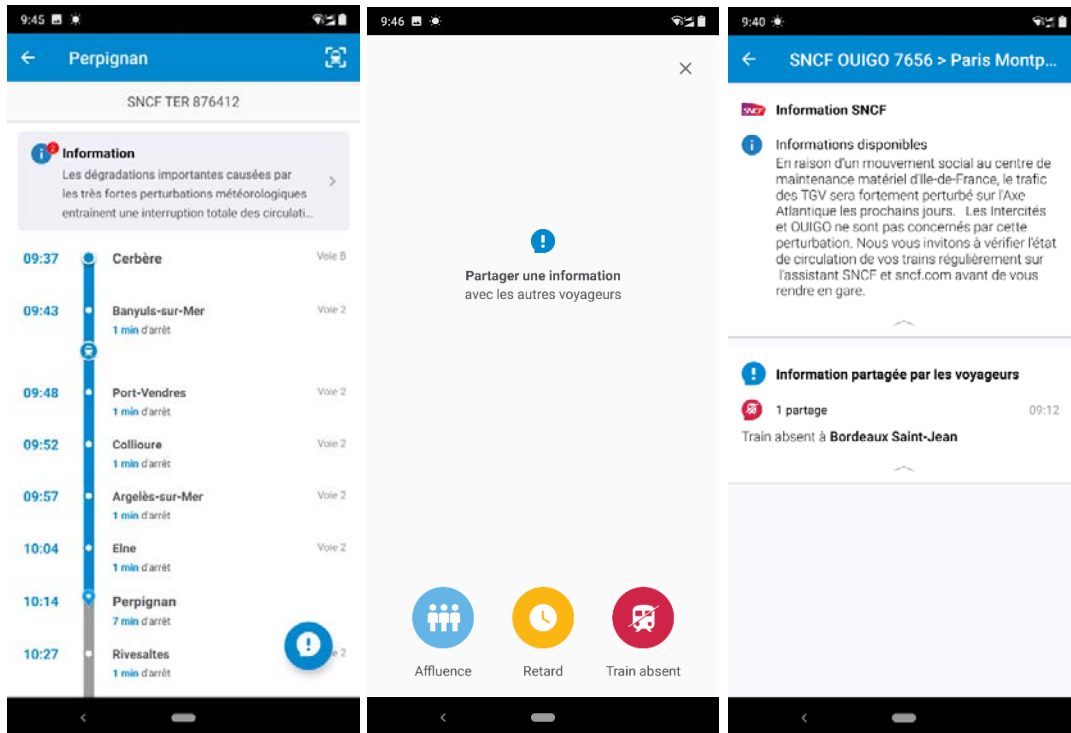


Figura 37 Captures de pantalla de l'aplicació Assistant SNCF on es veuen les tres pantalles dels avisos d'usuari

## 25. Conclusions

El projecte plantejat de forma inicial s'ha pogut resoldre de forma satisfactòria. Tots els punts i funcionalitats planificats s'han pogut realitzar i incorporar en la versió final de l'aplicació utilitzant tecnologies i plataformes de desenvolupament web modernes i recents, en plena expansió darrerament.

Per una banda, els *frameworks* PHP per al *frontend* han suposat una clara evolució pel que fa a la programació d'aplicacions basades en el web. Tot i tenir una corba d'aprenentatge possiblement més extensa que el propi codi pur PHP, permeten la realització d'interfícies i la utilització d'elements de forma més senzilla que programar-ho tot des de zero.

En aquest projecte, s'ha fet servir ReactJS, que gràcies als seus estats i la transmissió dels mateixos entre components, ha permès crear una aplicació interactiva amb molts elements que s'obtenen de forma asíncrona de forma ràpida, senzilla i totalment transparent per a l'usuari. A més, amb l'ús de les seves eines de desenvolupador, és fàcil detectar possibles errors i fer-ne un bon *debugging* per provar les diferents funcionalitats que es van incorporant.

Un altre avantatge, han estat l'ús de components de tercers que es poden incorporar directament a React, ja que funcionalitats que serien complicades per programar, es poden afegir a l'aplicació només coneixent quin ús se'n pot fer i què se n'obté llegint la seva documentació.

Per altra banda, en el *backend*, s'ha aprofundit amb PHP i l'ús de la programació orientada a objectes en aquest àmbit. Aplicar-ho en el desenvolupament d'una API REST ha estat de gran utilitat perquè ha permès l'emmagatzematge dels valors de les variables dels diferents objectes en la pròpia execució de PHP. D'aquesta forma, generar la resposta es podia fer de forma més senzilla i ràpida sol·licitant els valors i retornant els quals interessava en cada moment. Per últim, també ha permès que alguns objectes tinguessin altres objectes implementats com a variables, creant una potent relació entre ells, similar a la que s'ha emmagatzemat a la base de dades.

Relacionat amb els dos aspectes, les proves realitzades han sigut de gran utilitat ja que han permès detectar errors i elements que no s'havien tingut presents durant la programació i s'han pogut arreglar abans de l'entrega final. Igualment, han influït els consells i explicacions de professionals de l'àmbit de la programació web i usabilitat d'interfícies, tant des del cantó de la universitat com del personal.

Per la vessant personal, com a autor d'aquest projecte, estic plenament satisfet del resultat del mateix. Ha estat tot un repte personal ja que no havia realitzat mai un treball d'aquesta l'envergadura. Per un cantó, per el repte tècnic. Ha estat el primer cop que treballava amb

un *framework* PHP, i de fet, al principi no estava del tot convençut a fer-lo servir. Primerament, a l'hora de fer l'elecció, ja que en trobem forces per triar i després per aprendre'l a utilitzar-lo. Aquest darrer pas va suposar diverses hores de formació, seguint tutorials i cursos en línia, a banda de la revisió d'alguns articles i textos sobre aquest llenguatge de programació. La progressió a l'hora de redactar el codi també ha estat evolutiva, ja que hi ha parts de l'aplicació programades durant les fases més primerenques que es podrien tornar a redactar de forma més eficient amb fórmules de codi que he anat coneixent a mesura que avançava en el desenvolupament de l'aplicació.

Un cop finalitzat el projecte, he pogut constatar els avantatges que han suposat la utilització d'aquests *frameworks*, tal i com he desgranat a l'inici d'aquestes conclusions, i la conveniència de l'elecció presa a l'inici del projecte. Aquest fet ha de suposar que el projecte no acabi aquí i hi hagi l'expectativa de poder-lo mantenir en un futur. A més, haver-me endinsat en l'ús de frameworks vull que em permeti enfrontar futurs projectes que em puguin aparèixer utilitzant aquestes tècniques.

Tot i això, el camí no ha anat acompanyat de facilitats, si no, que he hagut de buscar molts recursos per tal de resoldre les problemàtiques que se m'han plantejat i alguns no han tingut el resultat desitjat. De fet, ha sigut imprescindible complir amb la política de mantenir un sistema de còpies de seguretat per restaurar l'aplicació en cas de fallada greu.

Per un altre cantó, realitzar aquesta memòria seguint els passos per realitzar un projecte ha estat de gran utilitat. Tot i emprar moltes hores per a la seva redacció, ha servit per seguir uns passos concrets en el procés del treball i establir una metodologia en el seu procés. També per tal que quedi constància de tota la feina i treball realitzats. Això també m'ha servit per veure en quins àmbits necessito tenir una visió més àmplia, sobretot en aspectes de planificació perquè he hagut de refer alguns plantejaments a mesura que avançava en el desenvolupament i que no havia contemplat.

Tots aquests aspectes m'han permès adquirir nous coneixements que es podran posar en pràctica en nous projectes professionals que puguin sorgir en un futur, a més de poder aplicar tots els elements apresos de les diferents matèries que he cursat durant el meu pas pel grau en multimèdia. Ha estat gratament satisfactori veure plasmats tots aquests coneixements en el resultat final d'aquest projecte, sobretot els dels àmbits de programació web, arquitectura de la informació, disseny gràfic i gestió de projectes.

Finalment, per acabar d'acomplir amb els objectius del projecte, la intenció és poder concretar la disposició d'aquesta aplicació cap al públic general i que sigui accessible pels diferents usuaris que puguin sentir-se implicats en l'ús d'aquesta aplicació. La voluntat és que l'aplicació pugui tenir recorregut i que es puguin realitzar les diferents actualitzacions i millores que s'han plantejat.

# Annex 1. Lliurables del projecte

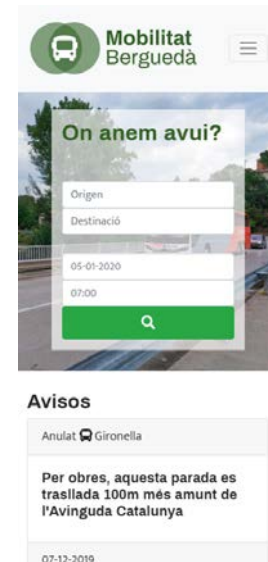
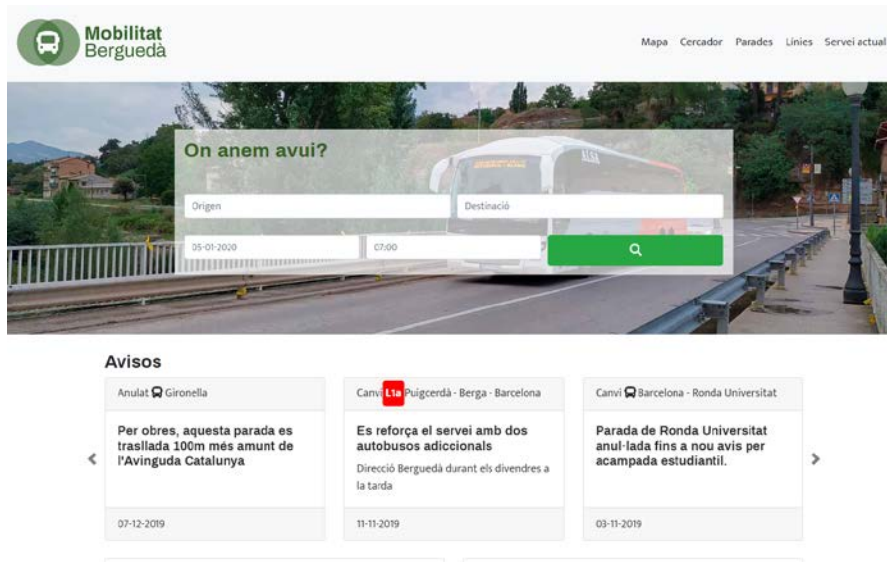
<b>Fitxers del projecte (PAC_FINAL_prj)</b>	
<b>backend.zip</b>	Inclou la carpeta api que conté tots els fitxers PHP que generen l'API.
<b>frontend.zip</b>	Conté la carpeta <i>mapamobilitat</i> (nom en clau) amb els arxius del codi font del frontend en react. <ul style="list-style-type: none"><li>- Build, carpeta amb la versió de producció 1.0.0, es pot executar en un servidor php directament</li><li>- Node_modules, son les biblioteques externes necessàries per executar el projecte.</li><li>- Public i src, arxius del codi font del projecte, necessaris si es vol executar en mode de desenvolupament.</li></ul>
<b>mobilitatbergueda.sql</b>	Fitxer que conté l'estructura de la base de dades amb les dades introduïdes fins al 5 de gener de 2019.

<b>Documentació del projecte (PAC_FINAL_doc)</b>	
<b>PAC_FINAL_mem.pdf</b>	Memòria del projecte en format PDF.
<b>PAC_FINAL_vid.mp4</b>	Vídeo de presentació-defensa del projecte en format MP4 amb la locució de les diapositives de la presentació.
<b>PAC_FINAL_diap.pdf</b>	PDF amb les diapositives de la presentació
<b>PAC_FINAL_prs.pdf</b>	Presentació escrita-visual de l'aplicació amb les seves característiques i elements més destacats

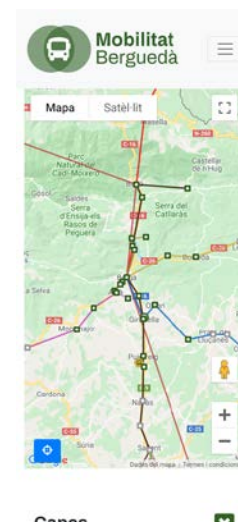
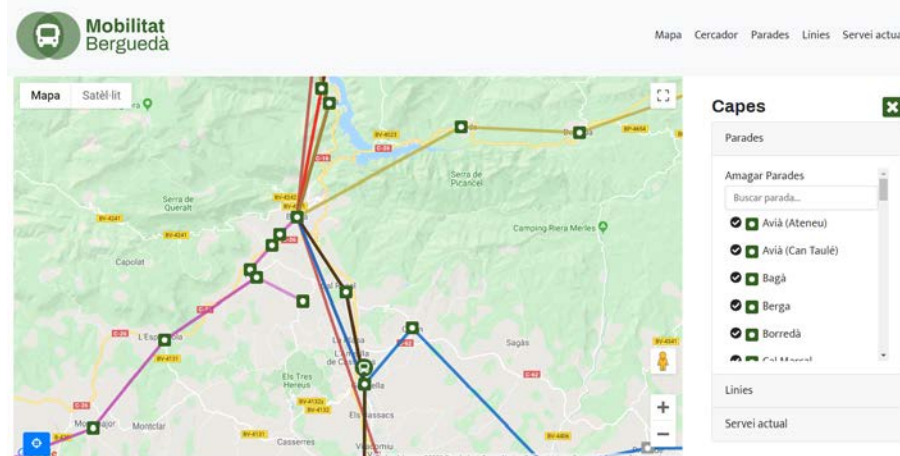
# Annex 2. Captures de pantalla

En aquest annex es plasmen diferents captures de pantalla de la versió 1.0.0 de l'aplicació 'Mobilitat Berguedà'. Es compara la versió escriptori amb la versió per a dispositius mòbils.

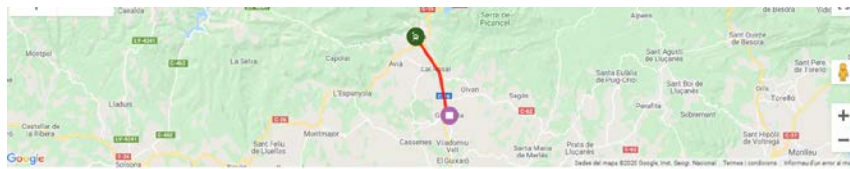
## 1. Portada



## 2. Mapa interactiu



### 3. Resultat d'una cerca



#### Itinerari

#### De Berga a Gironella

Origen: 08600 Berga, Barcelona, Spain  
 Destinació: 08680 Gironella, Barcelona, Spain  
 El 05-01-2020 a les 07:00

[Modifica itinerari](#)

Recorregut a peu fins a parada Berga (44 m - 1 minut)

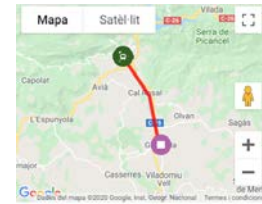
#### Resultats

	Origen	Destinació
L1a	07:00	07:12
L2	08:00	08:12
L1a	14:30	14:42
L2	15:50	16:02

07:00 Berga  
 L1a Servei de Berga a Barcelona - Estació del Nord  
 Direcció Barcelona - Estació del Nord

#### Avisos

No hi ha avisos destacats



#### Itinerari

#### De Berga a Gironella

Origen: 08600 Berga, Barcelona, Spain  
 Destinació: 08680 Gironella, Barcelona, Spain  
 El 05-01-2020 a les 07:00

[Modifica itinerari](#)

Recorregut a peu fins a parada Berga (44 m - 1 minut)

#### Resultats

	Origen	Destinació
L1a	07:00	07:12
L2	08:00	08:12

### 4. Detall d'una parada

Mapa Cercador Parades Línies Servei actual

← Parades

### Berga

8600 Berga  
Administració d'autobusos

#### Horaris en aquesta parada

Destinació:  
Avià (Ateneu)

Berga → Avià (Ateneu)

De dilluns a divendres (no festius)	14:45
-------------------------------------	-------

Avià (Ateneu) → Berga

De dilluns a divendres (no festius)	09:25 15:40 16:55 19:05
-------------------------------------	-------------------------

← Parades

### Berga

8600 Berga  
Administració d'autobusos

#### Horaris en aquesta parada

Destinació:  
Avià (Ateneu)

Berga → Avià (Ateneu)

De dilluns a divendres (no festius)	14:45
-------------------------------------	-------

### 5. Detall d'una línia

Mapa Cercador Parades Línies Servei actual

← Línies

### L1a Puigcerdà - Berga - Barcelona

Mostra mapa del recorregut

#### Avisos

Carri

**Es reforça el servei amb dos autobusos addicionals**  
 Direcció Berguedà durant els divendres a la tarda

11-11-2019

- Llívia
- Puigcerdà
- Alp
- Bagà
- Guardiola de Berguedà
- Berga
- Cal Rosal
- Gironella
- Puig-reig
- Navàs
- Balsareny
- Sallent
- La Botjosa
- Barcelona - Ronda Universitat
- Barcelona - Estació del Nord

Direcció:

Dia de la setmana:

← Línies

### L1a Puigcerdà - Berga - Barcelona

Mostra mapa del recorregut

- Llívia
- Puigcerdà
- Alp
- Bagà
- Guardiola de Berguedà
- Berga
- Cal Rosal
- Gironella
- Puig-reig
- Navàs
- Balsareny
- Sallent
- La Botjosa
- Barcelona - Ronda Universitat



## 6. Detall d'una expedició

**L1a** Puigcerdà - Berga - Barcelona

### Servei des de Barcelona - Estació del Nord a Berga

Sortida a les 14:30. Arribada a les 16:11.  
5 parades  
Circula diumenges i festius

**14:30** Barcelona - Estació del Nord  
**14:40** Barcelona - Ronda Universitat  
**15:46** Sallent  
**15:56** Navàs  
**16:11** Berga

**Avisos**  
En aquests moments no hi ha avisos vigents.  
**Incidències assenyalades avui**  
15:24: Retard de 5 minuts al pas per Barcelona - Ronda Universitat

[Informa d'una incidència](#)

**L1a** Puigcerdà - Berga - Barcelona

### Servei des de Barcelona - Estació del Nord a Berga

Sortida a les 14:30. Arribada a les 16:11.  
5 parades  
Circula diumenges i festius  
[Mostra mapa del recorregut](#)

**14:30** Barcelona - Estació del Nord  
**14:40** Barcelona - Ronda Universitat  
**15:46** Sallent  
**15:56** Navàs  
**16:11** Berga

**Avisos**  
En aquests moments no hi ha avisos vigents.  
**Incidències assenyalades avui**  
15:24: Retard de 5 minuts al pas per Barcelona - Ronda Universitat

[Informa d'una incidència](#)

## 7. Widget cercador amb calendari

**Mobilitat Berguedà**

Mapa Cercador Parades Línies Servei actual

**Cercador**  
Gironella Puig-reig  
05-01-2020 07:00

**Línies**

- L1a** Puigcerdà - Berga - Barcelona
- L1b** Andorra - Berga - Barcelona
- L1c** Berga - UAB - Barcelona
- L2** La Pobla de Lillet - Manresa
- L3** Berga - Vic
- L4** Berga - Ripoll
- L5** Berga - Solsona
- L5b** Berga - Graugès
- L9** La Pobla de Lillet - Tarragona - Cambrils
- TD1** Berga - Cercs - La Nou de Berguedà

Un projecte fet per:  
Ricard Codina

**Mobilitat Berguedà**

Universitat Oberta de Catalunya

**Mobilitat Berguedà**

Mapa Cercador

**Cercador**  
Berga  
Gironella

05-01-2020

gener 2020

di dt dc dj dv ds dg

30 31 1 2 3 4

6 7 8 9 10 11 12

13 14 15 16 17 18 19

20 21 22 23 24 25 26

27 28 29 30 31 1 2

3 4 5 6 7 8 9

el Nord universitat

**15:56** NAVAS

## 8. Pàgina de l'administrador

**Mobilitat Berguedà**

Inici administrador

- Parades
- Línies
- Expedicions
- Avisos
- Avisos dels usuaris

[Desconnectar](#)

**Parades**  
[Afegir](#)

**Berguedà**

Avià (Ateneu)	✍	✕
Avià (Can Taulé)	✍	✕
Bagà	✍	✕
Berga	✍	✕
Borredà	✍	✕
Cal Marçal	✍	✕
Cal Rosal	✍	✕
Cercs	✍	✕
Gironella	✍	✕
Graugès	✍	✕
Guardiola de Berguedà	✍	✕
L'Espunyola	✍	✕
La Nou de Berguedà	✍	✕
La Pobla de Lillet	✍	✕
La Rodonella	✍	✕
La Vallidà	✍	✕
Montmajor	✍	✕
...	✍	✕

**Parades**  
[Afegir](#)

**Berguedà**

Avià (Ateneu)	✍	✕
Avià (Can Taulé)	✍	✕
Bagà	✍	✕
Berga	✍	✕
Borredà	✍	✕
Cal Marçal	✍	✕
Cal Rosal	✍	✕
Cercs	✍	✕
Gironella	✍	✕
Graugès	✍	✕
Guardiola de Berguedà	✍	✕
L'Espunyola	✍	✕
La Nou de Berguedà	✍	✕
La Pobla de Lillet	✍	✕
La Rodonella	✍	✕
La Vallidà	✍	✕
Montmajor	✍	✕
Olvan	✍	✕
Polígon Industrial La Vallidà	✍	✕
...	✍	✕

## **Annex 3. Llibre d'estil**

En aquest annex, es troba el llibre d'estil que s'ha seguit per desenvolupar el projecte.

Llibre d'estil web

**Mobilitat Berguedà**

# Tipografia

Per aquesta web s'utilitzaran dues tipografies diferents, una pels títols superiors i una altra pels títols inferiors i la resta de textos. L'interlineat dels textos serà de 14 px.

A l'exemple següent hi ha indicat el tipus de lletra i la mida per cada tipus de text de la web:

**H1 Archivo Bold 30px**

**H2 Archivo Bold 24px**

H2 Mukta Mahee Extra Light 24px

**H3 Mukta Mahee Bold 16px**

**H4 Mukta Mahee Bold 14px**

P Mukta Mahee Regular 12px

Títols  
(diferents nivells)

Paràgrafs

Archivo Bold

abcdefghijklmnopqrstuvwxy

ABCDEFGHIJKLMNOPQRSTUVWXYZ

0123456789.,?!

Mukta Mahee Regular

abcdefghijklmnopqrstuvwxy

ABCDEFGHIJKLMNOPQRSTUVWXYZ

0123456789.,?!

Mukta Mahee Bold

abcdefghijklmnopqrstuvwxy

ABCDEFGHIJKLMNOPQRSTUVWXYZ

0123456789.,?!

Mukta Mahee Extra Light

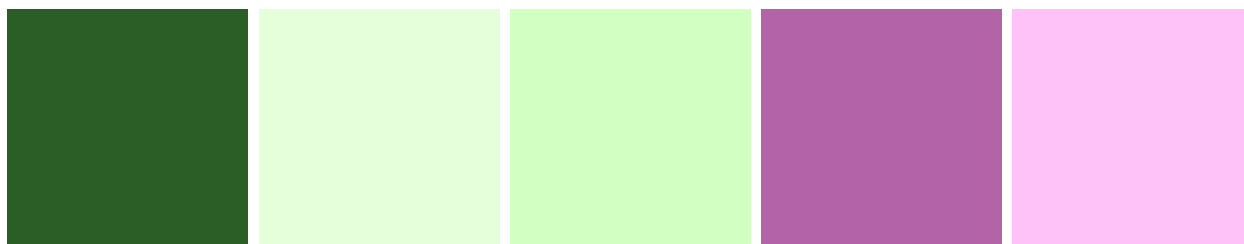
abcdefghijklmnopqrstuvwxy

ABCDEFGHIJKLMNOPQRSTUVWXYZ

0123456789.,?!

# Colors

El color principal és el verd i es combina amb el seu complementari, a més del blanc i el negre per als textos i títols de la pàgina.



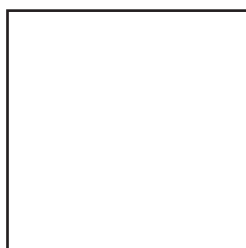
**R: 51**  
**G: 97**  
**B: 34**  
**#336122**

**R: 229**  
**G: 255**  
**B: 219**  
**#E5FFDB**

**R: 210**  
**G: 255**  
**B: 194**  
**#D2FFC2**

**R: 179**  
**G: 100**  
**B: 169**  
**#B364A9**

**R: 255**  
**G: 194**  
**B: 248**  
**#FFC2F8**



**R: 255**  
**G: 255**  
**B: 255**  
**#FFFFFF**



**R: 0**  
**G: 0**  
**B: 0**  
**#000000**

## Logotip

El color principal és el verd i es combina amb el seu complementari, a més del blanc i el negre per als textos i títols de la pàgina.

El logotip principal és el següent, i es pot utilitzar tant en vertical com horitzontal.



El logotip es pot utilitzar sobre fons blanc o de color, amb les seves corresponents variacions a tinta blanca i negra.



Igualment, es pot fer un ús de **variacions cromàtiques** del logotip per identificar diferents serveis que s'ofereixin dins de la mateixa marca.



## Icones

Les icones faran servir les que es troben dins de la biblioteca **Font Awesome Pro 5** que disposa de multitud d'icones per a tot tipus d'usos. S'ha comprat la llicència per a ús educatiu i es pot integrar directament al lloc web a través del CSS.



## Annex 4. Tests d'usuari

Per tal de procedir als tests d'usuari, es va procedir al disseny d'una plantilla amb diversos escenaris a realitzar. A continuació, es poden veure els fulls de resposta dels tres participants a les proves.

<b>Nom:</b>		
<b>Equip:</b>		
<b>Objectiu</b>	<b>Temps</b>	<b>Observacions</b>
Itinerari Berga-Barcelona 27 de desembre a les 10:25		
Informació de la parada de Guardiola de Berguedà		
Autobusos que circulen actualment		
Afegir incidència en un bus		
Consultar incidència d'un bus		
Veure horari Vic > Berga d'un dimarts		
Veure sortides de Gironella a Barcelona		
Indicar al mapa la parada de Vilada		
Geolocalització al mapa		
Veure el mapa de la línia 4		
Situació d'un bus concret		

Data:

Firma participant:

Observacions:



## TEST USUARI APP 'Mobilitat Berguedà'

Nom: MARC CANTURRI CASOLIVA		
Equip: SAMSUNG A70 - ANDROÏD - NAVEGADOR SAMSUNG		
Objectiu	Temps	Observacions
Itinerari Berga-Barcelona 27 de desembre a les 10:25	24"	
Informació de la parada de Guardiola de Berguedà	14"	
Autobusos que circulen actualment	7"	<del>canvia</del>
Afegir incidència en un bus	30"	canvia MOM "no preuati"
Consultar incidència d'un bus	25"	no queda clar quin en expedició
Veure horari Vic > Berga d'un dimarts	20"	
Veure sortides de Gironella a Barcelona	1'27"	mana avall
Indicar al mapa la parada de Vilada	20"	
Geolocalització al mapa	—	botó poc integrat amb la resta
Veure el mapa de la línia 4	5"	
Situació d'un bus concret	24"	color icame, no escau

Data: 23/12/19

Firma participant:



Observacions:

múltiples parades en una mateixa població es podrien unificar

## TEST USUARI APP 'Mobilitat Berguedà'

Nom: JAUME FIGULS CASAS		
Equip: DELL DESKTOP WINDOWS 10 - GOOGLE CHROME		
Objectiu	Temps	Observacions
Itinerari Berga-Barcelona 27 de desembre a les 10:25	1'15"	Per moltes hores 24h. no queda clar clic recomen <sup>apen</sup> at
Informació de la parada de Guardiola de Berguedà	10"	
Autobusos que circulen actualment	12"	
Afegir incidència en un bus	42"	no queda clar hora sortida al llistat.
Consultar incidència d'un bus	15"	
Veure horari Vic > Berga d'un dimarts	38"	
Veure sortides de Gironella a Barcelona	45"	queda massa avall
Indicar al mapa la parada de Vilada	20"	buscades no agafa accents.
Geolocalització al mapa	15"	
Veure el mapa de la línia 4	8"	
Situació d'un bus concret	35"	ícoma no es veu clarament

Data: 23/12/19

Firma participant:




Observacions:

## TEST USUARI APP 'Mobilitat Berguedà'

Nom: DIVINA LOPEZ GORRETA		
Equip: PORTÀTIL HP - WINDOWS 10 - GOOGLE CHROME		
Objectiu	Temps	Observacions
Itinerari Berga-Barcelona 27 de desembre a les 10:25	35"	
Informació de la parada de Guardiola de Berguedà	7"	
Autobusos que circulen actualment	6"	
Afegir incidència en un bus	1' 58"	Botó incidència reconegut. No queda clar el botó
Consultar incidència d'un bus	22"	destaca més els autobusos
Veure horari Vic > Berga d'un dimarts	15"	
Veure sortides de Gironella a Barcelona	52"	queda massa avall
Indicar al mapa la parada de Vilada	8"	
Geolocalització al mapa	30"	busca el botó al punt de capes
Veure el mapa de la línia 4	25"	
Situació d'un bus concret	20"	Costa identificar la icona.

Data: 23/12/19

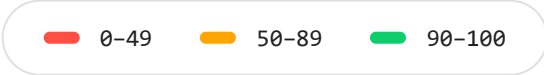
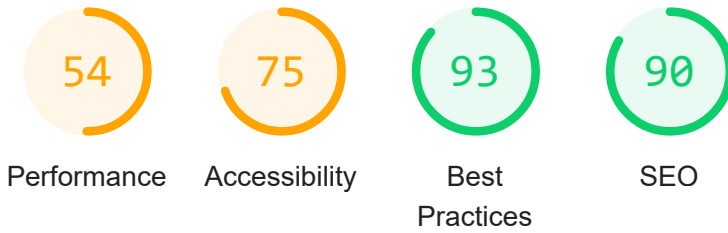
Firma participant:



Observacions:

## **Annex 5. Informe del test d'usabilitat**

A continuació, s'incorpora el resultat de l'informe d'usabilitat generat per Chrome Dev Tools sobre la versió 0.6.0 de l'aplicació. Les conclusions d'aquest informe s'expliquen al punt 17.2 de la memòria.



- Print Summary
- Print Expanded
- Copy JSON
- Save as HTML
- Save as JSON
- Open in Viewer
- Toggle Dark Theme

Progressive Web App



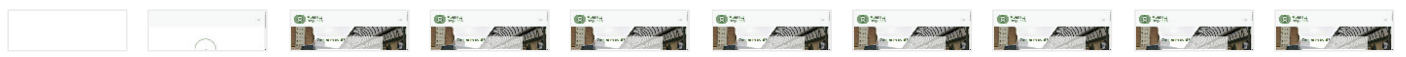
# Performance

## Metrics

▲ First Contentful Paint	4.6 s	▲ First Meaningful Paint	5.0 s
■ Speed Index	4.6 s	▲ First CPU Idle	7.6 s
■ Time to Interactive	6.3 s	▲ Max Potential First Input Delay	390 ms

[View Trace](#)

Values are estimated and may vary. The performance score is based only on these metrics.





**Opportunities** — These suggestions can help your page load faster. They don't directly affect the Performance score.

Opportunity	Estimated Savings
▲ Serve images in next-gen formats	0.9 s

Image formats like JPEG 2000, JPEG XR, and WebP often provide better compression than PNG or JPEG, which means faster downloads and less data consumption. [Learn more](#).

Show 3rd-party resources (0)

URL	Size	Potential Savings
 /img/bus_nord.jpg (mobilitat.ricardcodina.cat)	415 KB	184 KB
 /img/logo_UOC.png (mobilitat.ricardcodina.cat)	28 KB	17 KB

■ Preload key requests	0.49 s
------------------------	--------

Consider using `` to prioritize fetching resources that are currently requested later in page load. [Learn more](#).

Show 3rd party resources (0)

URL	Potential Savings
/precache-manifest.1d354b0...js (mobilitat.ricardcodina.cat)	490 ms

■ Eliminate render-blocking resources 0.39 s ^

Resources are blocking the first paint of your page. Consider delivering critical JS/CSS inline and deferring all non-critical JS/styles. [Learn more](#).

Show 3rd-party resources (1)

URL	Size	Potential Savings
/css?family=Archivo:700 Mukta+Mahee:200,400,700&display=swap (fonts.googleapis.com)	1 KB	790 ms
...css/2.5971471f.chunk.css (mobilitat.ricardcodina.cat)	23 KB	940 ms
...css/main.f35671f5.chunk.css (mobilitat.ricardcodina.cat)	3 KB	490 ms

■ Preconnect to required origins 0.3 s ^

Consider adding `preconnect` or `dns-prefetch` resource hints to establish early connections to important third-party origins. [Learn more](#).

URL	Potential Savings
https://api.mobilitat.ricardcodina.cat	300 ms

■ Remove unused CSS 0.15 s ^

Remove dead rules from stylesheets and defer the loading of CSS not used for above-the-fold content to reduce unnecessary bytes consumed by network activity. [Learn more](#).


Show 3rd party resources (0)

URL	Size	Potential Savings
...css/2.5971471f.chunk.css (mobilitat.ricardcodina.cat)	23 KB	22 KB

■ Efficiently encode images 0.15 s ^

Optimized images load faster and consume less cellular data. [Learn more](#).

Show 3rd party resources (0)

URL	Size	Potential Savings
 /img/bus_nord.jpg (mobilitat.ricardcodina.cat)	415 KB	45 KB

**Diagnostics** — More information about the performance of your application. These numbers don't [directly affect](#) the Performance score.

▲ **Serve static assets with an efficient cache policy** — 8 resources found

A long cache lifetime can speed up repeat visits to your page. [Learn more](#).

Show 3rd-party resources (0)

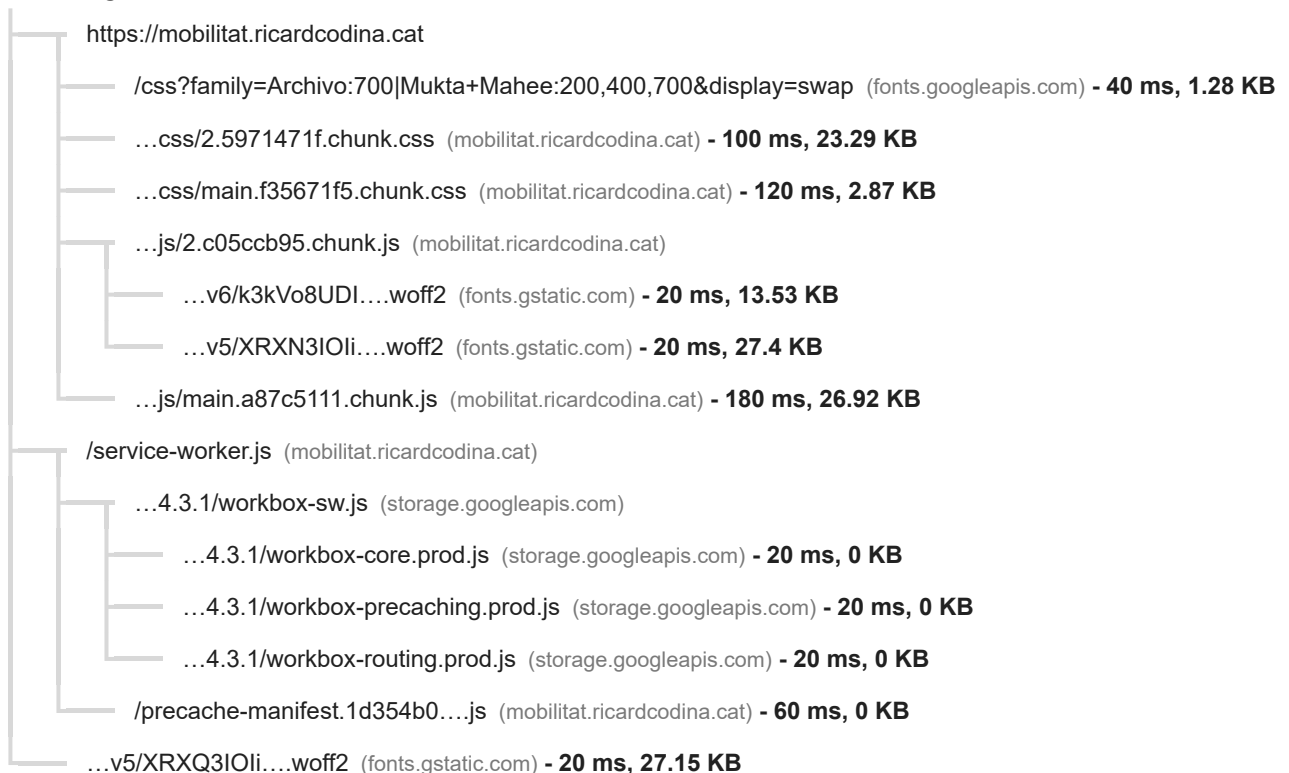
URL	Cache TTL	Size
/img/bus_nord.jpg (mobilitat.ricardcodina.cat)	None	415 KB
...js/2.c05ccb95.chunk.js (mobilitat.ricardcodina.cat)	None	292 KB
/img/logo_UOC.png (mobilitat.ricardcodina.cat)	None	29 KB
...js/main.a87c5111.chunk.js (mobilitat.ricardcodina.cat)	None	27 KB
...css/2.5971471f.chunk.css (mobilitat.ricardcodina.cat)	None	23 KB
...media/logo-blanc.5ae646f9.svg (mobilitat.ricardcodina.cat)	None	7 KB
...media/logo.d176d5a2.svg (mobilitat.ricardcodina.cat)	None	7 KB
...css/main.f35671f5.chunk.css (mobilitat.ricardcodina.cat)	None	3 KB

● **Avoid chaining critical requests** — 6 chains found

The Critical Request Chains below show you what resources are loaded with a high priority. Consider reducing the length of chains, reducing the download size of resources, or deferring the download of unnecessary resources to improve page load. [Learn more](#).

Maximum critical path latency: **660 ms**

*Initial Navigation*



● **Keep request counts low and transfer sizes small** — 39 requests • 1,349 KB

To set budgets for the quantity and size of page resources, add a budget.json file. [Learn more.](#)


Resource Type	Requests	Transfer Size
Total	39	1,349 KB
Image	7	458 KB
Other	22	399 KB
Script	3	395 KB
Font	3	68 KB
Stylesheet	3	27 KB
Document	1	2 KB
Media	0	0 KB
Third-party	12	145 KB

### Passed audits (15)

#### ● Properly size images — Potential savings of 27 KB

Serve images that are appropriately-sized to save cellular data and improve load time. [Learn more.](#)


Show 3rd-party resources (0)

URL	Size	Potential Savings
 /img/logo_UOC.png (mobilitat.ricardcodina.cat)	28 KB	27 KB

#### ● Defer offscreen images — Potential savings of 7 KB

Consider lazy-loading offscreen and hidden images after all critical resources have finished loading to lower time to interactive. [Learn more.](#)

Show 3rd-party resources (0)

URL	Size	Potential Savings
 ...media/logo-blanc.5ae646f9.svg (mobilitat.ricardcodina.cat)	7 KB	7 KB

#### ● Minify CSS

Minifying CSS files can reduce network payload sizes. [Learn more.](#)



If your build system minifies your CSS files automatically, ensure that you are deploying the production build of your application. You can check this with the React Developer Tools extension. [Learn more.](#)

#### ● Minify JavaScript

Minifying JavaScript files can reduce payload sizes and script parse time. [Learn more.](#)





If your build system minifies your JS files automatically, ensure that you are deploying the production build of your application. You can check this with the React Developer Tools extension. [Learn more.](#)

● Enable text compression — Potential savings of 31 KB ^

Text-based resources should be served with compression (gzip, deflate or brotli) to minimize total network bytes. [Learn more.](#)

Show 3rd-party resources (0)

URL	Size	Potential Savings
/parada/ (api.mobilitat.ricardcodina.cat)	17 KB	16 KB
/linia/ (api.mobilitat.ricardcodina.cat)	11 KB	9 KB
/avis/llistaAvisos.php?objecte=senseExpedicions (api.mobilitat.ricardcodina.cat)	4 KB	4 KB
/parada/?limit=8 (api.mobilitat.ricardcodina.cat)	3 KB	2 KB

● Server response times are low (TTFB) — Root document took 50 ms ^

Time To First Byte identifies the time at which your server sends a response. [Learn more.](#)



If you are server-side rendering any React components, consider using `renderToNodeStream()` or `renderToStaticNodeStream()` to allow the client to receive and hydrate different parts of the markup instead of all at once. [Learn more.](#)

● Avoid multiple page redirects ^

Redirects introduce additional delays before the page can be loaded. [Learn more.](#)



If you are using React Router, minimize usage of the `<Redirect>` component for [route navigations](#).

● Use video formats for animated content ^

Large GIFs are inefficient for delivering animated content. Consider using MPEG4/WebM videos for animations and PNG/WebP for static images instead of GIF to save network bytes. [Learn more](#)

● Avoids enormous network payloads — Total size was 1,349 KB ^

Large network payloads cost users real money and are highly correlated with long load times. [Learn more.](#)

Show 3rd-party resources (3)

URL	Size
/img/bus_nord.jpg (mobilitat.ricardcodina.cat)	415 KB
...js/2.c05ccb95.chunk.js (mobilitat.ricardcodina.cat)	292 KB
...js/2.c05ccb95.chunk.js?__WB_REVISION__=3b22035... (mobilitat.ricardcodina.cat)	292 KB
/gtag/js?id=G-BNN5TXET5C (www.googletagmanager.com)	75 KB
/img/logo_UOC.png (mobilitat.ricardcodina.cat)	29 KB
...v5/XRXN3IOli...woff2 (fonts.gstatic.com)	27 KB
...v5/XRXQ3IOli...woff2 (fonts.gstatic.com)	27 KB

URL	Size
...js/main.a87c5111.chunk.js (mobilitat.ricardcodina.cat)	27 KB
...js/main.a87c5111.chunk.js?__WB_REVISION__=71afa23... (mobilitat.ricardcodina.cat)	27 KB
...css/2.5971471f.chunk.css (mobilitat.ricardcodina.cat)	23 KB

● Avoids an excessive DOM size — 198 elements

A large DOM will increase memory usage, cause longer [style calculations](#), and produce costly [layout reflows](#). [Learn more](#).



Consider using a “windowing” library like `react-window` to minimize the number of DOM nodes created if you are rendering many repeated elements on the page. [Learn more](#). Also, minimize unnecessary re-renders using [shouldComponentUpdate](#), [PureComponent](#), or [React.memo](#) and [skip effects](#) only until certain dependencies have changed if you are using the Effect hook to improve runtime performance.

Statistic	Element	Value
Total DOM Elements		198
Maximum DOM Depth	<input aria-hidden="true" readonly="" tabindex="-1" value="">	17
Maximum Child Elements	<div class="iconesLiniaPortada">	10

● User Timing marks and measures

Consider instrumenting your app with the User Timing API to measure your app's real-world performance during key user experiences. [Learn more](#).



Use the React DevTools Profiler, which makes use of the Profiler API, to measure the rendering performance of your components. [Learn more](#).

● JavaScript execution time — 0.7 s

Consider reducing the time spent parsing, compiling, and executing JS. You may find delivering smaller JS payloads helps with this. [Learn more](#).

Show 3rd-party resources (1)

URL	Total CPU Time	Script Evaluation	Script Parse
Other	711 ms	259 ms	1 ms
...js/main.a87c5111.chunk.js (mobilitat.ricardcodina.cat)	331 ms	323 ms	7 ms
/gtag/js?id=G-BNN5TXET5C (www.googletagmanager.com)	77 ms	71 ms	5 ms
...js/2.c05ccb95.chunk.js (mobilitat.ricardcodina.cat)	55 ms	13 ms	41 ms

● Minimizes main-thread work — 1.2 s

Consider reducing the time spent parsing, compiling and executing JS. You may find delivering smaller JS payloads helps with this. [Learn more](#)

Category	Time Spent
Script Evaluation	667 ms

Category	Time Spent
Other	254 ms
Style & Layout	109 ms
Script Parsing & Compilation	57 ms
Rendering	46 ms
Parse HTML & CSS	28 ms
Garbage Collection	15 ms

- All text remains visible during webfont loads ^

Leverage the font-display CSS feature to ensure text is user-visible while webfonts are loading. [Learn more.](#)

- Minimize third-party usage — **Third-party code blocked the main thread for 0 ms** ^

Third-party code can significantly impact load performance. Limit the number of redundant third-party providers and try to load third-party code after your page has primarily finished loading. [Learn more.](#)

Third-Party	Size	Main-Thread Blocking Time
<a href="#">Google Tag Manager</a>	75 KB	0 ms
<a href="#">Other Google APIs/SDKs</a>	0 KB	0 ms



## Accessibility

These checks highlight opportunities to [improve the accessibility of your web app](#). Only a subset of accessibility issues can be automatically detected so manual testing is also encouraged.

**Names and labels** — These are opportunities to improve the semantics of the controls in your application. This may enhance the experience for users of assistive technology, like a screen reader.

- ▲ Buttons do not have an accessible name ^

When a button doesn't have an accessible name, screen readers announce it as "button", making it unusable for users who rely on screen readers. [Learn more.](#)

Failing Elements

```
button.btn.btn-success.btn-block.btn-lg
```

- ▲ Image elements do not have [alt] attributes ^

Informative elements should aim for short, descriptive alternate text. Decorative elements can be ignored with an empty alt attribute. [Learn more.](#)

## Failing Elements

`img.logoUniversitat`

### ▲ Form elements do not have associated labels ^

Labels ensure that form controls are announced properly by assistive technologies, like screen readers. [Learn more.](#)

## Failing Elements

`input.rbt-input-main.form-control.rbt-input`

`input.rbt-input-main.form-control.rbt-input`

`input#react-datetime-bootstrap-ee5c3b8d-3d1c-4b1f-96e1-97f39152e2e3.form-control.date-time`

`input#react-datetime-bootstrap-2a157d86-8817-428e-9f07-a1f0df26ded5.form-control.date-time`

### Contrast — These are opportunities to improve the legibility of your content.

### ▲ Background and foreground colors do not have a sufficient contrast ratio. ^

Low-contrast text is difficult or impossible for many users to read. [Learn more.](#)

## Failing Elements

`div.siglesLinia`

`div.siglesLinia.maBoto`

`div.siglesLinia.maBoto`

`div.siglesLinia.maBoto`

`div.siglesLinia.maBoto`

`div.siglesLinia.maBoto`

`div.siglesLinia.maBoto`

### Best practices — These items highlight common accessibility best practices.

### ▲ [id] attributes on the page are not unique ^

The value of an id attribute must be unique to prevent other instances from being overlooked by assistive technologies. [Learn more.](#)

## Failing Elements

`div#react-datetime-bootstrap`

### Additional items to manually check (11) — These items address areas which an automated testing tool cannot cover. [Learn more](#) in our guide on [conducting an accessibility review](#). ^

### ● The page has a logical tab order ^

Tabbing through the page follows the visual layout. Users cannot focus elements that are offscreen. [Learn more.](#)

- Interactive controls are keyboard focusable ^

Custom interactive controls are keyboard focusable and display a focus indicator. [Learn more.](#)
- Interactive elements indicate their purpose and state ^

Interactive elements, such as links and buttons, should indicate their state and be distinguishable from non-interactive elements. [Learn more.](#)
- The user's focus is directed to new content added to the page ^

If new content, such as a dialog, is added to the page, the user's focus is directed to it. [Learn more.](#)
- User focus is not accidentally trapped in a region ^

A user can tab into and out of any control or region without accidentally trapping their focus. [Learn more.](#)
- Custom controls have associated labels ^

Custom interactive controls have associated labels, provided by `aria-label` or `aria-labelledby`. [Learn more.](#)
- Custom controls have ARIA roles ^

Custom interactive controls have appropriate ARIA roles. [Learn more.](#)
- Visual order on the page follows DOM order ^

DOM order matches the visual order, improving navigation for assistive technology. [Learn more.](#)
- Offscreen content is hidden from assistive technology ^

Offscreen content is hidden with `display: none` or `aria-hidden=true`. [Learn more.](#)
- Headings don't skip levels ^

Headings are used to create an outline for the page and heading levels are not skipped. [Learn more.](#)
- HTML5 landmark elements are used to improve navigation ^

Landmark elements (`<main>`, `<nav>`, etc.) are used to improve the keyboard navigation of the page for assistive technology. [Learn more.](#)

## Passed audits (16) ^

- `[aria-*]` attributes match their roles ^

Each ARIA `role` supports a specific subset of `aria-*` attributes. Mismatching these invalidates the `aria-*` attributes. [Learn more.](#)
- `[role]`s have all required `[aria-*]` attributes ^

Some ARIA roles have required attributes that describe the state of the element to screen readers. [Learn more.](#)
- Elements with an ARIA `[role]` that require children to contain a specific `[role]` have all required children. ^

Some ARIA parent roles must contain specific child roles to perform their intended accessibility functions. [Learn more.](#)
- `[role]`s are contained by their required parent element ^

Some ARIA child roles must be contained by specific parent roles to properly perform their intended accessibility functions. [Learn more.](#)

● [role] values are valid ^

ARIA roles must have valid values in order to perform their intended accessibility functions. [Learn more.](#)

● [aria-\*] attributes have valid values ^

Assistive technologies, like screen readers, can't interpret ARIA attributes with invalid values. [Learn more.](#)

● [aria-\*] attributes are valid and not misspelled ^

Assistive technologies, like screen readers, can't interpret ARIA attributes with invalid names. [Learn more.](#)

● The page contains a heading, skip link, or landmark region ^

Adding ways to bypass repetitive content lets keyboard users navigate the page more efficiently. [Learn more.](#)

● Document has a <title> element ^

The title gives screen reader users an overview of the page, and search engine users rely on it heavily to determine if a page is relevant to their search. [Learn more.](#)

● <html> element has a [lang] attribute ^

If a page doesn't specify a lang attribute, a screen reader assumes that the page is in the default language that the user chose when setting up the screen reader. If the page isn't actually in the default language, then the screen reader might not announce the page's text correctly. [Learn more.](#)

● <html> element has a valid value for its [lang] attribute ^

Specifying a valid [BCP 47 language](#) helps screen readers announce text properly. [Learn more.](#)

● Links have a discernible name ^

Link text (and alternate text for images, when used as links) that is discernible, unique, and focusable improves the navigation experience for screen reader users. [Learn more.](#)

● Lists contain only <li> elements and script supporting elements (<script> and <template>). ^

Screen readers have a specific way of announcing lists. Ensuring proper list structure aids screen reader output. [Learn more.](#)

● List items (<li>) are contained within <ul> or <ol> parent elements ^

Screen readers require list items (`<li>`) to be contained within a parent `

` or `

` to be announced properly. [Learn more.](#)

● [user-scalable="no"] is not used in the <meta name="viewport"> element and the [maximum-scale] attribute is not less than 5. ^

Disabling zooming is problematic for users with low vision who rely on screen magnification to properly see the contents of a web page. [Learn more.](#)

● No element has a [tabindex] value greater than 0 ^

A value greater than 0 implies an explicit navigation ordering. Although technically valid, this often creates frustrating experiences for users who rely on assistive technologies. [Learn more.](#)

## Not applicable (14)

- [accesskey] values are unique

Access keys let users quickly focus a part of the page. For proper navigation, each access key must be unique. [Learn more.](#)

- <audio> elements contain a <track> element with [kind="captions"]

Captions make audio elements usable for deaf or hearing-impaired users, providing critical information such as who is talking, what they're saying, and other non-speech information. [Learn more.](#)

- <dl>'s contain only properly-ordered <dt> and <dd> groups, <script> or <template> elements.

When definition lists are not properly marked up, screen readers may produce confusing or inaccurate output. [Learn more.](#)

- Definition list items are wrapped in <dl> elements

Definition list items (`<dt>` and `<dd>`) must be wrapped in a parent `<dl>` element to ensure that screen readers can properly announce them. [Learn more.](#)

- <frame> or <iframe> elements have a title

Screen reader users rely on frame titles to describe the contents of frames. [Learn more.](#)

- <input type="image"> elements have [alt] text

When an image is being used as an `<input>` button, providing alternative text can help screen reader users understand the purpose of the button. [Learn more.](#)

- Presentational <table> elements avoid using <th>, <caption> or the [summary] attribute.

A table being used for layout purposes should not include data elements, such as the th or caption elements or the summary attribute, because this can create a confusing experience for screen reader users. [Learn more.](#)

- The document does not use <meta http-equiv="refresh">

Users do not expect a page to refresh automatically, and doing so will move focus back to the top of the page. This may create a frustrating or confusing experience. [Learn more.](#)

- <object> elements have [alt] text

Screen readers cannot translate non-text content. Adding alt text to `<object>` elements helps screen readers convey meaning to users. [Learn more.](#)

- Cells in a <table> element that use the [headers] attribute refer to table cells within the same table.

Screen readers have features to make navigating tables easier. Ensuring `<td>` cells using the `[headers]` attribute only refer to other cells in the same table may improve the experience for screen reader users. [Learn more.](#)

- <th> elements and elements with [role="columnheader"/"rowheader"] have data cells they describe.

Screen readers have features to make navigating tables easier. Ensuring table headers always refer to some set of cells may improve the experience for screen reader users. [Learn more.](#)

- [lang] attributes have a valid value

Specifying a valid [BCP 47 language](#) on elements helps ensure that text is pronounced correctly by a screen reader. [Learn more.](#)

- <video> elements contain a <track> element with [kind="captions"]

When a video provides a caption it is easier for deaf and hearing impaired users to access its information. [Learn more.](#)

- `<video>` elements contain a `<track>` element with `[kind="description"]`



Audio descriptions provide relevant information for videos that dialogue cannot, such as facial expressions and scenes. [Learn more.](#)



## Best Practices

- ▲ Does not use HTTP/2 for all of its resources — 22 requests not served via HTTP/2



HTTP/2 offers many benefits over HTTP/1.1, including binary headers, multiplexing, and server push. [Learn more.](#)

Show 3rd party resources (0)

URL	Protocol
<code>https://mobilitat.ricardcodina.cat</code>	http/1.1
<code>...css/2.5971471f.chunk.css</code> (mobilitat.ricardcodina.cat)	http/1.1
<code>...css/main.f35671f5.chunk.css</code> (mobilitat.ricardcodina.cat)	http/1.1
<code>...js/2.c05ccb95.chunk.js</code> (mobilitat.ricardcodina.cat)	http/1.1
<code>...js/main.a87c5111.chunk.js</code> (mobilitat.ricardcodina.cat)	http/1.1
<code>/meta.json</code> (mobilitat.ricardcodina.cat)	http/1.1
<code>/service-worker.js</code> (mobilitat.ricardcodina.cat)	http/1.1
<code>...media/logo.d176d5a2.svg</code> (mobilitat.ricardcodina.cat)	http/1.1
<code>...media/logo-blanc.5ae646f9.svg</code> (mobilitat.ricardcodina.cat)	http/1.1
<code>/img/logo_UOC.png</code> (mobilitat.ricardcodina.cat)	http/1.1
<code>/img/bus_nord.jpg</code> (mobilitat.ricardcodina.cat)	http/1.1
<code>/precache-manifest.1d354b0...js</code> (mobilitat.ricardcodina.cat)	http/1.1
<code>/index.html?__WB_REVISION__=a058217...</code> (mobilitat.ricardcodina.cat)	http/1.1
<code>...css/2.5971471f.chunk.css?__WB_REVISION__=3b22035...</code> (mobilitat.ricardcodina.cat)	http/1.1
<code>...css/main.f35671f5.chunk.css?__WB_REVISION__=71afa23...</code> (mobilitat.ricardcodina.cat)	http/1.1
<code>...js/2.c05ccb95.chunk.js?__WB_REVISION__=3b22035...</code> (mobilitat.ricardcodina.cat)	http/1.1
<code>...js/main.a87c5111.chunk.js?__WB_REVISION__=71afa23...</code> (mobilitat.ricardcodina.cat)	http/1.1
<code>...js/runtime-main.adb929d7.js?__WB_REVISION__=cdcd565...</code> (mobilitat.ricardcodina.cat)	http/1.1
<code>...media/logo-blanc.5ae646f9.svg?__WB_REVISION__=5ae646f...</code> (mobilitat.ricardcodina.cat)	http/1.1
<code>...media/logo.25bf045c.svg?__WB_REVISION__=25bf045...</code> (mobilitat.ricardcodina.cat)	http/1.1



URL	Protocol
...media/logo.d176d5a2.svg?__WB_REVISION__=d176d5a... (mobilitat.ricardcodina.cat)	http/1.1
...media/puntparada.9e4b6697.svg?__WB_REVISION__=9e4b669... (mobilitat.ricardcodina.cat)	http/1.1

## Passed audits (14)

### ● Avoids Application Cache

Application Cache is deprecated. [Learn more.](#)

### ● Uses HTTPS

All sites should be protected with HTTPS, even ones that don't handle sensitive data. HTTPS prevents intruders from tampering with or passively listening in on the communications between your app and your users, and is a prerequisite for HTTP/2 and many new web platform APIs. [Learn more.](#)

### ● Uses passive listeners to improve scrolling performance

Consider marking your touch and wheel event listeners as `passive` to improve your page's scroll performance. [Learn more.](#)

### ● Avoids document.write()

For users on slow connections, external scripts dynamically injected via `document.write()` can delay page load by tens of seconds. [Learn more.](#)

### ● Links to cross-origin destinations are safe

Add `rel="noopener"` or `rel="noreferrer"` to any external links to improve performance and prevent security vulnerabilities. [Learn more.](#)

### ● Avoids requesting the geolocation permission on page load

Users are mistrustful of or confused by sites that request their location without context. Consider tying the request to a user action instead. [Learn more.](#)

### ● Page has the HTML doctype

Specifying a doctype prevents the browser from switching to quirks-mode. [Learn more.](#)

### ● Avoids front-end JavaScript libraries with known security vulnerabilities

Some third-party scripts may contain known security vulnerabilities that are easily identified and exploited by attackers. [Learn more.](#)

### ● Detected JavaScript libraries

All front-end JavaScript libraries detected on the page. [Learn more.](#)

Name	Version
React	
Workbox	4.3.1

### ● Avoids requesting the notification permission on page load

Users are mistrustful of or confused by sites that request to send notifications without context. Consider tying the request to user gestures instead. [Learn more](#).

- Avoids deprecated APIs ^

Deprecated APIs will eventually be removed from the browser. [Learn more](#).

- Allows users to paste into password fields ^

Preventing password pasting undermines good security policy. [Learn more](#).

- No browser errors logged to the console ^

Errors logged to the console indicate unresolved problems. They can come from network request failures and other browser concerns. [Learn more](#)

- Displays images with correct aspect ratio ^

Image display dimensions should match natural aspect ratio. [Learn more](#).



## SEO

These checks ensure that your page is optimized for search engine results ranking. There are additional factors Lighthouse does not check that may affect your search ranking. [Learn more](#).

**Content Best Practices** — Format your HTML in a way that enables crawlers to better understand your app's content.

- ▲ Image elements do not have [alt] attributes ^

Informative elements should aim for short, descriptive alternate text. Decorative elements can be ignored with an empty alt attribute. [Learn more](#).

Failing Elements

`img.logoUniversitat`

**Additional items to manually check (1)** — Run these additional validators on your site to check additional SEO best practices. ^

- Structured data is valid ^

Run the [Structured Data Testing Tool](#) and the [Structured Data Linter](#) to validate structured data. [Learn more](#).

**Passed audits (9)** ^

- Has a `<meta name="viewport">` tag with width or initial-scale ^

Add a `<meta name="viewport">` tag to optimize your app for mobile screens. [Learn more.](#)

● Document has a `<title>` element ^

The title gives screen reader users an overview of the page, and search engine users rely on it heavily to determine if a page is relevant to their search. [Learn more.](#)

● Document has a meta description ^

Meta descriptions may be included in search results to concisely summarize page content. [Learn more.](#)

● Page has successful HTTP status code ^

Pages with unsuccessful HTTP status codes may not be indexed properly. [Learn more.](#)

● Links have descriptive text ^

Descriptive link text helps search engines understand your content. [Learn more.](#)

● Page isn't blocked from indexing ^

Search engines are unable to include your pages in search results if they don't have permission to crawl them. [Learn more.](#)

● robots.txt is valid ^

If your robots.txt file is malformed, crawlers may not be able to understand how you want your website to be crawled or indexed. [Learn more.](#)

● Document has a valid hreflang ^

hreflang links tell search engines what version of a page they should list in search results for a given language or region. [Learn more.](#)

● Document avoids plugins ^

Search engines can't index plugin content, and many devices restrict plugins or don't support them. [Learn more.](#)

**Not applicable (3)** ^

● Document has a valid `rel=canonical` ^

Canonical links suggest which URL to show in search results. [Learn more.](#)

● Document uses legible font sizes ^

Font sizes less than 12px are too small to be legible and require mobile visitors to "pinch to zoom" in order to read. Strive to have >60% of page text  $\geq 12$ px. [Learn more.](#)

● Tap targets are sized appropriately ^

Interactive elements like buttons and links should be large enough (48x48px), and have enough space around them, to be easy enough to tap without overlapping onto other elements. [Learn more.](#)



# Progressive Web App

These checks validate the aspects of a Progressive Web App. [Learn more.](#)

## Fast and reliable

- Page load is fast enough on mobile networks ^

A fast page load over a cellular network ensures a good mobile user experience. [Learn more.](#)

- Current page responds with a 200 when offline ^

If you're building a Progressive Web App, consider using a service worker so that your app can work offline. [Learn more.](#)

- start\_url responds with a 200 when offline ^

A service worker enables your web app to be reliable in unpredictable network conditions. [Learn more.](#)

## Installable

- Uses HTTPS ^

All sites should be protected with HTTPS, even ones that don't handle sensitive data. HTTPS prevents intruders from tampering with or passively listening in on the communications between your app and your users, and is a prerequisite for HTTP/2 and many new web platform APIs. [Learn more.](#)

- Registers a service worker that controls page and start\_url ^

The service worker is the technology that enables your app to use many Progressive Web App features, such as offline, add to homescreen, and push notifications. [Learn more.](#)

- Web app manifest meets the installability requirements ^

Browsers can proactively prompt users to add your app to their homescreen, which can lead to higher engagement. [Learn more.](#)

## PWA Optimized

- Redirects HTTP traffic to HTTPS ^

If you've already set up HTTPS, make sure that you redirect all HTTP traffic to HTTPS in order to enable secure web features for all your users. [Learn more.](#)

- Configured for a custom splash screen ^

A themed splash screen ensures a high-quality experience when users launch your app from their homescreens. [Learn more.](#)

- Sets a theme color for the address bar. ^

The browser address bar can be themed to match your site. [Learn more.](#)

- Content is sized correctly for the viewport ^

If the width of your app's content doesn't match the width of the viewport, your app might not be optimized for mobile screens. [Learn more.](#)

- Has a `<meta name="viewport">` tag with width or initial-scale ^

Add a `<meta name="viewport">` tag to optimize your app for mobile screens. [Learn more.](#)

● Contains some content when JavaScript is not available ^

Your app should display some content when JavaScript is disabled, even if it's just a warning to the user that JavaScript is required to use the app. [Learn more.](#)

● Provides a valid `apple-touch-icon` ^

For ideal appearance on iOS when users add a progressive web app to the home screen, define an `apple-touch-icon`. It must point to a non-transparent 192px (or 180px) square PNG. [Learn More.](#)

**Additional items to manually check (3)** — These checks are required by the baseline [PWA Checklist](#) but are not automatically checked by Lighthouse. They do not affect your score but it's important that you verify them manually. ^

● Site works cross-browser ^

To reach the most number of users, sites should work across every major browser. [Learn more.](#)

● Page transitions don't feel like they block on the network ^

Transitions should feel snappy as you tap around, even on a slow network. This experience is key to a user's perception of performance. [Learn more.](#)

● Each page has a URL ^

Ensure individual pages are deep linkable via URL and that URLs are unique for the purpose of shareability on social media. [Learn more.](#)

#### Runtime Settings

<b>URL</b>	https://mobilitat.ricardcodina.cat/
<b>Fetch time</b>	Dec 28, 2019, 9:35 AM GMT+1
<b>Device</b>	Emulated Desktop
<b>Network throttling</b>	150 ms TCP RTT, 1,638.4 Kbps throughput (Simulated)
<b>CPU throttling</b>	4x slowdown (Simulated)
<b>User agent (host)</b>	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.3945.88 Safari/537.36
<b>User agent (network)</b>	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/74.0.3694.0 Safari/537.36 Chrome-Lighthouse
<b>CPU/Memory Power</b>	1616

## Annex 6. Bibliografia

- Autoritat del Transport Metropolità. (2018). *TransMet Xifres 2018*. Barcelona: Autoritat del Transport Metropolità. Consultat el 24 / 09 / 2019, a [https://observatori.atm.cat/transmet-xifres/TransMet\\_Xifres\\_2018.pdf](https://observatori.atm.cat/transmet-xifres/TransMet_Xifres_2018.pdf)
- Coder Academy. (10 / 06 / 2016). *Top 32 Sites Built With ReactJS*. Recollit de Medium: <https://medium.com/@coderacademy/32-sites-built-with-reactjs-172e3a4bed81>
- Create React App Docs. (13 / 10 / 2019). *Supported Browsers and Features*. Recollit de <https://create-react-app.dev/docs/supported-browsers-features/>
- freeCodeCamp. (13 / 08 / 2016). *What is an API?* Recollit de <https://www.freecodecamp.org/news/what-is-an-api-in-english-please-b880a3214a82/>
- Grafton-Green, P. (18 / 03 / 2018). London transport: The TfL Tube lines with most signal failures and delays. *Evening Standard*. Recollit de <https://www.standard.co.uk/news/transport/revealed-the-tube-lines-with-the-most-signal-failures-and-delays-a3792006.html>
- Guzman, F. (24 / 05 / 2016). *Weblantropia*. Recollit de <http://www.weblantropia.com/2016/05/24/restful-api-que-es/>
- Hostinet. (sense data). *Lenguajes del lado servidor o cliente*. Recollit de <https://www.hostinet.com/formacion/general/lenguajes-del-lado-servidor-o-cliente/>
- Institut d'Estadística de Catalunya. (2018). *El municipi en xifres: Berguedà*. Recollit de <https://www.idescat.cat/emex/?id=14#h2>
- Jimdo 8Days. (12 / 08 / 2019). *Website Usability Testing: Guide with Expert Tips*. Recollit de <https://www.jimdo.com/blog/diy-guide-to-website-usability-testing/>
- Julián, G. (02 / 02 / 2012). *¿Qué es un ataque DDoS y cómo pararlo?* Recollit de GenBeta: <https://www.genbeta.com/web/son-los-ataques-ddos-efectivos-como-medio-de-protesta>
- Microsoft Docs. (30 / 01 / 2019). *Tecnologías web comunes del lado cliente*. Recollit de Guía de la arquitectura de aplicaciones .NET: <https://docs.microsoft.com/es-es/dotnet/architecture/modern-web-apps-azure/common-client-side-web-technologies>

Percival, R., & Slim, R. (sense data). *The Complete React Js & Redux Course - Build Modern Web Apps*. Recollit de Udemy: <https://www.udemy.com/course/the-complete-react-js-redux-course-build-modern-web-apps/>

Perficient Digital. (2018). *Mobile vs. Desktop Usage in 2018*. Estats Units. Recollit de <https://www.perficientdigital.com/insights/our-research/mobile-vs-desktop-usage-study>

Powell-Morse, A. (8 / 12 / 2016). *Waterfall Model: What Is It and When Should You Use It?* Recollit de Airbrake: <https://airbrake.io/blog/sdlc/waterfall-model>

RATP. (11 / 07 / 2017). *Un nouveau site pour vous!* Recollit de <https://www.ratp.fr/groupe-ratp/newsroom/services/un-nouveau-site-pour-vous>

React. (2013). *React - A JavaScript library for building user interfaces*. Recollit de <https://reactjs.org/>

Simform. (2019). *MongoDB vs MySQL: A Comparative Study on Databases*.

*SQL Injection Prevention*. (sense data). Recollit de [https://cheatsheetseries.owasp.org/cheatsheets/SQL\\_Injection\\_Prevention\\_Cheat\\_Sheet.html](https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html)

StatCounter. (2016). *Internet Usage Worldwide (2009-2016)*. Recollit de <https://boygeniusreport.files.wordpress.com/2016/11/stat-counter-mobile-web.jpg>

*The Evolution of the Web*. (2012). Recollit de <http://www.evolutionoftheweb.com/>

The PHP Group. (2001-2019). *PHP Manual*. Recollit de <https://www.php.net/manual/en/intro.pdo.php>

W3Schools. (2015). *ECMAScript 6 - ECMAScript 2015*. Recollit de Browser Support for ES6 : [https://www.w3schools.com/js/js\\_es6.asp](https://www.w3schools.com/js/js_es6.asp)

*S'inclouen també a la bibliografia tots els enllaços anotats als peus de pàgina que apareixen en aquesta memòria.*

## Annex 7. Vita

Ricard Codina Codina, té 26 anys, i és resident a Gironella, comarca del Berguedà, Catalunya. Actualment, i des del 2015, treballa com a maquinista a Rodalies de Catalunya, sota l'empresa de Renfe Viajeros, i és un gran apassionat del transport públic (del món ferroviari, en particular), les noves tecnologies i els mitjans de comunicació.

Des dels 15 anys, ha col·laborat a l'emissora local de ràdio de Berga, Ràdio Berga, en tasques de tècnic de so, producció d'àudio i feines variades en entorns digitals. En els últims anys, ha assumit també tasques de disseny i maquetació de la revista Aquí Berguedà i manteniment i desenvolupament del portal web amb el mateix nom, Aquí Berguedà.cat.

Aquest afany pels mitjans audiovisuals i les noves tecnologies digitals el va portar, un cop obtingut el títol de conducció de vehicles ferroviaris, a emprendre els estudis del grau en Multimèdia de la Universitat Oberta de Catalunya, el treball final del qual teniu entre mans.

Amb tots els coneixements adquirits durant el transcurs del grau i la culminació amb aquest projecte vol evolucionar una mica més professionalment, amb l'objectiu de permetre emprendre nous reptes i preparar les solucions adients per a cada cas que se li plantegi.